Nways

Event Logging System Messages Guide

SC30-3682-13

Nways

Event Logging System Messages Guide

SC30-3682-13

Note
Before using this document, read the general information under "Notices" on page vii.

Fourteenth Edition (July 1999)

This edition applies to: Version 3.3 of the IBM Nways Multiprotocol Access Services, Version 3.3 of the IBM Nways Multiprotocol Routing Services, Version 3.3 of the IBM Access Integration Services, Version 2.2 of the IBM Nways Multiprotocol Switched Services, IBM Nways Multiprotocol Switched Services Family Clients Version 1.0, and Version 5.0 of the 8371 Networking Multilayer Ethernet Switch, and to all subsequent releases and modifications until otherwise indicated in new editions or technical updates to this manual.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address below.

IBM welcomes your comments. A form for readers' comments is provided at the back of this publication. If the form has been removed, you may address your comments to:

International Business Machines Corporation
Design and Information Development
Department CGF
P.O. Box 12195
Research Triangle Park, NC 27709-9990
U.S.A.

You can find online copies of the 8371 documentation at this URL:

http://www.networking.ibm.com/did/8371bks.html

At this URL, you will also find a link to an online form for reader's comments.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make them available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of the intellectual property rights of IBM may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
500 Columbus Avenue
Thornwood NY 10594 USA

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement.

This document is not intended for production use and is furnished as is without any warranty of any kind, and all warranties are hereby disclaimed including the warranties of merchantability and fitness for a particular purpose.

Notice to Users of Online Versions of This Book

For online versions of this book, you are authorized to:

- Copy, modify, and print the documentation contained on the media, for use within your enterprise, provided you reproduce the copyright notice, all warning statements, and other required statements on each copy or partial copy.
- Transfer the original unaltered copy of the documentation when you transfer the related IBM product (which may be either machines you own, or programs, if the program's license terms permit a transfer). You must, at the same time, destroy all other copies of the documentation.

You are responsible for payment of any taxes, including personal property taxes, resulting from this authorization.

THERE ARE NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Some jurisdictions do not allow the exclusion of implied warranties, so the above exclusion may not apply to you.

Your failure to comply with the terms above terminates this authorization. Upon termination, you must destroy your machine-readable documentation.

Trademarks

The term “IBM” is a trademark of IBM Corporation in the United States or other countries or both.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Microsoft, Windows, Windows NT, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

Other company, product, and service names may be trademarks or service marks of others.

About This Manual

This manual explains how to interpret the messages logged with the Event Logging System (*ELS*). ELS messages are provided by these software products:

- Multiprotocol Routing Services (MRS)
- Multiprotocol Access Services (MAS)
- Access Integration Services (AIS)
- Multiprotocol Switched Services (MSS)
- Multiprotocol Switched Services Client (MSSC)
- Operational software for the 8371 Networking Multilayer Ethernet Switch

Who Should Read This Book

The intended user of this book is the provider of service and network operators.

How This Manual is Organized

This book contains an introductory chapter followed by separate chapters for each category of event. The categories are arranged in alphabetical order by chapter title.

Chapter 1. Introduction

This chapter describes how events are logged and how to interpret messages. Also described are the concepts of subsystem, event number, and logging level. A large part of the ELS functionality is based on commands that use the subsystem, event number, and logging levels as parameters.

Message Presentation

The format of the message explanations in this guide is as follows:

Level: Describes the logging level of the error message.

Short Syntax:
Shows the message that is displayed on the router console. This is a compressed form of the message.

Long Syntax:
Shows the expanded text of the message.

Description:
Explains the meaning of the error message.

Cause: Describes possible causes of the error that caused this message.

Action:
Specifies possible action to correct the error.

Causes of Events

Events monitored by the Event Logging System (ELS) occur continuously while the router is operating. Any of the following reasons can cause them.

- System activity
- Status changes
- Service requests
- Data transmission and reception
- Data and internal errors

When an event occurs, ELS receives data from the system that identifies the source and nature of the event. Then, ELS generates a message that uses the data received as part of the message.

Interpreting a Message

This section describes how to interpret a message generated by ELS. Figure 1 on page 2 shows the principal elements of a message and "Message Description" on page 2 describes the elements.

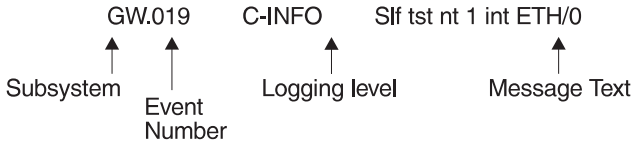


Figure 1. Elements of a Message

Message Element
Meaning

Subsystem
Subsystem is an abbreviation for a router component such as a protocol, packet forwarder, or interface. In Figure 1 on page 2, **GW** identifies the subsystem (gateway) through which this event occurred.

Examples of subsystems include ARP, IP, TKR, and X.25. On a router, the subsystems depend on the hardware and software configured for that router.

You can use the ELS **list subsystem** command to list the subsystems that are configured on your router.

Event Number
Event Number is a number that is assigned to each message within a subsystem. In Figure 1, the event number is **19** (within the GW subsystem).

The event number always appears with the subsystem abbreviation, for example, **GW.019**. The subsystem and event number together identify an *individual* event.

You can use the ELS **list subsystem** command to list the event within a subsystem.

Logging Level
Logging Level is a field that classifies each message by the type of event that generated it. Logging levels are as follows:

Logging Level
Type

UI - ERROR
Unusual internal errors

CI - ERROR
Common internal errors

UE - ERROR
Unusual external errors

CE - ERROR
Common external errors

ERROR
Includes all error levels above

U-INFO
Unusual Informational comment

C-INFO
Common Informational comment

INFO Includes all comment levels above

STANDARD
Includes all error levels and all comment levels (default)

P-TRACE
Per packet trace

U-TRACE
Unusual operation packet trace message

C-TRACE
Common operation packet trace message

TRACE
Includes all trace levels above

ALL Includes all logging levels

Message Text
Message text appears on the console screen in short form. In the sections that follow, variables such as *source_address* or *network* are replaced with actual data when the message displays on the console. These and other variables are replaced in the message text.

The variable *error_code* appearing in the message description (usually preceded by "rsn" or "reason") indicates the type of packet error detected. The next section describes the error and packet completion codes.

Code	Meaning
0	Packet successfully queued for output
1	Random, unidentified error
2	Packet not queued for output due to flow control reasons
3	Packet not queued because network is down
4	Packet not queued to avoid looping or bad broadcast
5	Packet not queued because destination host is down (only on networks where this can be detected)

When you send out an SNMP query, the response you get from the router is usually a 12-digit number, such as 1.3.6.1.4.1.1.1.3.4.85.31. This number refers to various information regarding your query, such as the ELS operating number (1.3.6.1.4.1.1.1.3), the object (.4), the ELS subsystem number (.85), and the event number (.31). For example, the eleventh digit, .85, is the numerical equivalent to the subsystem element X.25. The following list describes the subsystem numerical equivalents.

Subsystem	Numeric Equivalent	Subsystem	Numeric Equivalent	Subsystem	Numeric Equivalent
AAA	189	AI	110	ALLC	141
AP2	53	APPN	117	ARP	5
ATM	115	BAN	111	BBCM	134
BGP	104	BR	74	BRLY	198
BRS	3	BS	172	BTP	14
CAS	212	CEME	166	CMGR	222
COMP	113	DBG	174	DFP	171
DGW	151	DHC6	204	DHCP	146
DIAL	163	DLS	107	DLS2	175
DN	25	DNAV	43	DOUT	144

Subsystem	Numeric Equivalent	Subsystem	Numeric Equivalent	Subsystem	Numeric Equivalent
DS	208	DVM	22	ENCR	148
ENV	112	ES	216	ESC	133
ESIS	41	ETH	81	EVL	126
EZ	109	FDDI	88	FLT	2
FR	92	FSD	170	GCOM	169
GKI	223	GW	1	H225	221
H245	220	H323	219	ICM6	191
ICMP	11	IKE	210	ILEC	130
ILMI	119	IP	10	IPIF	203
IPPN	100	IPSP	159	IPV6	190
IPX	35	ISDN	99	ISIS	42
ISO	40	L2	162	LAG	214
LAPD	165	LCS	135	LEC	116
LECS	124	LES	123	LES2	176
LLC	103	LNM	102	LSA	136
LSI	155	MARS	128	MCF	105
MDM	206	MFC	18	MFC6	194
MLP	145	MMC	213	MPC	137
MPOA	156	NAT	167	NBS	114
NDP6	192	NDR	142	NHRP	131
NOT	127	PCA	161	PERF	200
PKI	215	PIM	196	PIM6	197
PLCY	209	PM	149	PPP	95
Q931	164	QLLC	152	R2MP	56
RIP	15	RIP6	195	RSVP	138
RTP	218	RTRF	207	SAAL	120
SACM	168	SCSP	140	SDLC	90
SE	157	SEC	147	SEST	158
SL	83	SNH	173	SNMP	21
SPF	17	SRLY	75	SRT	72
SRTF	205	STP	73	SVC	121
TCP	12	TFTP	19	TKR	84
TSDK	183	TSNC	180	TSNS	199
TSRC	181	TSRS	182	TSTD	178
UDP	13	UDP6	193	V25B	108
V34	143	VCRM	139	VLAN	150
VN	60	VOIC	211	VOIP	217
VRRP	177	WEBC	201	WEBH	202
WRS	101	XN	30	X25	85
X251	96	X252	97	X253	98
XTP	132	ZIP2	54		

Error and Packet Completion Codes

The console displays the following network information: *nt 1 int Eth/0* or *network 1, interface Eth/0* where:

- *1* is the network number (each network on the router is numbered sequentially from zero).

- *0* is the unit number (the interfaces on each hardware type are numbered sequentially from zero).

Ethernet and Token-Ring hardware addresses appear as a long hexadecimal number, such as X'020701003e2c'.

IP (Internet Protocol) addresses are printed as four decimal bytes separated by periods, such as 18.123.0.16.

IP Version 6 (IPv6) addresses are composed of 128 bits and are displayed as 8 sets of 4 hexadecimal numbers separated by colons, such as abcd:1234:0000:1234:5555:FFEE:7777:0123.

Chapter 2. AAA Protocol (AAA)

This chapter describes AAA Protocol (AAA) messages. For information on message content and how to use the message, refer to the Introduction.

AAA.001

Level: C-INFO

Short Syntax: AAA.001 AAAuthen: *string*

Long Syntax: AAA.001 AAA Authen Message: *string*

Description: Generic Authentication message

AAA.002

Level: C-INFO

Short Syntax: AAA.002 AAAuthen ppp: *string*

Long Syntax: AAA.002 AAA PPP Authen: *string*

Description: PPP Authentication message

AAA.003

Level: C-INFO

Short Syntax: AAA.003 AAAuthen login: *string*

Long Syntax: AAA.003 AAA Login Authen: *string*

Description: Login authentication message

AAA.004

Level: C-INFO

Short Syntax: AAA.004 AAAuthen tunnel: *string*

Long Syntax: AAA.004 AAA tunnel authen: *string*

Description: Tunnel authentication message

AAA.011

Level: C-INFO

Short Syntax: AAA.011 AAAuthor: *string*

Long Syntax: AAA.011 AAA Author: *string*

Description: Generic authorization Message for AAA

AAA.012

Level: C-INFO

Short Syntax: AAA.012 AAAuthor ppp: *string*

Long Syntax: AAA.012 AAA PPP Author: *string*

Description: PPP authorization Message for AAA

AAA.013

Level: C-INFO

Short Syntax: AAA.013 AAAuthor login: *string*

Long Syntax: AAA.013 AAA Login Author: *string*

Description: Login authorization Message for AAA

AAA.014

Level: C-INFO

Short Syntax: AAA.014 AAAuthor tunnel: *string*

Long Syntax: AAA.014 AAA Tunnel Author: *string*

Description: Tunnel authorization Message for AAA

AAA.021

Level: C-INFO

Description: Generic Accounting message for AAA

AAA.022

Level: C-INFO

Short Syntax: AAA.022 AAAcct ppp: *string*

Long Syntax: AAA.022 AAA PPP Acct: *string*

Description: PPP Accounting Message for AAA

AAA.023

Level: C-INFO

Short Syntax: AAA.023 AAAcct login: *string*

Long Syntax: AAA.023 AAA Login Acct: *string*

Description: Login Accounting Message for AAA

AAA.024

Level: C-INFO

Short Syntax: AAA.024 AAAcct tunnel: *string*

Long Syntax: AAA.024 AAA Tunnel Acct: *string*

Description: Tunnel Accounting Message for AAA

AAA.031

Level: C-INFO

Short Syntax: AAA.031 AAA: *An AAA message*

Long Syntax: AAA.031 AAA Message: *An AAA message*

Description: Generic Message for AAA

AAA.032

Level: C-INFO

Short Syntax: AAA.032 AAA ppp: *An AAA message*

Long Syntax: AAA.032 AAA Message: *An AAA message*

Description: Generic PPP Message for AAA

AAA.033

Level: C-INFO

Short Syntax: AAA.033 AAA login: *An AAA message*

Long Syntax: AAA.033 AAA Message: *An AAA message*

Description: Generic Login Message for AAA

AAA.034

Level: C-INFO

Short Syntax: AAA.034 AAA tunnel: *An AAA message*

Long Syntax: AAA.034 AAA Message: *An AAA message*

Description: Generic Tunnel Message for AAA

AAA.039

Level: C-INFO

Short Syntax: AAA.039 AAA: *An AAA message*

Long Syntax: AAA.039 AAA Message: *An AAA message*

Description: Generic Message for AAA

Chapter 3. Address Resolution Protocol (ARP)

This chapter describes Address Resolution Protocol (ARP) messages. For information on message content and how to use the message, refer to the Introduction.

ARP.001

Level: U-INFO

Short Syntax: ARP.001 Q ovf nt *network*

Long Syntax: ARP.001 Queue overflow net *network*

Description: An ARP packet was discarded, rather than being queued, because the queue of unprocessed ARP packets was too long. This means that ARP packets are arriving faster than they can be processed. Note that this event does not get counted in ELS, it is instead counted in the ARP console. The counters (kept per input network) can be read using the ARP>STATISTICS command, in the ''input packet overflows'' section.

Cause: This is often a symptom of a so-called ''ARP storm''. Some packets (usually an IP broadcast) arrive at hosts (usually a popular workstation) which do not recognize the destination address; they then attempt (in contravention of the Host specification) to forward the packet, but to do so they need the ARP mapping. Since they all receive the broadcast at the same time, they all attempt to forward the packet at the same time, and all do an ARP request at the same time.

Action: Prevail on the appropriate host manufacturer to bring their software into compliance with the specification. In the short term, it may be possible to disable the source of the packets, or cause it to use an address that the misbehaving hosts do recognize as a broadcast.

ARP.002

Level: P-TRACE

Short Syntax: ARP.002 Pkt in *operation_type hardware_address_space protocol_type* nt *network ID*

Long Syntax: ARP.002 Packet received *operation_type hardware_address_space protocol_type* net *network ID*

Description: An ARP packet of the type indicated has just arrived for processing.

ARP.003

Level: U-INFO

Short Syntax: ARP.003 Unkwn hdw *hardware_address_space* nt *network ID*

Long Syntax: ARP.003 Unknown hardware space *hardware_address_space* net *network ID*

Description: An incoming ARP packet was received on a network which is not using ARP for address translation in any protocol.

Cause: The gateway is misconfigured.

Action: Correct the configuration.

Cause: A protocol is in use on that network which requires the use of ARP, but the router does not support that protocol.

Action: None.

ARP.004

Level: UE-ERROR

Short Syntax: ARP.004 Bd hdw *hardware_address_space hardware_address_length* nt *network ID*

Long Syntax: ARP.004 Bad hardware address space *hardware_address_space hardware_address_length* nt *network ID*

Description: An incoming ARP packet was received with a hardware address space code or hardware address length which does not match the one which should be used on that network.

Cause: This is probably caused by an error (possible a byte swap problem) in some other equipment on the network.

Action: Use a network management tool to detect the source host and contact the manufacturer of the equipment and report the problem.

ARP.005

Level: P-TRACE

Short Syntax: ARP.005 Unkwn prt *protocol_type* nt *network ID*

Long Syntax: ARP.005 Unknown protocol type *protocol_type* net *network ID*

Description: An incoming ARP packet was received for a protocol for which the router is not using ARP for address translation.

Cause: The gateway is misconfigured.

Action: Correct the configuration.

Cause: A protocol is in use on that network which requires the use of ARP, but the router does not support that protocol.

Action: None.

ARP.006

Level: UE-ERROR

Short Syntax: ARP.006 Bd prt *protocol_type protocol_address_length* nt *network ID*

Long Syntax: ARP.006 Bad protocol address length *protocol_type protocol_address_length* net *network ID*

Description: An incoming ARP packet was received with a protocol address length which does not match the one which should be used on that network.

Cause: This is probably caused by an error (possible a byte swap problem) in some other equipment on the network.

Action: Use a network management tool to detect the source host and contact the manufacturer of the equipment and report the problem.

ARP.007

Level: U-TRACE

Short Syntax: ARP.007 Mk ent *hardware_address_space protocol_type* nt *network ID*

Long Syntax: ARP.007 Make translation entry *hardware_address_space protocol_type* net *network ID*

Description: An incoming ARP packet addressed to this host contained a mapping which was not in the translation cache. A new cache entry was filled in with the information in the packet.

ARP.008

Level: UE-ERROR

Short Syntax: ARP.008 Bd opc *operation_type hardware_address_space protocol_type* nt *network ID*

Long Syntax: ARP.008 Bad operation code *operation_type hardware_address_space protocol_type* net *network ID*

Description: An incoming ARP packet was received with an illegal operation code.

Cause: This is probably caused by an error (possibly a byte swap problem) in some other equipment on the network.

Action: Use a network management tool to detect the source host and contact the manufacturer of the equipment and report the problem.

ARP.009

Level: U-TRACE

Short Syntax: ARP.009 Rply *hardware_address_space protocol_type* nt *network ID*

Long Syntax: ARP.009 Reply sent *hardware_address_space protocol_type* net *network ID*

Description: An ARP reply is being sent as the result of a request for a translation from another host.

ARP.010

Level: UI-ERROR

Short Syntax: ARP.010 Err on rply nt *network ID*

Long Syntax: ARP.010 Transmission error on sending reply net *network ID*

Description: An outgoing ARP or inverse ARP reply packet was dropped as the result of some problem in the router.

Cause: There are many potential causes of this problem; an overloaded output queue, a down network, etc.

Action: Consult logging output from the relevant network subsystem for more information.

ARP.011

Level: U-TRACE

Short Syntax: ARP.011 Del ent *hardware_address_space protocol_type* nt *network ID*

Long Syntax: ARP.011 Deleting translation entry *hardware_address_space protocol_type* net *network ID*

Description: A translation cache entry timed out (which was not used or refreshed recently) has been deleted. Consult the ARP manual for more details on controlling this process.

ARP.012

Level: UI-ERROR

Short Syntax: ARP.012 No iorb fr rqst nt *network ID*

Long Syntax: ARP.012 No buffer for outgoing request packet net *network ID*

Description: An outgoing reply packet was dropped as the result of a lack of buffers in the router.

Cause: There are many potential causes of this problem; temporary overloads, etc.

Action: Consult logging output from the rest of the router for more information. If the problem persists, contact Customer Service.

ARP.014

Level: U-TRACE

Short Syntax: ARP.014 Rqst *hardware_address_space protocol_type* nt *network ID*

Long Syntax: ARP.014 Translation request sent *hardware_address_space protocol_type* net *network ID*

Description: An ARP translation request is being sent as the result of the transmission of a packet from the router for which the translation of another host's address is needed.

ARP.016

Level: P-TRACE

Short Syntax: ARP.016 unkn dst prot ad nt *network ID*

Long Syntax: ARP.016 Unknown destination protocol address net *network ID*

Description: This message is generated when an ARP request specifies an unknown protocol address (i.e. request not for this router).

Cause: ARP request for a host on this network that is not this router.

Action: None needed. This is normal for the ARP protocol, all requests are sent as broadcasts.

ARP.017

Level: UI-ERROR

Short Syntax: ARP.017 Rqst send failed rsn *reason_code* nt *network ID*

Long Syntax: ARP.017 Transmission of request failed for reason *reason_code* net *network ID*

Description: An outgoing ARP request packet was dropped as the result of some problem in the router. The reason_code gives the cause.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

ARP.018

Level: UI-ERROR

Short Syntax: ARP.018 rcv: No mem for cache ent, prot *protocol_type* nt *network ID*

Long Syntax: ARP.018 receive: No memory for cache entry, protocol *protocol_type* net *network ID*

Description: During the input processing of an ARP packet, the router did not have memory available to make an ARP cache entry for the given protocol.

Cause: The router is extremely low on heap memory.

Action: Find some way to reduce memory usage.

ARP.019

Level: UI-ERROR

Short Syntax: ARP.019 xmt: No mem for cache ent, prot *protocol_type* nt *network ID*

Long Syntax: ARP.019 transmit: No memory for cache entry, protocol *protocol_type* net *network ID*

Description: During the output processing of an ARP packet, the router did not have memory available to make an ARP cache entry for the given protocol.

Cause: The router is extremely low on heap memory.

Action: Find someway to reduce memory usage.

ARP.020

Level: U-TRACE

Short Syntax: ARP.020 Inverse Rply sent *hardware_address_space protocol_type* nt *network ID*

Long Syntax: ARP.020 Inverse Reply sent *hardware_address_space protocol_type* net *network ID*

Description: An inverse ARP reply is being sent as the result of a request for a translation from another host.

ARP.021

Level: P-TRACE

Short Syntax: ARP.021 inv arp req drp, no prot addr for prot *protocol_type* nt *network ID*

Long Syntax: ARP.021 inverse ARP request dropped, no protocol address *protocol_type* nt *network ID*

Description: This message is generated when an inverse ARP request arrives but can not be answered and is discarded because the router does not have a protocol addresses for the requested protocol on the interface.

Cause: The router either does not have the protocol configured on the interface, or protocol initialization on the interface is not complete, or inverse ARP is not

configured for this protocol, inverse ARP is not supported for this protocol.

Action: None needed. This is normal.

Cause: If the protocol requested is AppleTalk, The router may still be in the process of going through its probe logic before the AppleTalk protocol address is valid.

Action: None needed. This is normal.

ARP.022

Level: U-TRACE

Short Syntax: ARP.022 Inv Rqst sent *hardware_address_space protocol_type* to *hardware_address* nt *network ID*

Long Syntax: ARP.022 Inverse Request sent *hardware_address_space protocol_type* to *hardware_address* net *network ID*

Description: An inverse ARP request is being sent in an attempt to inform the other side of our protocol address.

ARP.030

Level: U-INFO

Short Syntax: ARP.030 ATM CIP NtDwn: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.030 ATM CIP NetDown: Client protocol/proto addr *protocol_number/ protocol_address* nt *network ID*

Description: This client has received a net down up call. All channels and calls will be cleared. Upon receiving a NetUp upcall, the interface will attempt to reestablish all calls.

ARP.031

Level: U-INFO

Short Syntax: ARP.031 ATM CIP NtUp: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.031 ATM CIP NetUp: Clnt prot/addr *protocol_number/ protocol_address* net *network ID*

Description: This client has received a net up. If already up, this client will do nothing. If down, the client will register the address, place and recieve calls, and will reopen any configured PVCs.

ARP.032

Level: C-INFO

Short Syntax: ARP.032 ATM CIP AddrStateChg (Active): Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.032 ATM CIP AddrStateChg (Active): Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: This client has received an address state change from the switch. This means that the address ESI and SEL have been registered with the switch. The client can procede in setting up and receiving calls

---

ARP.033

Level: C-INFO

Short Syntax: ARP.033 ATM CIP UNI Vers rcved: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.033 ATM CIP UNI Vers rcved: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: This client has received a net down up call. All channels and calls will be cleared. Upon receiving a NetUp upcall, the interface will attempt to reestablish all calls.

ARP.034

Level: UI-ERROR

Short Syntax: ARP.034 ATM CIP GetAddrByHandle rc= *return_code*: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.034 ATM CIP GetAddrByHandle rc= *return_code*: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: While attempting to get the address from the switch, an error was detected.

ARP.035

Level: UI-ERROR

Short Syntax: ARP.035 ATM CIP LlcOpenCallSap rc= *return_code*: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.035 ATM CIP LlcOpenCallSap rc= *return_code*: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: While attempting to open a call sap, an error was detected. A call sap is required in order to place or receive ATM calls to a remote destination.

ARP.036

Level: UE-ERROR

Short Syntax: ARP.036 ATM CIP Addr Deactivated!: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.036 ATM CIP Addr Deactivated!: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: The ATM address for this client was deactivated. All calls are deleted. This client will be waiting for the address to be reactivated. PVCs will still remain operable.

ARP.037

Level: UE-ERROR

Short Syntax: ARP.037 ATM CIP Addr Refused!: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.037 ATM CIP Addr Refused!: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: The requested address has been refused by the switch.

Cause: The likely cause is that a duplicate MAC address is already registered with the switch.

ARP.038

Level: UI-ERROR

Short Syntax: ARP.038 ATM CIP AddrStChg unknown: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.038 ATM CIP AddrStChg unknown: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: The Address State Change function was invoked, but the requested state is unknown.

ARP.039

Level: UI-ERROR

Short Syntax: ARP.039 ATM CIP LecsListReport?:

Long Syntax: ARP.039 ATM CIP LecsListReport?:

Description: An internal malfunction. The specified function was invoked on a classical IP client for which no such function is defined.

ARP.040

Level: U-INFO

Short Syntax: ARP.040 ATM CIP ReceiveCall: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.040 ATM CIP ReceiveCall: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: A call was received by this client. ARP_87 will be displayed (Remote Client ATM Address) following ARP_40 if there is a valid Cdb.

ARP.041

Level: UE-ERROR

Short Syntax: ARP.041 ATM CIP HangUpCall (invld PCR): Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.041 ATM CIP HangUpCall (invld PCR): Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: A call was received by this client where the Peak Cell Rate specified was greater than the allowed maximum. The call release cause is RJT_IE_PARM_VALUE, PRM_FWD_PEAKRATE_LP.

ARP.042

Level: UE-ERROR

Short Syntax: ARP.042 ATM CIP OpenDataPath failr(*return_code*): Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.042 ATM CIP OpenDataPath failr(*return_code*): Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: When attempting to open up a data path with the specified parameters, a failure occured. The call will be hung up with the appropriate cause code.

ARP.043

Level: UE-ERROR

Short Syntax: ARP.043 ATM CIP atmRcvCallAck fail(*return_code*): Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.043 ATM CIP atmRcvCallAck fail(*return_code*): Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: When attempting to acknowledge the incoming call, a failure occured.

Cause: The cause is an internal control block problem.

ARP.044

Level: C-INFO

Short Syntax: ARP.044 ATM CIP PlaceCallAck: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.044 ATM CIP PlaceCallAck: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: A call that we have placed has been received and acknowledged by the remote destination. We will open up a data path to the remote side, and will begin transmitting and receiving on the VCC. ARP_87 will be displayed (Remote Client ATM Address) following ARP_44

ARP.045

Level: U-INFO

Short Syntax: ARP.045 ATM CIP atmArpDisconnectCall: NULL CORRELATOR received

Long Syntax: ARP.045 ATM CIP atmArpDisconnectCall: NULL CORRELATOR received

Description: A call was released immediately before we received it.

ARP.046

Level: U-INFO

Short Syntax: ARP.046 ATM CIP atmArpDisconnectCall: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.046 ATM CIP atmArpDisconnectCall: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: Either a call already active, or a call that we are placing has been released. The reason for the release is shown in additional ELS messages. This is a normal occurance. If the channel is required, we will reinitiate it. Control channels, for example are retried every 15 seconds until we connect to the server.

Cause: Either the network or the remote user has released the call.

ARP.047

Level: U-INFO

Short Syntax: ARP.047 ATM CIP atmArpDiscCall: rsn= *reason_code*, cause= *cause_code*, diagLen= *diag_len*, diagData[0]= *diag_data*

Long Syntax: ARP.047 ATM CIP atmArpDiscCall: rsn= *reason_code*, cause= *cause_code*, diagLen= *diag_len*, diagData[0]= *diag_data*

Description: The information in this message is the reason for which the call has been released.

ARP.048

Level: U-INFO

Short Syntax: ARP.048 ATM CIP atmArpDiscCall: vpi= *vcc_vpi*, vci= *vcc_vci*, AtmAddr= *vcc_remote_atm_address*

Long Syntax: ARP.048 ATM CIP atmArpDiscCall: vpi= *vcc_vpi*, vci= *vcc_vci*, AtmAddr= *vcc_remote_atm_address*

Description: The information in this message is the channel vpi/vci, and remote atm address of the channel that is being disconnected.

ARP.049

Level: U-INFO

Short Syntax: ARP.049 ATM CIP atmArpDiscCall WalkDwn PCR= *walk_down_PCR*, SCR= *walk_down_SCR*:Clnt prot/addr *protocol_num/ protocol_address* nt *network ID*

Long Syntax: ARP.049 ATM CIP atmArpDiscCall WalkDwn PCR= *walk_down_PCR*, SCR= *walk_down_SCR*:Clnt prot/addr *protocol_num/ protocol_address* nt *network ID*

Description: The call that was released, was released due to cell rate. The ATMARP client will attempt to walk down to commonly used data rates in order to establish a connection with the target listed in ARP_48.

Cause: Either the network or the remote user has released the call due to cell rate mismatches.

ARP.050

Level: UI-ERROR

Short Syntax: ARP.050 ATM CIP ArpDisconnectLeaf?:

Long Syntax: ARP.050 ATM CIP ArpDisconnectLeaf?:

Description: An internal malfunction. The specified function was invoked on a classical IP client for which no such function is defined.

ARP.051

Level: C-TRACE

Short Syntax: ARP.051 ATM CIP atmArpRcvFrame: (prot = *protocol_number*) nt *network ID*

Long Syntax: ARP.051 ATM CIP atmArpRcvFrame: (prot = *protocol_number*) nt *network ID*

Description: A 1483 encapsulated packet has been received for the protocol number in the message on the interface in the message. This will occur for all packets received if this trace point is turned on.

ARP.052

Level: UE-ERROR

Short Syntax: ARP.052 ATM CIP atmArpRcvFrame: Unknown prot = *protocol_number* nt *network ID*

Long Syntax: ARP.052 ATM CIP atmArpRcvFrame: Unknown prot = *protocol_number* nt *network ID*

Description: A packet with an unknown protocol ID has been received off of the specified network. This may or may not be expected traffic. In any event, the packet will be discarded. No forwarding will occur.

ARP.053

Level: UI-ERROR

Short Syntax: ARP.053 ATM CIP atmArpAddLeafAck?:

Long Syntax: ARP.053 ATM CIP atmArpAddLeafAck?:

Description: An internal malfunction. The specified function was invoked on a classical IP client for which no such function is defined.

ARP.054

Level: UI-ERROR

Short Syntax: ARP.054 ATM CIP atmArpInit Registr failure (rc= *return_code*): Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.054 ATM CIP atmArpInit Registr failure (rc= *return_code*): Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: This client has failed to register as a user to the underlying device driver and net handler. This client will be inoperable.

Action: Reboot the router and contact the appropriate service personelle.

ARP.055

Level: C-INFO

Short Syntax: ARP.055 ATM CIP atmArpInit Registr successfull: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.055 ATM CIP atmArpInit Registr successfull: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: This client has successfully registered with the underlying device driver and net handler. This is normal initialization.

ARP.056

Level: UI-ERROR

Short Syntax: ARP.056 ATM CIP atmArpInit OpnBffFrmSap Failed (rc= *return_code*): Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.056 ATM CIP atmArpInit OpnBffFrmSap Failed (rc= *return_code*): Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: This client has failed while opening a buffered frame sap. This is cause by an internal error. This client will be inoperable.

Action: Reboot the router and contact the appropriate service personelle.

ARP.057

Level: C-INFO

Short Syntax: ARP.057 ATM CIP atmArpInit Address Activation pending: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.057 ATM CIP atmArpInit Address Activation pending: Client protocol/address *protocol_number/ protocol_address* net *network ID*

Description: This client has initiated the sequence that registers the client ATM address with the switch. When the registration completes, another message of Address State change will be logged describing the status of the clients ATM address.

Action: No action required. This is normal processing.

ARP.058

Level: C-INFO

Short Syntax: ARP.058 ATM CIP atmArpInit Address Activation success: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.058 ATM CIP atmArpInit Address Activation success: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: This client has been successful at activating an address.

ARP.059

Level: CE_ERROR

Short Syntax: ARP.059 ATM CIP:AAL IE:Not prsnt, or Invld AAL type (x *AAL_type*)

Long Syntax: ARP.059 ATM CIP:AAL IE:Not present, or Invalid AAL type (x *AAL_type*)

Description: Invalid AAL type, AAL type should be AAL5

ARP.060

Level: CE_ERROR

Short Syntax: ARP.060 ATM CIP:AAL IE:Invld fwd max SDU sz (*fwd_max_SDU_size*)

Long Syntax: ARP.060 ATM CIP:AAL IE:Invalid forward maximum SDU size (*fwd_max_SDU_size*)

Description: Forward maximum SDU size is not valid

ARP.061

Level: CE_ERROR

Short Syntax: ARP.061 ATM CIP:AAL IE:Invld bak max SDU sz for P2P call (*bak_max_SDU_size*)

Long Syntax: ARP.061 ATM CIP:AAL IE:Invalid backward maximum SDU size for Point-to-Point Call (*bak_max_SDU_size*)

Description: For a point-to-point call, the backward maximum SDU size is too small. The call will be accepted, but for receive data only. ARP is not supported.

ARP.062

Level: CE_ERROR

Short Syntax: ARP.062 ATM CIP:AAL IE:Invld bak max SDU sz for P2MP call (*bak_max_SDU_size*)

Long Syntax: ARP.062 ATM CIP:AAL IE:Invalid backward maximum SDU size for Point-to-MultiPoint Call (*bak_max_SDU_size*)

Description: For a point-to-multipoint call, the backward maximum SDU size is invalid, should be zero or one.

ARP.066

Level: CE_ERROR

Short Syntax: ARP.066 ATM CIP:AAL IE:Invld SSCS type (x *SSCS_type*)

Long Syntax: ARP.066 ATM CIP:AAL IE:Invalid SSCS type (x *SSCS_type*)

Description: Invalid SSCS type, SSCS type should be null

ARP.067

Level: CE_ERROR

Short Syntax: ARP.067 ATM CIP:Cell Rate IE:Fwd SCR(CLP=0+1) excds max *fwd_sustainable_rate*

Long Syntax: ARP.067 ATM CIP:Cell Rate IE:Forward Sustainable Cell Rate(CLP=0+1) exceeds maximum *fwd_sustainable_rate*

Description: Forward Sustainable Cell Rate for low priority data exceeds maximum reserved cell rate

ARP.068

Level: CE_ERROR

Short Syntax: ARP.068 ATM CIP:Cell Rate IE:Fwd SCR(CLP=0) excds max *fwd_sustainable_rate*

Long Syntax: ARP.068 ATM CIP:Cell Rate IE:Forward Sustainable Cell Rate(CLP=0) exceeds maximum *fwd_sustainable_rate*

Description: Forward Sustainable Cell Rate for high priority data exceeds maximum reserved cell rate

ARP.069

Level: CE_ERROR

Short Syntax: ARP.069 ATM CIP:Cell Rate IE:Fwd PCR(CLP=0+1) excds max *fwd_peak_rate*

Long Syntax: ARP.069 ATM CIP:Cell Rate IE:Forward Peak Cell Rate(CLP=0+1) exceeds maximum *fwd_peak_rate*

Description: Forward Peak Cell Rate for low priority data exceeds maximum reserved cell rate

ARP.070

Level: CE_ERROR

Short Syntax: ARP.070 ATM CIP:Cell Rate IE:Bak SCR(CLP=0+1) excds max *bak_sustainable_rate*

Long Syntax: ARP.070 ATM CIP:Cell Rate IE:Backward Sustainable Cell Rate(CLP=0+1) exceeds maximum *bak_sustainable_rate*

Description: Backward Sustainable Cell Rate for low priority data exceeds maximum reserved cell rate

ARP.071

Level: CE_ERROR

Short Syntax: ARP.071 ATM CIP:Cell Rate IE:Bak SCR(CLP=0) excds max *bak_sustainable_rate*

Long Syntax: ARP.071 ATM CIP:Cell Rate IE:Backward Sustainable Cell Rate(CLP=0) exceeds maximum *bak_sustainable_rate*

Description: Backward Sustainable Cell Rate for high priority data exceeds maximum reserved cell rate

ARP.072

Level: CE_ERROR

Short Syntax: ARP.072 ATM CIP:Cell Rate IE:Bak PCR(CLP=0+1) excds max *bak_peak_rate*

Long Syntax: ARP.072 ATM CIP:Cell Rate

IE:Backward Peak Cell Rate(CLP=0+1) exceeds maximum *bak_peak_rate*

Description: Backward Peak Cell Rate for low priority data exceeds maximum reserved cell rate

---

ARP.073

Level: CE_ERROR

Short Syntax: ARP.073 ATM CIP:Bearer IE:Invld class (x *bearer_class*)

Long Syntax: ARP.073 ATM CIP:Bearer IE:Invalid class (x *bearer_class*)

Description: Invalid bearer class, bearer class should be class C or class X

ARP.074

Level: CE_ERROR

Short Syntax: ARP.074 ATM CIP:Bearer IE:Invld conn type (x *conn_type*)

Long Syntax: ARP.074 ATM CIP:Bearer IE:Invalid connection type (x *conn_type*)

Description: Invalid connection type, connection type should be point-to-point

---

ARP.075

Level: CE_ERROR

Short Syntax: ARP.075 ATM CIP:QOS IE:Invld fwd QOS class (x *fwd_QOS*)

Long Syntax: ARP.075 ATM CIP:QOS IE:Invalid forward QOS class (x *fwd_QOS*)

Description: Connection is best effort service, and forward Quality Of Service should be QOS class 0

ARP.076

Level: CE_ERROR

Short Syntax: ARP.076 ATM CIP:QOS IE:Invld bak QOS class (x *bak_QOS*)

Long Syntax: ARP.076 ATM CIP:QOS IE:Invalid backward QOS class (x *bak_QOS*)

Description: Connection is best effort, and backward Quality Of Service should be QOS class 0

ARP.077

Level: CE_ERROR

Short Syntax: ARP.077 ATM CIP:Calling Party addr IE not prsnt

Long Syntax: ARP.077 ATM CIP:Calling Party address IE not present

Description: Calling Party address IE is not present

ARP.078

Level: CE_ERROR

Short Syntax: ARP.078 ATM CIP:Calling Party Addr IE:Invld ATM addr lngth (*remote_addr_length*)

Long Syntax: ARP.078 ATM CIP:Calling Party Addr IE:Invalid ATM address length (*remote_addr_length*)

Description: Calling Party Address IE has invalid ATM address length

---

ARP.079

Level: CE_ERROR

Short Syntax: ARP.079 ATM CIP:Calling Party Addr IE:ATM addr fld scrn

Long Syntax: ARP.079 ATM CIP:Calling Party Addr IE:ATM address failed screening

Description: ATM address was verified and did not pass screening

---

ARP.080

Level: CE_ERROR

Short Syntax: ARP.080 ATM CIP:Calling Party Addr IE:Invld ATM addr

Long Syntax: ARP.080 ATM CIP:Calling Party Address IE:Invalid ATM address

Description: Format of ATM address is incorrect, only private ATM address format is supported

---

ARP.081

Level: CE_ERROR

Short Syntax: ARP.081 ATM CIP:BLLI IE:Invld L2 prtcl (x *l2prot*)

Long Syntax: ARP.081 ATM CIP:BLLI IE:Invalid Layer 2 protocol (x *l2prot*)

Description: BLLI IE contains an invalid Layer 2 protocol, Layer 2 protocol should be 12 (ISO 8802/2)

ARP.082

Level: UI-ERROR

Short Syntax: ARP.082 ATM CIP:ArpFix No Client Address match: Clnt prot *protocol_number* nt *network ID*

Long Syntax: ARP.082 ATM CIP:ArpFix No Client Address match: Client protocol *protocol_number* net *network ID*

Description: While attempting to set up a configured PVC or SVC, no match was found to determine the

correct client to associate the PVC or SVC with.

ARP.083

Level: UI-ERROR

Short Syntax: ARP.083 ATM CIP:ArpFix Invld user or frm sap hndl: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.083 ATM CIP:ArpFix Invalid user or frame sap handle: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: While attempting to set up a configured PVC or SVC, the client user handle or frame sap handle was NULL.

ARP.084

Level: UI-ERROR

Short Syntax: ARP.084 ATM CIP:ArpFix OpnDataPath Failure (rc= *return_code*): Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.084 ATM CIP:ArpFix OpnDataPath Failure (rc= *return_code*): Client protocol/addr *protocol_number/ protocol_address* network *network ID*

Description: While attempting to initialize the hardware to set up a specific PVC, a failure was detected.

ARP.085

Level: UI-ERROR

Short Syntax: ARP.085 ATM CIP:ArpFix Cll sap invld: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.085 ATM CIP:ArpFix Call sap invalid: Client protocol/address *protocol_number/ protocol_address* network *network ID*

Description: While attempting to set up a configured SVC, the client user does not have a valid call sap.

ARP.086

Level: UI-ERROR

Short Syntax: ARP.086 ATM CIP: atmPlaceCall Failure (rc= *return_code*): Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.086 ATM CIP: atmPlaceCall Failure (rc= *return_code*): Client protocol/address *protocol_number/ protocol_address* net *network ID*

Description: While attempting to set up a configured SVC, the services of the device driver returned a value other than SUCCESS.

ARP.087

Level: U-INFO

Short Syntax: ARP.087 ATM CIP: Remote station : AtmAddr= *vcc_remote_atm_address*

Long Syntax: ARP.087 ATM CIP: Remote station : AtmAddr= *vcc_remote_atm_address*

Description: Setting up a configured SVC. This is the ATM address of the remote client. This message precedes ARP_88 on a PlaceCall SUCCESS and follows ARP_86 on a PlaceCall Failure.. This message is also displayed following ARP_40 and ARP_44

ARP.088

Level: C-INFO

Short Syntax: ARP.088 ATM CIP: atmPlaceCall Success: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.088 ATM CIP: atmPlaceCall Success: Clnt protocol/address *protocol_number/ protocol_address* net *network ID*

Description: A call was successfully placed. This channel should show up on the new channel list. It has not yet been answered. When it is answered, a PlaceCallAck message will appear in the log.

ARP.089

Level: U-INFO

Short Syntax: ARP.089 ATM CIP: chan aged: vpi= *vcc_vpi*, vci= *vcc_vci*, AtmAddr= *vcc_remote_atm_address*

Long Syntax: ARP.089 ATM CIP: channel aged out: vpi= *vcc_vpi*, vci= *vcc_vci*, AtmAddr= *vcc_remote_atm_address*

Description: The channel has been disconnected due to inactivity. The information in this message is the channel vpi/vci, and remote atm address of the channel that is being disconnected.

ARP.090

Level: UE-ERROR

Short Syntax: ARP.090 ATM CIP:Disconnect of cntrl vcc: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.090 ATM CIP:Disconnect of control vcc: Client protocol/address *protocol_number/ protocol_address* net *network ID*

Description: An active control channel has been disconnected. Resolution of addresses not currently in the ARP cache will be disrupted until a new control channel is active.

ARP.092

Level: U-TRACE

Short Syntax: ARP.092 ATM CIP: Mk ent *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.092 ATM CIP: Make ATM Arp entry prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: An incoming ATM ARP packet addressed to this host contained a mapping which was not in the translation cache. A new cache entry was filled in with the information in the packet.

ARP.093

Level: U-TRACE

Short Syntax: ARP.093 ATM CIP: Mv ent *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.093 ATM CIP: Move ATM Arp entry prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: An incoming ATM ARP packet addressed to this host arrived on a fixed channel and contained a mapping which was in the translation cache but used a non-fixed channel. The ARP entry was updated to use the fixed channel.

ARP.094

Level: U-TRACE

Short Syntax: ARP.094 ATM CIP: Rslv ent *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.094 ATM CIP: Resolve ATM Arp entry prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: An incoming ATM ARP or InARP packet addressed to this host contained a mapping whose protocol address was in the ARP cache, but had no channel attached. The ARP entry was updated to use the ATM address provided in the ARP packet. The ARP entry was attached to this channel.

ARP.095

Level: U-TRACE

Short Syntax: ARP.095 ATM CIP: InArp Req sent *vpi/ vci protocol_type* nt *network ID*

Long Syntax: ARP.095 ATM CIP: Inverse Arp request sent vpi= *vpi*, vci= *vci* prot *protocol_type* net *network ID*

Description: An InARP translation request is being sent on channel with given vpi and vci in an attempt to find a protocol address for the destination ATM address.

ARP.096

Level: U-TRACE

Short Syntax: ARP.096 ATM CIP: Arp_send NULL channel detected, nt *network ID*

Long Syntax: ARP.096 ATM CIP: Arp_send NULL channel detected, net *network ID*

Description: An outgoing arp packet was to be transmitted, but the channel to the ARP Server is not active. The packet will be discarded.

ARP.097

Level: U-TRACE

Short Syntax: ARP.097 ATM CIP: Arp Req sent *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.097 ATM CIP: ATM Arp request sent prot/addr *protocol_number/ protocol_address* net *network ID*

Description: An ARP translation request is being sent to the ATM Arp server in an attempt to find an ATM address for the given protocol address

---

ARP.098

Level: CE_ERROR

Short Syntax: ARP.098 ATM CIP: Dup ent *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.098 ATM CIP: Dup ATM Arp entry prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: An incoming ATM ARP or InARP packet addressed to this host contained a mapping whose protocol address was in the ARP cache and had a channel associated, but had a different ATM Address associated. The ARP packet was ignored.

ARP.099

Level: P_TRACE

Short Syntax: ARP.099 Trace ARP/ATMARP frame

Long Syntax: ARP.099 Trace ARP/ATMARP frame

Description: Packet tracing for the the ATM ARP component.

ARP.100

Level: UE-ERROR

Short Syntax: ARP.100 DROP: Bridging not enabled on VCC (vpi= *vpi*, vci= *vci*), nt *network ID*

Long Syntax: ARP.100 DROP: Bridging not enabled on VCC (vpi= *vpi*, vci= *vci*), network *network ID*

Description: A frame was received on a bridge type

defined in RFC 1483. However, since bridging has not been enabled on this circuit, frame is being discarded.

Cause: In a point-to-point WAN connection, this indicates that bridging is enabled on one end point router, and disabled on another. This is an illegal configuration.

Action: Either enable proper bridging behavior on both ends of the circuit or disable bridging on the bridge ports connected to this VCC. In other words, you must enable or disable bridging at both ends of the circuit.

ARP.101

Level: C-INFO

Short Syntax: ARP.101 DROP: Bridge port (*portnum*) not fwding on VCC (vpi= *vpi*, vci= *vci*), nt *network ID*

Long Syntax: ARP.101 DROP: Bridge port (*portnum*) not forwarding on VCC (vpi= *vpi*, vci= *vci*), network *network ID*

Description: A bridge frame is being discarded as a bridge port is not in forwarding state.

Cause: It could be that port has just come up and is progressing from blocking to listening to learning to forwarding state, or that Spanning Tree Protocol has determined that this port should stay in blocked state as a backup port.

ARP.102

Level: UE-ERROR

Short Syntax: ARP.102 DROP: *source_mac*-> *dest_mac*, Frame to bdg port behav mismatch on VCC (vpi= *vpi*, vci= *vci*), nt *network ID*

Long Syntax: ARP.102 DROP: *source_mac*-> *dest_mac*, Frame to bridge port behavior mismatch on VCC= (vpi= *vpi*, vci= *vci*), network *network ID*

Description: A bridged frame has been received and is being discarded due to mismatch in the frame type versus the bridge port behavior.

Cause: Either a source routed frame was received on a bridge port where source routing is disabled, or a transparent frame was received on a bridge port where transparent bridging is disabled.

Action: Enable proper bridging behavior on both ends of the circuit, or disable bridging on the bridge ports connected to this VCC.

ARP.103

Level: UE-ERROR

Short Syntax: ARP.103 Unsupported bdg frame type = 0x *type*, VCC (vpi= *vpi*, vci= *vci*) on nt *network ID*

Long Syntax: ARP.103 Unsupported bridge frame type = 0x *type* from VCC (vpi= *vpi*, vci= *vci*) on network *network ID*

Description: An unsupported bridge frame type has been encountered and the frame has been discarded.

Cause: Either a 802.4 bridge frame, a 802.6 bridge frame, or a bridge frame with a bridge protocol ID that is not supported by RFC 1483 has been received.

Action: Ensure compatible bridging behavior is configured on both ends of the circuit and contact customer service if the problem still occurs.

ARP.104

Level: UI-ERROR

Short Syntax: ARP.104 Unrecgnz outgoing bdg frame type = *type* on VCC (vpi= *vpi*, vci= *vci*) on nt *network ID*

Long Syntax: ARP.104 Unrecognized outgoing bridge frame type = *type* on VCC (vpi= *vpi*, vci= *vci*) on network *network ID*

Description: An unrecognized outgoing bridge frame type. Bridge has asked the ATM interface to send out a frame whose type cannot be translated into the encapsulation defined in RFC 1483.

Cause: Software problem

Action: Contact customer service

ARP.105

Level: UE-ERROR

Short Syntax: ARP.105 Unsupported ethertype = 0x *etype* (OUI = 0x *oui*) on VCC (vpi= *vpi*, vci= *vci*) on nt *network ID*

Long Syntax: ARP.105 Unsupported ethernet type = 0x *etype* (OUI = 0x *oui*) on VCC (vpi= *vpi*, vci= *vci*) on network *network ID*

Description: An unsupported ethernet type has been encountered.

Cause: Software out of date or incompatible, contact customer service.

ARP.106

Level: U-INFO

Short Syntax: ARP.106 ATM CIP: Var *msgType* info: *desc* addr *atmAddr*.

Long Syntax: ARP.106 ATM CIP: Variable *msgType* information: *desc* addr *atmAddr*.

Description: Variable address information for a message.

ARP.107

Level: U-INFO

Short Syntax: ARP.107 ATM CIP: No MARS cntrl vcc in func *functionCall*.

Long Syntax: ARP.107 ATM CIP: No MARS control vcc while in function call *functionCall*.

Description: A MARS Request message is being sent to the ATM MARS Server.

ARP.108

Level: UI-ERROR

Short Syntax: ARP.108 ATM CIP: Client control blk is null in func *functionCall*.

Long Syntax: ARP.108 ATM CIP: Client control block is null in function call *functionCall*.

Description: An internal function call requires a pointer to a valid client control block which is null. Record function name and report problem.

ARP.109

Level: UI-ERROR

Short Syntax: ARP.109 ATM CIP: Event control blk is null in func *functionCall*.

Long Syntax: ARP.109 ATM CIP: Event control block is null in function call *functionCall*.

Description: An internal function call requires a pointer to a valid event control block which is null. Record function name and report problem.

ARP.110

Level: U-INFO

Short Syntax: ARP.110 Tx Q ovf in func *functionCall* nt *network*.

Long Syntax: ARP.110 Transmit queue overflow in function *functionCall* net *network*.

Description: A MARS packet was discarded, rather than being queued, because the queue of pending ARP/MARS packets to be transmitted was too long. This means that ARP/MARS packets are being sent faster than they can be processed.

ARP.111

Level: P-TRACE

Short Syntax: ARP.111 ATM CIP: *state* MARS Client msg proc for *msgType* a *action*.

Long Syntax: ARP.111 ATM CIP: *state* MARS Client message processing for *msgType* a *action*.

Description: This is the action being performed by the MARS Client.

ARP.112

Level: P-TRACE

Short Syntax: ARP.112 ATM CIP: Msg is a respone to an outstanding *msgType*.

Long Syntax: ARP.112 ATM CIP: Message is a respone to an outstanding *msgType*.

Description: This message contains a response from the MARS Server to a request that originated at this MARS Client.

ARP.113

Level: UE-ERROR

Short Syntax: ARP.113 ATM CIP: *msgType* msg rcv contains invalid *value*.

Long Syntax: ARP.113 ATM CIP: *msgType* message recived contains and invalid *value* value.

Description: A message was sent to a MARS Client containing an invalid value.

Cause: MARS Server has a problem.

Action: Contact Systems Administrator.

ARP.114

Level: U-TRACE

Short Syntax: ARP.114 ATM CIP: *message*.

Long Syntax: ARP.114 ATM CIP: *message*.

Description: This is the action being performed by the MARS Client.

ARP.115

Level: U-TRACE

Short Syntax: ARP.115 ATM CIP: *msgType* spec info [spln *srcPln* tpln *targPln* thtl *targHtl* tstl *tarStl*].

Long Syntax: ARP.115 ATM CIP: *msgType* specific information [spln *srcPln* tpln *targPln* thtl *targHtl* tstl *tarStl*].

Description: This is the specific message content being sent or received by the MARS Client.

ARP.116

Level: U-TRACE

Short Syntax: ARP.116 ATM CIP: *msgType* spec info [spln *srcPln* thtl *targhtl* tstl *targstl* tpln *targpln* tnum *tnumAddr* seqxy *msgPart* msn *marsSeq*].

Long Syntax: ARP.116 ATM CIP: *msgType* specific

information [spln *srcPln* thtl *targhtl* tstl *targstl* tpln *targpln* tnum *tnumAddr* seqxy *msgPart* msn *marsSeq*].

Description: This is the specific message content being sent or received by the MARS Client.

ARP.117

Level: P-TRACE

Short Syntax: ARP.117 ATM CIP: *msgType* spec info [spln *srcPln* thtl *targhtl* tstl *targstl* flags *msgFlags* tnum *tnumAddr* seqxy *msgPart* msn *marsSeq*].

Long Syntax: ARP.117 ATM CIP: *msgType* specific information [spln *srcPln* thtl *targhtl* tstl *targstl* flags *msgFlags* tnum *tnumAddr* seqxy *msgPart* msn *marsSeq*].

Description: This is the specific message content being sent or received by the MARS Client.

ARP.118

Level: U-TRACE

Short Syntax: ARP.118 ATM CIP: *msgType* spec info [spln *srcPln* thtl *targhtl* tstl *targstl* tpln *targpln* tnum *tnumAddr* msn *marsSeq*].

Long Syntax: ARP.118 ATM CIP: *msgType* specific information [spln *srcPln* thtl *targhtl* tstl *targstl* tpln *targpln* tnum *tnumAddr* msn *marsSeq*].

Description: This is the specific message content being sent or received by the MARS Client.

ARP.119

Level: U-INFO

Short Syntax: ARP.119 ATM CIP: *action* MARS Client proc for an exp *timType* tim.

Long Syntax: ARP.119 ATM CIP: *action* MARS Client proccessing for an expired *timType* timer.

Description: A timer action has taken place while during the processing of MARS Client messages.

ARP.120

Level: U-INFO

Short Syntax: ARP.120 ATM CIP: MARS Client Response timer exp for mar$optype: *opType* .

Long Syntax: ARP.120 ATM CIP: MARS Client Response timer exp for mar$optype: *opType* .

Description: This message gives the MARS operation type for the expired timer.

ARP.121

Level: P-TRACE

Short Syntax: ARP.121 ATM CIP: *action* MARS Client io proc for *reqType*.

Long Syntax: ARP.121 ATM CIP: *action* MARS Client I/O proccessing for *reqType*.

Description: Action being taken by the MARS Client as the result of a received message.

ARP.122

Level: UI-ERROR

Short Syntax: ARP.122 ATM CIP: Channel cntl blk is null in func *functionCall*.

Long Syntax: ARP.122 ATM CIP: Channel control block is null in function call *functionCall*.

Description: An internal function call requires a pointer to a valid channel control block which is null. Record function name and report problem.

ARP.123

Level: UI-ERROR

Short Syntax: ARP.123 ATM CIP: atmAddLeaf Failure (rc= *return_code*): Clnt prot/addr *protocol_number*/ *protocol_address* nt *network ID*

Long Syntax: ARP.123 ATM CIP: atmAddLeaf Failure (rc= *return_code*): Client protocol/address *protocol_number*/ *protocol_address* net *network ID*

Description: While attempting to add a leaf to a configured SVC, the services of the device driver returned a value other than SUCCESS.

ARP.124

Level: UI-ERROR

Short Syntax: ARP.124 ATM CIP: atmAddLeaf Failure destination: AtmAddr= *vcc_remote_atm_address*

Long Syntax: ARP.124 ATM CIP: atmAddLeaf Failure destination: AtmAddr= *vcc_remote_atm_address*

Description: While attempting to add a leaf to a configured SVC, the services of the device driver returned a value other than SUCCESS. This is the addresses of the remote station that we are attempting to establish a leaf with.

ARP.125

Level: C-INFO

Short Syntax: ARP.125 ATM CIP: atmAddLeaf Success: Clnt prot/addr *protocol_number*/ *protocol_address* nt *network ID*

Long Syntax: ARP.125 ATM CIP: atmAddLeaf Success: Clnt protocol/address *protocol_number/ protocol_address* net *network ID*

Description: A leaf was successfully added. This leaf should show up on the active channel list as a leaf to one of the VCs. It has not yet been answered. When it is answered, an AddLeafAck message will appear in the log.

ARP.126

Level: UI-ERROR

Short Syntax: ARP.126 ATM CIP: Problem processing Redirect list (rc= *return_code*).

Long Syntax: ARP.126 ATM CIP: Problem processing Redirect list (rc= *return_code*).

Description: While attempting to procces a the learned list of backup MARS Servers obtained from the MARS_REDIRECT message an error occured.

ARP.127

Level: UI-ERROR

Short Syntax: ARP.127 ATM CIP: Protocol control blk is null in func *functionCall*.

Long Syntax: ARP.127 ATM CIP: Protocol control block is null in function call *functionCall*.

Description: An internal function call requires a pointer to a valid Protocol control block which is null. Record function name and report problem.

ARP.129

Level: UE-ERROR

Short Syntax: ARP.129 Invalid TLV values *tlvRc*

Long Syntax: ARP.129 Invalid TLV values *tlvRc*

Description: A message was sent to a MARS Client containing an invalid TLV value.

Cause: MARS Server has a problem.

Action: Contact Systems Administrator.

ARP.130

Level: UE-ERROR

Short Syntax: ARP.130 Invalid version *opVersion*

Long Syntax: ARP.130 Invalid MARS operation version specified in message *opVersion*

Description: A message was sent to a MARS Client containing an invalid version.

Cause: MARS Server has a problem.

Action: Contact Systems Administrator.

ARP.131

Level: UE-ERROR

Short Syntax: ARP.131 Unsupported op value *opValue*

Long Syntax: ARP.131 Invalid MARS operation value specified in message *opValue*

Description: A message was sent to a MARS Client containing an invalid operation.

Cause: MARS Server has a problem.

Action: Contact Systems Administrator.

ARP.132

Level: CE-ERROR

Short Syntax: ARP.132 Out of sequence op type *opType*

Long Syntax: ARP.132 Sequence error in MARS operation type specified in message *opType*

Description: A message was sent to a MARS Client containing an invalid operation.

Cause: Timing error.

Action: Contact Systems Administrator if problem continues.

ARP.133

Level: U-INFO

Short Syntax: ARP.133 Registration with MARS server rc = *registerRc*

Long Syntax: ARP.133 The client has attempted to register with the MARS server and has received a response of *registerRc*.

Description: Nonzero response to Register request with MARS.

Cause: Normal.

Action: Contact Systems Administrator if problem continues.

ARP.134

Level: U-TRACE

Short Syntax: ARP.134 ATM CIP: *msgType action*: fixhdr[afn *addrFamily* pro *proType* snap *pSnap0 pSnap1 pSnap2 pSnap3 pSnap4* chksum *checksum* ext *extoff* ver *opVersion* shtl *addrTypeLen* sstl *subAddrTypeLen*] on nt *network ID*.

Long Syntax: ARP.134 ATM CIP: A *msgType* message was *action* with a fixed header of [afn *addrFamily* pro *proType* snap *pSnap0 pSnap1 pSnap2 pSnap3 pSnap4* chksum *checksum* ext *extoff* ver *opVersion* shtl *addrTypeLen* sstl *subAddrTypeLen*] on net *network ID*.

Description: This is the fixed headr for a message

being sent or received by a MARS client.

ARP.135

Level: U-INFO

Short Syntax: ARP.135 ATM CIP ArpDisconnectLeaf: rsn= *reason_code*, cause= *cause_code*, diagLen= *diag_len*, diagData[0]= *diag_data* vpi= *vcc_vpi*, vci= *vcc_vci*, LeafAtmAddr= *leaf_remote_atm_address*

Long Syntax: ARP.135 ATM CIP ArpDisconnectLeaf: rsn= *reason_code*, cause= *cause_code*, diagLen= *diag_len*, diagData[0]= *diag_data* vpi= *vcc_vpi*, vci= *vcc_vci*, LeafAtmAddr= *leaf_remote_atm_address*

Description: The information in this message is the reason for which the leaf has been released. It also contains the channel vpi/vci for which this leaf was a member of along with the atm address of the leaf.

ARP.136

Level: UE-ERROR

Short Syntax: ARP.136 ATM CIP atmArpRcvFrame: Unknown *protype* value= *vauleNum* nt *network ID*

Long Syntax: ARP.136 ATM CIP atmArpRcvFrame: Unknown *protype* value= *vauleNum* nt *network ID*

Description: A packet with an unknown protocol ID has been received off of the specified network. This may or may not be expected traffic. In any event, the packet will be discarded. No forwarding will occur.

ARP.137

Level: U-INFO

Short Syntax: ARP.137 ATM CIP atmArpAddLeafAck: vpi= *vcc_vpi*, vci= *vcc_vci*, LeafAtmAddr= *leaf_remote_atm_address*

Long Syntax: ARP.137 ATM CIP atmArpAddLeafAck: vpi= *vcc_vpi*, vci= *vcc_vci*, LeafAtmAddr= *leaf_remote_atm_address*

Description: Confirms a successful addition of a new party to a point-to-multipoint call.

ARP.138

Level: U-TRACE

Short Syntax: ARP.138 ATM CIP: *msgType* spec info [spln *srcPln* tpln *targPln* pnum *numPairs* seq *priSeq* flags *msgFlags* cmi *clusterId* msn *marsSeq*].

Long Syntax: ARP.138 ATM CIP: *msgType* specific information [spln *srcPln* tpln *targPln* pnum *numPairs* seq *priSeq* flags *msgFlags* cmi *clusterId* msn *marsSeq*].

Description: This is the specific message content being sent or received by the MARS Client.

ARP.139

Level: U-INFO

Short Syntax: ARP.139 ATM CIP: atmaActivateServer Current Active Arp Server: AtmAddr= *vcc_remote_atm_address*

Long Syntax: ARP.139 ATM CIP: atmaActivateServer Current Active Arp Server: AtmAddr= *vcc_remote_atm_address*

Description: This is the ATM address of the current active Arp Server.

ARP.140

Level: U-INFO

Short Syntax: ARP.140 ATM CIP: start_SG failed: retcd= *retcd*, protid= *protid*, sgid= *sgid*, lsid= *lsid*, net= *net#*

Long Syntax: ARP.140 ATM CIP: start_SG failed: retcd= *retcd*, protid= *protid*, sgid= *sgid*, lsid= *lsid*, net= *net#*

Description: The starting of Server Group with SCSP failed. It could be that the Server group may be already started. Possible that the same Server Group Id is configured for different Subnets. The Server Group Id should be unique for each Subnet.

ARP.141

Level: U-INFO

Short Syntax: ARP.141 ATM CIP:AAL IE:Negotiating SDU sizes, Remote Station fwd max SDU sz (*fwd_max_SDU_size*)

Long Syntax: ARP.141 ATM CIP:AAL IE:Negotiating SDU sizes with Remote Station, Remote Station forward maximum SDU size (*fwd_max_SDU_size*)

Description: Remote Station Forward maximum SDU size is larger than our Backward maximum SDU size..Let us negotiate the SDU sizes with the Remote station..

ARP.142

Level: U-INFO

Short Syntax: ARP.142 ATM CIP:AAL IE:Negotiating SDU sizes, Remote Station bak max SDU sz (*bak_max_SDU_size*)

Long Syntax: ARP.142 ATM CIP:AAL IE:Negotiating SDU sizes with Remote Station, Remote Station backward maximum SDU size (*bak_max_SDU_size*)

Description: Remote Station Backward maximum SDU size is larger than our Forward maximum SDU size..Let us negotiate the SDU sizes with the Remote station..

ARP.143

Level: U-TRACE

Short Syntax: ARP.143 ATM CIP: No Chan *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.143 ATM CIP: Entry rcvd on Down Channel, prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: An incoming ATM ARP packet arrived on a channel that has gone down before the packet could be processed. The packet is discarded.

ARP.144

Level: UI-ERROR

Short Syntax: ARP.144 xmt: No mem for csa ent, prot *protocol_type* nt *network ID*

Long Syntax: ARP.144 transmit: No memory for csa record, protocol *protocol_type* net *network ID*

Description: During the output processing of an ARP packet, the router did not have memory available to inform SCSP through client state advertisement (csa) record for given ARP entry.

Cause: The router is extremely low on heap memory.

Action: Find someway to reduce memory usage.

ARP.145

Level: P_TRACE

Short Syntax: ARP.145 ATM CIP: InArp bad subnet *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.145 ATM CIP: InArp received from another subnet, prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: An InArp packet was received with a source protocol address that is not on one of the defined subnets on this interface. The packet is ignored.

ARP.146

Level: UI-ERROR

Short Syntax: ARP.146 ATM CIP: bad red call rcvd: Clnt prot/addr *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.146 ATM CIP: bad red call received: Client protocol/address *protocol_number/ protocol_address* net *network ID*

Description: A call was received to a local ATM address defined for CIP redundancy but was not answered for one of the following reasons: * the callee is configured as the one to place the call * the callee already has a redundancy channel * the caller ATM address does not match the configured caller ATM address. The ATM address of the caller follows in ARP_86. Check CIP redundancy configuration on this box and its partner.

ARP.147

Level: U-TRACE

Short Syntax: ARP.147 ATM CIP: ARP Pkt on rcv-only *protocol_number/ protocol_address* nt *network ID*

Long Syntax: ARP.147 ATM CIP: ARP Packet received on receive-only channel, prot/addr *protocol_number/ protocol_address* nt *network ID*

Description: An incoming ATM ARP packet arrived on a channel that is for reception of data only. The packet is discarded. Most likely cause is that the backward SDU size of this channel was smaller than the configured SDU for this interface.

Chapter 4. Advanced Peer-to-Peer (APPN)

This chapter describes Advanced Peer-to-Peer (APPN) messages. For information on message content and how to use the message, refer to the Introduction.

APPN.001

Level: C-INFO

Short Syntax: APPN.001 Rcvd netup for intf *n_net*

Long Syntax: APPN.001 Received netup for interface *n_net*

Description: This message is for each netup received per interface

APPN.002

Level: C-INFO

Short Syntax: APPN.002 Rcvd netdn for intf *n_net*

Long Syntax: APPN.002 Received netdown for interface *n_net*

Description: This message is for each netdown received per interface

APPN.003

Level: C-INFO

Short Syntax: APPN.003 Discarding APPN HPR pkt rcvd on dn intf.

Long Syntax: APPN.003 Discarding APPN HPR packet received on down interface.

Description: This message is for any packet sent on a port that is currently down

APPN.004

Level: C-INFO

Short Syntax: APPN.004 Unkwn Dialog Msge rcvd

Long Syntax: APPN.004 Unknown Dialog Message received

Description: When an unkown dialog message is recieved from EGPE

APPN.008

Level: C-INFO

Short Syntax: APPN.008 APPN dumped to file

Long Syntax: APPN.008 APPN dumped to file

Description: This message is when APPN takes a dump via talk 5

APPN.009

Level: C-INFO

Short Syntax: APPN.009 Stop APPN node

Long Syntax: APPN.009 Message has been sent to stop APPN node.

Description: This message is when APPN is told to stop via talk 5

APPN.010

Level: C-INFO

Short Syntax: APPN.010 APPN node not running

Long Syntax: APPN.010 APPN node is not running so cannot support talk 5 command.

Description: This message is when APPN is found to be gone when attempting a talk 5 cmd

APPN.011

Level: UE-ERROR

Short Syntax: APPN.011 *logged_string*

Long Syntax: APPN.011 *logged_string*

Description: This message is generated when an APPN subsystem generates a log entry. A log entry generally reports an error condition. See the logged text for more details.

Cause: An error occured in the APPN subsystem.

APPN.012

Level: UE-ERROR

Short Syntax: APPN.012 *logged_string*

Long Syntax: APPN.012 *logged_string*

Description: This message is generated when an APPN subsystem generates a log entry. A log entry generally reports an error condition. See the logged text for more details.

Cause: An error occured in the APPN subsystem.

APPN.013

Level: C-INFO

Short Syntax: APPN.013 APPN Msg: Comp:

component_name PrID: *Probe_ID* Op: *Operator_Name* Text: *message*

Long Syntax: APPN.013 APPN Messaeg: Component: *component_name* Probe ID: *Probe_ID* Operator: *Operator_Name*TextL *message*

Description: This message is generated when an APPN component wants to display a message to the user.

Cause: Any situation that warrants informing the user.

APPN.014

Level: P-TRACE

Short Syntax: APPN.014 *trace_info*

Long Syntax: APPN.014 *trace_info*

Description: When APPN's Data link control transmissions and receptions trace is enabled, this message displays XIDs and PIUs.

Cause: APPN traces an XID or PIU.

APPN.015

Level: P-TRACE

Short Syntax: APPN.015 *trace_info*

Long Syntax: APPN.015 *trace_info*

Description: When APPN Node-Level Traces are enabled, this message displays APPN node level traces.

Cause: An APPN Node-Level trace is generated.

APPN.016

Level: P-TRACE

Short Syntax: APPN.016 *trace_info*

Long Syntax: APPN.016 *trace_info*

Description: When APPN Component-level traces are enabled, this message displays APPN component-level traces.

Cause: An APPN Component-level trace is generated.

APPN.017

Level: ALWAYS

Short Syntax: APPN.017 *msg*

Long Syntax: APPN.017 *msg*

Description: Aping output.

Cause: A aping was issued from talk 5 with the -b option.

APPN.018

Level: UE-ERROR

Short Syntax: APPN.018 *msg*

Long Syntax: APPN.018 *msg*

Description: TN3270E Server NetDisp Advisor log

Cause: Error setting up the TN3270E Server NetDisp Advisor

APPN.023

Level: C-INFO

Short Syntax: APPN.023 DX *st1 st2 st3 st4*

Long Syntax: APPN.023 DX *st1 st2 st3 st4*

Description: Trace DLCX activation, deactivation, and error cases. No data trace.

APPN.024

Level: C-INFO

Short Syntax: APPN.024 *** *msg****

Long Syntax: APPN.024 *** *msg****

Description: This message is for general information from APPN CFG

APPN.025

Level: UE-ERROR

Short Syntax: APPN.025 *** *msg****

Long Syntax: APPN.025 *** *msg****

Description: This message is for error information from APPN CFG

APPN.026

Level: UE-ERROR

Short Syntax: APPN.026 *msg*

Long Syntax: APPN.026 *msg*

Description: This message is from the error log called from EGPE Elog will break the message up into 70 byte strings and passit in pieces to ELS

APPN.027

Level: C-INFO

Short Syntax: APPN.027 APPN *msg*

Long Syntax: APPN.027 EGPE/APPN node process was *msg* second.

Description: Indicates EGPE's MOS scheduler stopped or restarted the node, with time of day

APPN.028

Level: ALWAYS

Short Syntax: APPN.028 APPN *msg*

Long Syntax: APPN.028 APPN *msg*

Description: Indicates a critical event in APPN operation (like APPN abend dump)

APPN.029

Level: CE-ERROR

Short Syntax: APPN.029 *logged_string*

Long Syntax: APPN.029 *logged_string*

Description: This message is generated when an APPN subsystem generates an informational log entry. A log entry generally reports an error condition. See the logged text for more details.

Cause: An error occured in the APPN subsystem.

APPN.030

Level: CE-ERROR

Short Syntax: APPN.030 *logged_string*

Long Syntax: APPN.030 *logged_string*

Description: This message is generated when an APPN subsystem generates a warning log entry. A log entry generally reports an error condition. See the logged text for more details.

Cause: An error occured in the APPN subsystem.

APPN.031

Level: UE-ERROR

Short Syntax: APPN.031 *logged_string*

Long Syntax: APPN.031 *logged_string*

Description: This message is generated when an APPN subsystem generates a error log entry. A log entry generally reports an error condition. See the logged text for more details.

Cause: An error occured in the APPN subsystem.

APPN.032

Level: CI-ERROR

Short Syntax: APPN.032 *logged_string*

Long Syntax: APPN.032 *logged_string*

Description: This message is generated when an APPN subsystem generates a critical log entry. A log entry generally reports an error condition. See the logged text for more details.

Cause: An error occured in the APPN subsystem.

APPN.033

Level: UI-ERROR

Short Syntax: APPN.033 *logged_string*

Long Syntax: APPN.033 *logged_string*

Description: This message is generated when an APPN subsystem generates an action required log entry. A log entry generally reports an error condition. See the logged text for more details.

Cause: An error occured in the APPN subsystem.

Chapter 5. AppleTalk Phase 2 (AP2)

This chapter describes AppleTalk Phase 2 (AP2) messages. For information on message content and how to use the message, refer to the Introduction.

AP2.003

Level: P-TRACE

Short Syntax: AP2.003 q ovf *src_net*/ *src_node* -> *dest_net*/ *dest_node* nt *network*

Long Syntax: AP2.003 queue overflow *src_net*/ *src_node* -> *dest_net*/ *dest_node* net *network*

Description: The specified packet caused the forwarder input queue to overflow and was discarded.

AP2.005

Level: UE-ERROR

Short Syntax: AP2.005 pkt trnc *length* pkt ln *received_length src_net*/ *src_node* -> *dst_net*/ *dst_node*

Long Syntax: AP2.005 packet truncated *length* packet length *received_length src_net*/ *src_node* -> *dst_net*/ *dst_node*

Description: The physical length of the packet as received was not long enough to contain a packet of the length claimed by the DDP header. Both lengths include only the DDP header and data, and do not include the LAP header of data-link header.

AP2.007

Level: UE-ERROR

Short Syntax: AP2.007 bd hdr cksm frm *src_net*/ *src_node*, rcv *rcvd_csum*, comp *comp_csum*

Long Syntax: AP2.007 bad header checksum from *src_net*/ *src_node*, received *rcvd_csum*, computed *comp_csum*

Description: The computed checksum of the specified packet did not match the checksum value in the DDP header.

AP2.008

Level: U-INFO

Short Syntax: AP2.008 no rte *src_net*/ *src_node* -> *dest_net*/ *dest_node*

Long Syntax: AP2.008 no route *src_net*/ *src_node* -> *dest_net*/ *dest_node*

Description: No routing table entry was found for the destination net while trying to route the specified packet.

AP2.009

Level: UE-ERROR

Short Syntax: AP2.009 hp cnt ovf *src_net*/ *src_node* -> *dest_net*/ *dest_node*

Long Syntax: AP2.009 hop count overflow *src_net*/ *src_node* -> *dest_net*/ *dest_node*

Description: The specified packet was discarded while attempting forwarding due to overflow of the packet hop count.

Cause: Packets whose hop counts overflow are typically victims of a routing loop. This is usually a temporary condition.

Action: If the problem is excessive or persistent then check for improper network configuration.

AP2.010

Level: UI-ERROR

Short Syntax: AP2.010 no iorb for copy

Long Syntax: AP2.010 no i/o request block to copy packet

Description: The system was making a copy of a directed broadcast packet for internal processing of the packet, and was unable to allocate a system buffer to copy the packet. The packet will still be forwarded, but no local copy will be received.

Cause: There is a buffer shortage in the router. This may be a temporary condition.

AP2.011

Level: UI-ERROR

Short Syntax: AP2.011 No RTMP entry for FwdReq pkt to nt *dest_net*, rcvd nt *network*

Long Syntax: AP2.011 No RTMP entry for FwdReq pkt to net *dest_net*, received net *network*

Description: An Apple NBP Forward request packet was received and either RTMP has no entry for the network or the net is no longer directly connected.

AP2.012

Level: P-TRACE

Short Syntax: AP2.012 *src_net*/ *src_node* -> *dest_net*/ *dest_node*

Long Syntax: AP2.012 *src_net*/ *src_node* -> *dest_net*/ *dest_node*

Description: The specified AppleTalk packet was forwarded.

AP2.013

Level: UI-ERROR

Short Syntax: AP2.013 pkt too lg *pkt_len* > *max_len* nt *network src_net*/ *src_node* -> *dest_net*/ *dest_node*

Long Syntax: AP2.013 packet too large *pkt_len* > *max_len* net *network src_net*/ *src_node* -> *dest_net*/ *dest_node*

Description: A packet exceeded the maximum length of a packet on the outgoing network and was discarded.

AP2.014

Level: UI-ERROR

Short Syntax: AP2.014 pkt *src_net*/ *src_node* -> *dest_net*/ *dest_node* dsc, rsn *code*

Long Syntax: AP2.014 packet *src_net*/ *src_node* -> *dest_net*/ *dest_node* discarded, reason *code*

Description: An outgoing packet was not successfully transmitted for the reason indicated by the error code.

AP2.017

Level: UE-ERROR

Short Syntax: AP2.017 bad dst skt *socket*

Long Syntax: AP2.017 bad destination socket *socket*

Description: A locally destined packet had a destination socket on which there was no listener.

AP2.018

Level: UE-ERROR

Short Syntax: AP2.018 unk prt tp *type*

Long Syntax: AP2.018 unkown protocol type *type*

Description: A locally destined packet had an unrecognized value in the protocol type field.

AP2.019

Level: UE-ERROR

Short Syntax: AP2.019 no uniq nd addr avial nt *network*

Long Syntax: AP2.019 no unique node address available net *network*

Description: The handler was unable to find a unique node address available on this network.

Cause: There already exist the maximum number of nodes on the network; all node numbers are taken. The net range should be extended.

AP2.020

Level: C-INFO

Short Syntax: AP2.020 nt/nd addr assgnd *net_number*/ *node_number* nt *network*

Long Syntax: AP2.020 net/node address assigned *net_number*/ *node_number* net *network*

Description: The indicated net / node address has been assigned to the specified interface.

AP2.021

Level: C-INFO

Short Syntax: AP2.021 intfc up *net_num*/ *node_num* nt *network*

Long Syntax: AP2.021 interface up *net_num*/ *node_num* net *network*

Description: The specified interface has secured both a net and node address, and is now up and looking for a zone name.

AP2.022

Level: C-INFO

Short Syntax: AP2.022 intfc up *net_num*/ *node_num* zn *zone_name* nt *network*

Long Syntax: AP2.022 interface up *net_num*/ *node_num* zone *zone_name* net *network*

Description: The specified interface has secured a net, node and zone name, and is now up.

AP2.027

Level: UI-ERROR

Short Syntax: AP2.027 no mem for NBP pkt

Long Syntax: AP2.027 no memory for NBP packet

Description: An iorb was not available for sending an NBP packet.

AP2.028

Level: UI-ERROR

Short Syntax: AP2.028 NBP *type* disc nt *network* rsn *error_code*

Long Syntax: AP2.028 NBP *type* discarded net *network* reason *error_code*

Description: An NBP packet was not sent for the indicated reason.

AP2.029

Level: P-TRACE

Short Syntax: AP2.029 NBP *type* snt to net *net_number*

Long Syntax: AP2.029 NBP *type* sent to net *net_number*

Description: An NBP packet was sent to the indicated net.

AP2.031

Level: UI-ERROR

Short Syntax: AP2.031 no mem for AARP Probe

Long Syntax: AP2.031 no memory for AARP Probe

Description: A buffer was not available for an AARP Probe packet.

AP2.032

Level: UI-ERROR

Short Syntax: AP2.032 AARP Probe disc nt *network* rsn *error_code*

Long Syntax: AP2.032 AARP Probe discarded net *network* reason *error_code*

Description: An Apple ARP Probe was not sent for the indicated reason.

AP2.033

Level: P-TRACE

Short Syntax: AP2.033 AARP Probe snt nt *network*

Long Syntax: AP2.033 AARP Probe sent net *network*

Description: An Apple ARP Probe was sent on the indicated net.

AP2.034

Level: C-INFO

Short Syntax: AP2.034 AARP Rsps match tentative addr, new addr selected nt *network*

Long Syntax: AP2.034 AARP Response match tentative addr, new addr selected nt *network*

Description: An Apple ARP Response was received in response to our probe claiming the tentative address. A new node address was selected for continued probing.

AP2.035

Level: UE-ERROR

Short Syntax: AP2.035 Unrec AARP pkt typ *arp_type* rcvd nt *network*

Long Syntax: AP2.035 Unrecognized AARP packet type *arp_type* received net *network*

Description: An Apple ARP packet with an unrecognized type was received.

AP2.036

Level: P-TRACE

Short Syntax: AP2.036 AARP Probe rcvd *src_net*/ *src_node* nt *network*

Long Syntax: AP2.036 AARP Probe received *src_net*/ *src_node* net *network*

Description: An Apple ARP Probe packet was received.

AP2.037

Level: UI-ERROR

Short Syntax: AP2.037 AARP Response disc nt *network* rsn *error_code*

Long Syntax: AP2.037 AARP Response discarded net *network* reason *error_code*

Description: An Apple ARP Response was not sent for the indicated reason.

AP2.038

Level: P-TRACE

Short Syntax: AP2.038 AARP Response snt nt *network*

Long Syntax: AP2.038 AARP Response sent net *network*

Description: An Apple ARP Response to a probe was sent on the indicated net.

AP2.039

Level: UE-ERROR

Short Syntax: AP2.039 Echo pkt short (*length*) frm *src_net*/ *src_node* nt *network*

Long Syntax: AP2.039 Echo packet too short (*length* bytes) from *src_net*/ *src_node* net *network*

Description: An Echo packet was received that was too short to contain the echo packet header.

AP2.040

Level: U-TRACE

Short Syntax: AP2.040 Echo pkt, func *function_code*, frm *src_net*/ *src_node* nt *network*

Long Syntax: AP2.040 Echo packet, echo function *function_code*, received from *src_net*/ *src_node* net *network*

Description: An Echo Protocol packet, which was not an Echo Request or Echo Reply was received from the specified node. It will not be answered.

AP2.041

Level: P-TRACE

Short Syntax: AP2.041 Echo Req frm *src_net*/ *src_node* nt *network*, rplyng

Long Syntax: AP2.041 Echo Request from *src_net*/ *src_node* net *network*, replying

Description: An Echo Request packet was received from the specified host. A reply will be sent.

AP2.045

Level: UI-ERROR

Short Syntax: AP2.045 Echo Rply disc nt *network* rsn *error_code*

Long Syntax: AP2.045 Echo Reply discarded net *network* reason *error_code*

Description: An Echo Reply was not sent for the indicated reason.

AP2.047

Level: UE-ERROR

Short Syntax: AP2.047 pkt too short (*length*) net *network*

Long Syntax: AP2.047 Long DDP packet too short for header (*length* bytes) net *network*

Description: A long format DDP packet has been received that is shorter than the length of a long DDP header (13 bytes).

AP2.048

Level: UE-ERROR

Short Syntax: AP2.048 pkt too long (*length*) *src_net*/ *src_node* -> *dst_net*/ *dst_node*

Long Syntax: AP2.048 Long DDP packet too long (*length* bytes) *src_net*/ *src_node* -> *dst_net*/ *dst_node*

Description: A long format DDP packet has been received with more than the limit of 586 bytes of data after the DDP header.

AP2.049

Level: UE-ERROR

Short Syntax: AP2.049 DDP rsvd bits *src_net*/ *src_node* -> *dst_net*/ *dst_node*

Long Syntax: AP2.049 Long DDP packet reserved bit(s) set *src_net*/ *src_node* -> *dst_net*/ *dst_node*

Description: A long format DDP packet has been received with one (or more) of the two reserved bits above the hop count set.

AP2.056

Level: P-TRACE

Short Syntax: AP2.056 *source_net*/ *source_node* -> *destination_net*/ *destination_node* nt *network* ign

Long Syntax: AP2.056 *source_net*/ *source_node* -> *destination_net*/ *destination_node* net *network* ignored

Description: An AppleTalk packet was recognized but ignored because AppleTalk forwarding was not enabled on the interface.

AP2.059

Level: UI-ERROR

Short Syntax: AP2.059 Ilg zone *zone_name* seed w/o net seed nt *network*

Long Syntax: AP2.059 Illegal zone *zone_name* seed without network seed net *network*

Description: The user configured a zone name for a network in which no network number was configured. The zone name will be ignored.

AP2.060

Level: UE-ERROR

Short Syntax: AP2.060 NBP bd cnt *tuple_count* in *type* frm *src_net*/ *src_node* nt *network*

Long Syntax: AP2.060 NBP bad count *tuple_count* in *type* from *src_net*/ *src_node* net *network*

Description: The NBP Request packet from the specified host contained an illegal tuple count not equal to 1.

AP2.061

Level: P-TRACE

Short Syntax: AP2.061 NBP *type* rcvd frm *src_net*/ *src_node* nt *network*

Long Syntax: AP2.061 NBP *type* received from *src_net*/ *src_node* net *network*

Description: An NBP Broadcast Request or Forward Request was received from the specified host.

AP2.062

Level: U-INFO

Short Syntax: AP2.062 no knwn zn nm for nt *net_num* in NBP BrRq frm *src_net*/ *src_node*

Long Syntax: AP2.062 no known zone name for net *net_num* in NBP BrRq from *src_net*/ *src_node*

Description: An associated zone name for the requested net in a BrRq packet was not found.

AP2.063

Level: U-INFO

Short Syntax: AP2.063 zn *zone_name* not fnd in ZIT, NBP BrRq frm *src_net*/ *src_node*

Long Syntax: AP2.063 zone *zone_name* not found in ZIT, NBP BrRq from *src_net*/ *src_node*

Description: The requested zone in BrRq from the specified host was not found in the Zone Information Table.

AP2.064

Level: UI-ERROR

Short Syntax: AP2.064 no mem for NBP stat block, BrRq frm *src_net*/ *src_node* ign

Long Syntax: AP2.064 no memory for NBP status block, BrRq from *src_net*/ *src_node* ign

Description: No memory was available for status block to process NBP BrRq from the indicated host.

AP2.065

Level: UE-ERROR

Short Syntax: AP2.065 NBP shrt (*length*) frm *src_net*/ *src_node* nt *network*

Long Syntax: AP2.065 NBP short (*length* bytes) from *src_net*/ *src_node* nt *network*

Description: An NBP packet was received that is too short to contain the NBP header. The packet will be discarded.

AP2.066

Level: UE-ERROR

Short Syntax: AP2.066 NBP bd func *function* frm *src_net*/ *src_node* nt *network*

Long Syntax: AP2.066 NBP bad function *function* from *src_net*/ *src_node* nt *network*

Description: An NBP packet was received with an unsupported function code. The packet will be discarded.

AP2.067

Level: UE-ERROR

Short Syntax: AP2.067 NBP trnc (*length*) frm *src_net*/ *src_node* nt *network*

Long Syntax: AP2.067 NBP truncated (*length* bytes) from *src_net*/ *src_node* nt *network*

Description: An NBP packet was received that is too short to contain the NBP data. The packet will be discarded.

AP2.068

Level: UE-ERROR

Short Syntax: AP2.068 NBP *type* ilg *field* len *length* frm *src_net*/ *src_node* nt *network*

Long Syntax: AP2.068 NBP *type* ilg *field* len *length* from *src_net*/ *src_node* nt *network*

Description: An NBP packet was received that has an entity name more than 32 characters long. The packet will be discarded.

AP2.069

Level: P-TRACE

Short Syntax: AP2.069 NBP *type* snt to net *net_number* node *node_number*

Long Syntax: AP2.069 NBP *type* sent to net *net_number* node *node_number*

Description: An NBP packet was sent to the indicated destination.

AP2.070

Level: P-TRACE

Short Syntax: AP2.070 NBP LkUp rcvd frm *src_net*/ *src_node* nt *network*

Long Syntax: AP2.070 NBP LookUp received from *src_net*/ *src_node* net *network*

Description: An NBP LookUp Request was received from the specified host.

Chapter 6. AppleTalk Phase 2 Routing Table Maintenance Protocol (R2MP)

This chapter describes AppleTalk Phase 2 Routing Table Maintenance Protocol (R2MP) messages. For information on message content and how to use the message, refer to the Introduction.

R2MP.003

Level: U-INFO

Short Syntax: R2MP.003 nt num inferred *net_number* nt *network*

Long Syntax: R2MP.003 net number inferred *net_number* net *network*

Description: A net number has been inferred from an RTMP data packet and has been assigned to the specified interface.

R2MP.004

Level: UE-ERROR

Short Syntax: R2MP.004 nt nmbrs cnflct frm *net_num/ src_node* not in *net_num- net_num* on nt *network*

Long Syntax: R2MP.004 net numbers conflict from *net_num/ src_node* not in *net_num- net_num* on nt *network*

Description: The source net number of an RTMP packet conflicts with the current known net range for the specified interface.

Cause: Configuration error in some host on the network.

Action: Make sure that only one network range is being seeded by multiple routers on the same network.

R2MP.005

Level: UE-ERROR

Short Syntax: R2MP.005 bd net *net_range* in RTMP frm *src_net/ src_node*

Long Syntax: R2MP.005 bad net *net_range* in RTMP from *src_net/ src_node*

Description: An illegal network range was found in an RTMP data packet from the specified router.

R2MP.006

Level: UI-ERROR

Short Syntax: R2MP.006 nt rtng tbl ovrfl, dsc *net_range*

Long Syntax: R2MP.006 network routing table overflow, discarding *net_range*

Description: Insertion of the specified net into the routing table was not performed because the allocation of heap memory failed.

Action: If the problem is chronic, increase the heap memory available by: (1) upgrading memory, or (2) turning off unnecessary features. You may be able to reduce the size of AppleTalk tables using AppleTalk filters to filter out unnecessary routing information.

R2MP.007

Level: U-INFO

Short Syntax: R2MP.007 rte to *net_range* via *gw_net/ gw_node* excds max hps, disc

Long Syntax: R2MP.007 rte to *net_range* via *gw_net/ gw_node* exceeds max hops, discarded

Description: An RTMP data packet contained a new route to the specified net, but at too large a hop count. The route was discarded.

R2MP.008

Level: U-INFO

Short Syntax: R2MP.008 new rte to *net_range* via *gw_net/ gw_node*, hops *hops*

Long Syntax: R2MP.008 new route to *net_range* via *gw_net/ gw_node*, hops *hops*

Description: A new route was added to the routing table via the indicated first hop.

R2MP.009

Level: U-INFO

Short Syntax: R2MP.009 rte to *net_range* via *gw_net/ gw_node* dltd, hopc excded

Long Syntax: R2MP.009 rte to *net_range* via *gw_net/ gw_node* deleted, hopcount exceeded

Description: The route to the indicated network was deleted from the routing table due to a new route with too large a hop count.

R2MP.010

Level: U-INFO

Short Syntax: R2MP.010 rte to *net_range* aged away

Long Syntax: R2MP.010 rte to *net_range* aged away

Description: The route to the indicated network was deleted from the routing table due to aging.

R2MP.011

Level: UI-ERROR

Short Syntax: R2MP.011 no mem RTMP brdcst nt *network*, *packet_count* pkts snt

Long Syntax: R2MP.011 no memory for RTMP broadcast net *network*, *packet_count* packets sent

Description: No memory was available for a buffer to send an RTMP data packet. The reported number of packets was sent before the error occurred.

R2MP.012

Level: UI-ERROR

Short Syntax: R2MP.012 Outgng disc nt *network* rsn *error_code*

Long Syntax: R2MP.012 Outgoing discarded net *network* reason *error_code*

Description: An outgoing RTMP packet was not successfully transmitted for the specified reason.

R2MP.014

Level: P-TRACE

Short Syntax: R2MP.014 rqst rcv frm *src_net*/ *src_node* nt *network*

Long Syntax: R2MP.014 Request received from *src_net*/ *src_node* net *network*

Description: An RTMP Request was received from the specified host. An RTMP Response will be sent.

R2MP.017

Level: P-TRACE

Short Syntax: R2MP.017 Snt nt *network* pkts *packet_count*

Long Syntax: R2MP.017 Sent net *network* packets *packet_count*

Description: The indicated number of RTMP data packets was sent on the specified interface.

R2MP.019

Level: U-INFO

Short Syntax: R2MP.019 del nt *net_range* rt via *net_num*/ *node_num* nt *network*

Long Syntax: R2MP.019 del network *net_range* route via *net_num*/ *node_num* net *network*

Description: The route to the indicated network has been deleted from the routing table.

R2MP.023

Level: UE-ERROR

Short Syntax: R2MP.023 Dta bd len (*length*) frm *src_net*/ *src_node* nt *network*

Long Syntax: R2MP.023 Data bad length (*length* bytes) from *src_net*/ *src_node* net *network*

Description: The RTMP Data or Response packet did not have an even (or zero) number of RTMP routing tuples. The packet will be discarded.

R2MP.024

Level: UE-ERROR

Short Syntax: R2MP.024 Dta bd ID len (*ID_length*) frm *src_net*/ *src_node* nt *network*

Long Syntax: R2MP.024 Data bad sender's node ID length (*ID_length* bits) from *src_net*/ *src_node* net *network*

Description: A RTMP Data or Repsonse packet was received where the Sender's ID length was not 8 bits. This implementation requires this to be 8 bits. The packet will be discarded.

R2MP.026

Level: UE-ERROR

Short Syntax: R2MP.026 Dta bd vers (*version*) frm *src_net*/ *src_node* nt *network*

Long Syntax: R2MP.026 Data bad version (*version*) from *src_net*/ *src_node* net *network*

Description: The RTMP Data or Response packet did not have the correct version number (0x82) in the first RTMP routing tuple. The packet will be discarded.

R2MP.027

Level: P-TRACE

Short Syntax: R2MP.027 RDR rcv frm *src_net*/ *src_node* nt *network*

Long Syntax: R2MP.027 Route Data Request received from *src_net*/ *src_node* net *network*

Description: A RTMP Route Data Request or

Extended Route Data Request was received from the specified host. RTMP Data will be sent.

R2MP.028

Level: UE-ERROR

Short Syntax: R2MP.028 bad netrange *net_first- net_last* nt *network* spans *net_first- net_last*

Long Syntax: R2MP.028 Bad netrange *net_first- net_last* net *network* spans *net_first- net_last*

Description: A netrange overlaps either an interface netrange or an existing net. The first netrange will be discarded.

Cause: Bad network configuration.

R2MP.029

Level: UI-ERROR

Short Syntax: R2MP.029 filtered int netrange *net_first-net_last* nt *network*

Long Syntax: R2MP.029 Filtered Interface netrange *net_first- net_last* net *network*

Description: An interface netrange is filtered by its own net filter. The interface will be disabled. The user should reconfigure either the filter or the interface netrange.

R2MP.030

Level: UE-ERROR

Short Syntax: R2MP.030 filtered net *net* on nt *network*

Long Syntax: R2MP.030 Filtered net *net* on net *network*

Description: A net was filtered by an interface net filter.

R2MP.031

Level: UE-ERROR

Short Syntax: R2MP.031 filtered netrange *net_first-net_last* frm *src_net/ src_node* on nt *network*

Long Syntax: R2MP.031 Filtered netrange *net_first-net_last* from *src_net/ src_node* on net *network*

Description: A netrange from another router was filtered by an interface net filter.

R2MP.032

Level: CE-ERROR

Short Syntax: R2MP.032 Req frm *src_net/ src_node* nt *network*, port ntwk num 0

Long Syntax: R2MP.032 Request from *src_net/ src_node* net *network*, port's network number 0

Description: A RTMP Request or Route Data Request packet was received on an interface whose port network number was still zero. The request will be ignored.

Cause: Port has not yet gleaned network number from seed router.

Action: Wait until network number gleaned.

Cause: No seed router on network for network number.

Action: Reconfigure a router to be seed.

R2MP.033

Level: P-TRACE

Short Syntax: R2MP.033 data pkt frm *src_net/ src_node* nt *network*

Long Syntax: R2MP.033 data packet from *src_net/ src_node* net *network*

Description: A RTMP data packet has been received.

R2MP.034

Level: UE-ERROR

Short Syntax: R2MP.034 rqst, bd src node *src_net/ src_node* nt *network*

Long Syntax: R2MP.034 Request, bad source node *src_net/ src_node* net *network*

Description: A RTMP Request or Route Data Request was received with an illegal source address (0 or 255).

R2MP.035

Level: UE-ERROR

Short Syntax: R2MP.035 rqst, unk func *R2MP_function* frm *src_net/ src_node* nt *network*, disc

Long Syntax: R2MP.035 Request, unkown function *R2MP_function* from *src_net/ src_node* net *network*

Description: A RTMP Request was received with an unknown function code. The packet will be ignored.

R2MP.036

Level: UE-ERROR

Short Syntax: R2MP.036 Rqst short (*length*) frm *src_net/ src_node* nt *network*

Long Syntax: R2MP.036 Request too short (*length* bytes) from *src_net/ src_node* net *network*

Description: The RTMP request packet was too short to contain the required RTMP header data. The packet will be discarded.

R2MP.037

Level: UE-ERROR

Short Syntax: R2MP.037 Dta short (*length*) frm *src_net*/ *src_node* nt *network*

Long Syntax: R2MP.037 Data packet short (*length* bytes) from *src_net*/ *src_node* net *network*

Description: The RTMP Data or Response packet was too short to contain the required RTMP header data. The packet will be discarded.

R2MP.038

Level: UE-ERROR

Short Syntax: R2MP.038 ilg rtmp net 0 from *src_net*/ *src_node* nt *network*

Long Syntax: R2MP.038 illegal rtmp net number 0 from *src_net*/ *src_node* net *network*

Description: A RTMP Data or Response packet with a sender's network number of 0 was received. The packet will be discarded.

Cause: Sending node has software bug, should not send RTMP Data or Response when network number is zero.

Chapter 7. AppleTalk Phase 2 Zone Information Protocol (ZIP2)

This chapter describes AppleTalk Phase 2 Zone Information Protocol (ZIP2) messages. For information on message content and how to use the message, refer to the Introduction.

ZIP2.001

Level: U-INFO

Short Syntax: ZIP2.001 del zone *zone*

Long Syntax: ZIP2.001 deleting zone *zone*

Description: The indicated zone was deleted from the Zone Information Table.

ZIP2.002

Level: UI-ERROR

Short Syntax: ZIP2.002 no mem for new zone *zone*

Long Syntax: ZIP2.002 no memory for new zone *zone*

Description: The indicated zone was not inserted into the Zone Information Table due to insufficient memory in the router.

ZIP2.003

Level: UI-ERROR

Short Syntax: ZIP2.003 no mem for ZIP query net *net_number*

Long Syntax: ZIP2.003 no memory for ZIP query net *net_number*

Description: The router was unable to generate a zone name query for the indicated network because no memory was available for the outgoing packet.

ZIP2.004

Level: UI-ERROR

Short Syntax: ZIP2.004 query disc nt *network* rsn *error_code*

Long Syntax: ZIP2.004 query discarded net *network* reason *error_code*

Description: A zone name query was not transmitted on the indicated net for the specified reason.

ZIP2.006

Level: C-INFO

Short Syntax: ZIP2.006 query for *net_num* brdcst nt *network*

Long Syntax: ZIP2.006 query for *net_num* broadcast on net *network*

Description: A ZIP query was sent for the indicated net was broadcast on the specified interface.

ZIP2.008

Level: P-TRACE

Short Syntax: ZIP2.008 rply rcvd frm *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.008 reply received from *src_net*/ *src_node* net *network*

Description: A ZIP reply packet was received from the indicated router.

ZIP2.009

Level: C-INFO

Short Syntax: ZIP2.009 ZIT entry, zn nm *zone* assgnd to nt *net_number*

Long Syntax: ZIP2.009 ZIT entry, zone name *zone* assigned to net *net_number*

Description: The specified zone name for the indicated net was added to the Zone Information Table.

ZIP2.011

Level: UI-ERROR

Short Syntax: ZIP2.011 rply disc nt *network* rsn *error_code*

Long Syntax: ZIP2.011 reply discarded net *network* reason *error_code*

Description: A ZIP reply was not sent for the indicated reason.

ZIP2.013

Level: P-TRACE

Short Syntax: ZIP2.013 qry rcvd frm *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.013 query received from *src_net*/ *src_node* net *network*

Description: A ZIP query packet was received from the indicated node.

ZIP2.014

Level: UE-ERROR

Short Syntax: ZIP2.014 Bad GtNtInf rq frm *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.014 Bad GetNetInfo request from *src_net*/ *src_node* net *network*

Description: A ZIP GetNetInfo request was discarded due to either a short packet length or non-blank fields.

Cause: The remote node has a programming error.

ZIP2.015

Level: U-INFO

Short Syntax: ZIP2.015 GtNtInf rqst frm *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.015 GetNetInfo request from *src_net*/ *src_node* net *network*.

Description: A ZIP GetNetInfo request was received from the indicated source.

ZIP2.016

Level: UI-ERROR

Short Syntax: ZIP2.016 no buf for ZIP GtNtInf rply to *src_net*/ *src_node*

Long Syntax: ZIP2.016 no packet buffer for ZIP GetNetInfo reply to *src_net*/ *src_node*.

Description: No packet buffer was available for sending a ZIP GetNetInfo reply to the specified source.

ZIP2.017

Level: UE-ERROR

Short Syntax: ZIP2.017 rply trunc frm *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.017 reply truncated from *src_net*/ *src_node* net *network*

Description: A ZIP reply was received that was not long enough to contain all of the ZIP tuples. All tuples before the DDP end of the packet will be processed.

ZIP2.018

Level: UI-ERROR

Short Syntax: ZIP2.018 GtNtInf rply disc nt *network* rsn *error_code*

Long Syntax: ZIP2.018 GetNetInfo reply discarded net *network* reason *error_code*

Description: A ZIP GetNetInfo reply was not sent for the indicated reason.

ZIP2.019

Level: U-INFO

Short Syntax: ZIP2.019 GtNtInf rply for *net_range* frm *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.019 GetNetInfo reply for net *net_range* from *src_net*/ *src_node* net *network*

Description: A GetNetInfo reply was received for the given net range from the indicated source over the indicated net.

ZIP2.020

Level: UE-ERROR

Short Syntax: ZIP2.020 GtNtInf rply trunc (*length*) frm *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.020 GetNetInfo reply truncated (*length* bytes) from *src_net*/ *src_node* net *network*

Description: A GetNetInfo reply was received with the packet too short to hold all the information.

Cause: The remote node has a programming error.

ZIP2.021

Level: U-INFO

Short Syntax: ZIP2.021 Ntfy frm *src_net*/ *src_node* nt *network*, ign

Long Syntax: ZIP2.021 ZIP Notify from *src_net*/ *src_node* net *network*, ignored

Description: A ZIP Notify was received, these are currently ignored.

ZIP2.023

Level: UE-ERROR

Short Syntax: ZIP2.023 ATP shrt (*length*) frm *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.023 ATP short (*length* bytes) from *src_net*/ *src_node* net *network*

Description: An ATP packet was received that was too short to contain the ATP header. The packet will be discarded.

ZIP2.024

Level: P-TRACE

Short Syntax: ZIP2.024 *type* rcvd frm *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.024 *type* received from *src_net*/ *src_node* net *network*

Description: A ZIP GetMyZone, GetZoneList, or GetLocalZones ATP packet was received from the indicated host.

ZIP2.025

Level: UE-ERROR

Short Syntax: ZIP2.025 ATP bd hdr frm *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.025 ATP bad header from *src_net*/ *src_node* net *network*

Description: Bad ATP header from specified host. TReq not XO, or low bit of Bitmap not set. The packet will be discarded.

ZIP2.026

Level: UE-ERROR

Short Syntax: ZIP2.026 ATP bd func *function* frm *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.026 ATP bd function *function* from *src_net*/ *src_node* net *network*

Description: A ZIP ATP packet was received with a bad function code in the ATP user bytes. The packet will be discarded.

ZIP2.027

Level: UE-ERROR

Short Syntax: ZIP2.027 *type* too long (*length*) frm *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.027 *type* too long (*length* bytes) from *src_net*/ *src_node* net *network*

Description: A ZIP GetMyZone or GetZoneList ATP request packet was too long.

ZIP2.028

Level: UE-ERROR

Short Syntax: ZIP2.028 GetZoneList strt indx 0 frm *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.028 GetZoneList start index 0 from *src_net*/ *src_node* net *network*

Description: An ZIP GetZoneList or GetLocalZones packet was received with a start index of 0.

ZIP2.029

Level: UE-ERROR

Short Syntax: ZIP2.029 GetMyZone strt indx not 0 frm *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.029 GetMyZone start index not 0 from *src_net*/ *src_node* net *network*

Description: A GetMyZone ATP packet was received where the start index was not 0. The packet will be discarded.

ZIP2.030

Level: U-INFO

Short Syntax: ZIP2.030 No zn nm assoc wth nt *network*

Long Syntax: ZIP2.030 No zone name associated with net *network*

Description: There is no zone name associated with the indicated directly connected network.

Cause: This is a temporary condition where the router has received a ZIP GetMyZone packet before it has learned the zone name of the network for this interface.

ZIP2.031

Level: UI-ERROR

Short Syntax: ZIP2.031 *type*Reply disc nt *network* rsn *error_code*

Long Syntax: ZIP2.031 *type*Reply discarded net *network* reason *error_code*

Description: A ZIP GetZoneList, GetMyZone or GetLocalZones Reply was not sent for the indicated reason.

ZIP2.032

Level: UE-ERROR

Short Syntax: ZIP2.032 Ntfy trunc (*length*) frm *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.032 Notify truncated (*length* bytes) from *src_net*/ *src_node* net *network*

Description: A ZIP Notify packet was received that was not long enough to contain the claimed zone name length.

ZIP2.033

Level: UE-ERROR

Short Syntax: ZIP2.033 *type* usr byt 2 not 0 frm *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.033 *type* user byte 2 not 0 from *src_net*/ *src_node* net *network*

Description: A ZIP GetMyZone, GetZoneList or GetLocalZones ATP packet was received with user byte 2 of the ATP header not 0 from the indicated host. The packet will be discarded.

ZIP2.034

Level: UE-ERROR

Short Syntax: ZIP2.034 GetZoneList st indx *index,* high frm *src_net/ src_node* nt *network*

Long Syntax: ZIP2.034 GetZoneList start index *index,* too high from *src_net/ src_node* net *network*

Description: A ZIP GetZoneList or GetLocalZones packet was received asking for zones with indices above the one given, but none were found.

Cause: A change in the ZIT, such as a zone deletion, has caused the indices to change values since the last GetZoneList request.

Action: Try again.

Cause: The remote node has a programing error.

ZIP2.035

Level: CE-ERROR

Short Syntax: ZIP2.035 query cnt 0 frm *src_net/ src_node* nt *network*

Long Syntax: ZIP2.035 query count 0 from *src_net/ src_node* net *network*

Description: A ZIP Query packet was received with a network count of 0.

ZIP2.036

Level: CE-ERROR

Short Syntax: ZIP2.036 rply cnt 0 frm *src_net/ src_node* nt *network*

Long Syntax: ZIP2.036 reply count 0 from *src_net/ src_node* net *network*

Description: A ZIP Reply packet was received with a network count of 0.

ZIP2.038

Level: UE-ERROR

Short Syntax: ZIP2.038 cnt *network_count* & len (*length*) disag frm *src_net/ src_node* nt *network*

Long Syntax: ZIP2.038 Network count *network_count* and DDP length (*length* bytes) disagree from *src_net/ src_node* net *network*

Description: A ZIP Query packet was received where the expected length based on the ZIP network count does not agree with the actual DDP length of the packet.

Cause: Programming error at remote node.

ZIP2.039

Level: C-INFO

Short Syntax: ZIP2.039 unk nt *network_number* in qry frm *src_net/ src_node* nt *network*

Long Syntax: ZIP2.039 Unknown network number *network_number* in Query from *src_net/ src_node* net *network*

Description: A ZIP query packet was received with the specified network number in the ZIP data, but this network is not in the RTMP database, or does not have a zone name in the ZIP database. Processing of the packet will continue.

ZIP2.040

Level: UE-ERROR

Short Syntax: ZIP2.040 unk nt *network_number* in rply frm *src_net/ src_node* nt *network*

Long Syntax: ZIP2.040 Unknown network number *network_number* in Reply from *src_net/ src_node* net *network*

Description: A ZIP Reply packet was received with the specified network number in the ZIP data, but this network is not in the RTMP database. Processing of the packet will continue.

ZIP2.041

Level: C-INFO

Short Syntax: ZIP2.041 rq on unseed pt frm *src_net/ src_node* nt *network*

Long Syntax: ZIP2.041 Request on unseeded port from *src_net/ src_node* net *network*

Description: A ZIP query or request was received on an unseeded port that hasn't obtained its net range from a seeded router. Processing of the packet will stop.

ZIP2.042

Level: UE-ERROR

Short Syntax: ZIP2.042 rply bd tpl nm len *length* nt *network* frm *src_net/ src_node* nt *network*, ign

Long Syntax: ZIP2.042 reply bad tuple name length *length* network *network* from *src_net/ src_node* net *network*, ignored

Description: A ZIP reply packet was received where one of the zone names was not of a legal length (between 1 and 32 characters). Processing of the reply ends with the ZIP tuple for the noted network number.

ZIP2.043

Level: UI-ERROR

Short Syntax: ZIP2.043 no mem for GtNtInf rq nt *network*

Long Syntax: ZIP2.043 no memory for GetNetInfo request net *network*

Description: The router was unable to generate a GetNetInfo request for the indicated network because no memory was available for the outgoing packet.

ZIP2.044

Level: UI-ERROR

Short Syntax: ZIP2.044 GtNtInf disc nt *network* rsn *error_code*

Long Syntax: ZIP2.044 GetNetInfo discarded net *network* reason *error_code*

Description: A GetNetInfo request was not transmitted on the indicated net for the specified reason.

ZIP2.045

Level: C-INFO

Short Syntax: ZIP2.045 GtNtInf brdcst nt *network*

Long Syntax: ZIP2.045 GetNetInfo broadcast on net *network*

Description: A GetNetInfo request for the indicated net was broadcast on the specified interface.

ZIP2.046

Level: UE-ERROR

Short Syntax: ZIP2.046 zone *zonename* filtered from nt *network*

Long Syntax: ZIP2.046 zonename *zonename* filtered from net *network*

Description: Zonename information was received on an interface but filtered by the input filter list.

ZIP2.047

Level: C-INFO

Short Syntax: ZIP2.047 query for *net_num* snt to *net_num*/ *node_num* nt *network*

Long Syntax: ZIP2.047 query for *net_num* sent to *net_num*/ *node_num* net *network*

Description: A ZIP query was sent for the indicated net to the specified router.

ZIP2.048

Level: UE-ERROR

Short Syntax: ZIP2.048 unrcgnzd ZIP typ *type* fr *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.048 unrecognized ZIP type *type* from *src_net*/ *src_node* net *network*

Description: A ZIP packet with an unrecognized command type was encountered.

ZIP2.049

Level: UI-ERROR

Short Syntax: ZIP2.049 no buf for ZIP rply to *net_num*/ *node*

Long Syntax: ZIP2.049 no packet buffer for ZIP reply to *net_num*/ *node*

Description: No packet buffer was available for sending a ZIP reply to the specified router.

ZIP2.050

Level: C-INFO

Short Syntax: ZIP2.050 rply net *net_num* snt to *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.050 reply net *net_num* sent to *src_net*/ *src_node* net *network*

Description: A ZIP reply was sent to the indicated router.

ZIP2.051

Level: UE-ERROR

Short Syntax: ZIP2.051 short (*length*) frm *src_net*/ *src_node* nt *network*

Long Syntax: ZIP2.051 packet short (*length* bytes) from *src_net*/ *src_node* net *network*

Description: A ZIP packet was received that was not long enough to contain the 2 byte ZIP header after the DDP header. The packet will be discarded.

Chapter 8. Asynchronous Transfer Mode Network Interface (ATM)

This chapter describes Asynchronous Transfer Mode Network Interface (ATM) messages. For information on message content and how to use the message, refer to the Introduction.

ATM.001

Level: C-INFO

Short Syntax: ATM.001 Create configuration support, nt *network ID*

Long Syntax: ATM.001 Create configuration support, on network *network ID*

Description: Trying to create the config_support object.

ATM.002

Level: C-INFO

Short Syntax: ATM.002 Create Timer_master, nt *network ID*

Long Syntax: ATM.002 Create Timer_master, on network *network ID*

Description: Trying to create the Timer_master object.

ATM.003

Level: C-INFO

Short Syntax: ATM.003 Create connection manager, nt *network ID*

Long Syntax: ATM.003 Create connection manager, on network *network ID*

Description: Trying to create the conn_mgr object.

ATM.004

Level: C-INFO

Short Syntax: ATM.004 Create ilmi_wrapper, nt *network ID*

Long Syntax: ATM.004 Create ilmi_wrapper, on network *network ID*

Description: Trying to create the ilmi_wrapper object.

ATM.005

Level: C-INFO

Short Syntax: ATM.005 Create ilmi, nt *network ID*

Long Syntax: ATM.005 Create ilmi, on network *network ID*

Description: Trying to create the ilmi (ATM_address_table) object.

ATM.006

Level: C-INFO

Short Syntax: ATM.006 Create ilmi_user, nt *network ID*

Long Syntax: ATM.006 Create ilmi_user, on network *network ID*

Description: Trying to create the ilmi_user object.

ATM.007

Level: C-INFO

Short Syntax: ATM.007 Create saal_wrapper, nt *network ID*

Long Syntax: ATM.007 Create saal_wrapper, on network *network ID*

Description: Trying to create the saal_wrapper object.

ATM.008

Level: C-INFO

Short Syntax: ATM.008 Create qsaal, nt *network ID*

Long Syntax: ATM.008 Create qsaal, on network *network ID*

Description: Trying to create the qsaal object.

ATM.009

Level: C-INFO

Short Syntax: ATM.009 Create signalling (Q93B_protocol), nt *network ID*

Long Syntax: ATM.009 Create signalling (Q93B_protocol), on network *network ID*

Description: Trying to create the Q93B_protocol (Signalling) object.

ATM.010

Level: C-INFO

Short Syntax: ATM.010 Calling object_addrs_avail, nt *network ID*

Long Syntax: ATM.010 Calling object_addrs_avail, on network *network ID*

Description: Calling object_addrs_avail for all created objects.

ATM.011

Level: C-INFO

Short Syntax: ATM.011 Starting ilmi_wrapper, nt *network ID*

Long Syntax: ATM.011 Starting ilmi_wrapper, on network *network ID*

Description: Calling ilmi_wrapper->start.

ATM.012

Level: C-INFO

Short Syntax: ATM.012 Starting ilmi, nt *network ID*

Long Syntax: ATM.012 Starting ilmi, on network *network ID*

Description: Calling ilmi->start.

ATM.013

Level: UI-ERROR

Short Syntax: ATM.013 Start of ilmi_wrapper failed, nt *network ID*, rc *retcode*

Long Syntax: ATM.013 Start of ilmi_wrapper failed, on network *network ID*, return code = *retcode*

Description: ilmi_wrapper->start failed.

ATM.014

Level: UI-ERROR

Short Syntax: ATM.014 Start of ilmi failed, nt *network ID*, rc *retcode*

Long Syntax: ATM.014 Start of ilmi failed, on network *network ID*, return code = *retcode*

Description: ilmi->start failed.

ATM.015

Level: UI-ERROR

Short Syntax: ATM.015 Creation of configuration support failed, nt *network ID*, rc *retcode*

Long Syntax: ATM.015 Creation of configuration support failed, on network *network ID*, return code = *retcode*

Description: Create config_support failed.

ATM.016

Level: UI-ERROR

Short Syntax: ATM.016 Creation of Timer_master failed, nt *network ID*, rc *retcode*

Long Syntax: ATM.016 Creation of Timer_master failed, on network *network ID*, return code = *retcode*

Description: Create Timer_master failed.

ATM.017

Level: UI-ERROR

Short Syntax: ATM.017 Creation of connection manager failed, nt *network ID*, rc *retcode*

Long Syntax: ATM.017 Creation of connection manager failed, on network *network ID*, return code = *retcode*

Description: Create conn_mgr failed.

ATM.018

Level: UI-ERROR

Short Syntax: ATM.018 Creation of ilmi_wrapper failed, nt *network ID*, rc *retcode*

Long Syntax: ATM.018 Creation of ilmi_wrapper failed, on network *network ID*, return code = *retcode*

Description: Create ilmi_wrapper failed.

ATM.019

Level: UI-ERROR

Short Syntax: ATM.019 Creation of ilmi failed, nt *network ID*, rc *retcode*

Long Syntax: ATM.019 Creation of ilmi failed, on network *network ID*, return code = *retcode*

Description: Create ilmi failed.

ATM.020

Level: UI-ERROR

Short Syntax: ATM.020 Creation of ilmi_user failed, nt *network ID*

Long Syntax: ATM.020 Creation of ilmi_user failed, on network *network ID*

Description: Create ilmi_user failed.

ATM.021

Level: UI-ERROR

Short Syntax: ATM.021 Creation of saal_wrapper failed, nt *network ID*, rc *retcode*

Long Syntax: ATM.021 Creation of saal_wrapper failed, on network *network ID*, return code = *retcode*

Description: Create saal_wrapper failed.

ATM.022

Level: UI-ERROR

Short Syntax: ATM.022 Creation of qsaal failed, nt *network ID*, rc *retcode*

Long Syntax: ATM.022 Creation of qsaal failed, on network *network ID*, return code = *retcode*

Description: Create qsaal failed.

ATM.023

Level: UI-ERROR

Short Syntax: ATM.023 Creation of signalling (Q93B_protocol) failed, nt *network ID*, rc *retcode*

Long Syntax: ATM.023 Creation of signalling (Q93B_protocol) failed, on network *network ID*, return code = *retcode*

Description: Create Q93B_protocol failed.

ATM.024

Level: UI-ERROR

Short Syntax: ATM.024 Bad inbound control frame discarded, handle = *conn_handle* nt *network ID*

Long Syntax: ATM.024 Bad inbound control frame discarded, handle = *conn_handle*, on network *network ID*

Description: Bad inbound control frame

ATM.025

Level: C-INFO

Short Syntax: ATM.025 Frame xmit by net_dsio, handle = *conn_handle* nt *network ID*

Long Syntax: ATM.025 Frame transmitted using net_dsio, handle = *conn_handle*, on network *network ID*

Description: A frame has been transmitted on the ATM interface, using net_dsio.

ATM.026

Level: UI-ERROR

Short Syntax: ATM.026 Inbound frame discarded, handle = *conn_handle* nt *network ID*, rc *stat*

Long Syntax: ATM.026 Inbound frame discarded, handle = *conn_handle*, on network *network ID*, status = *stat*

Description: Bad status in iorb

ATM.027

Level: UI-ERROR

Short Syntax: ATM.027 Outbound frame not sent, handle = *conn_handle* nt *network ID*, rc *retcode*

Long Syntax: ATM.027 Outbound frame not sent, handle = *conn_handle*, on network *network ID*, return code = *retcode*

Description: Bad status from netout

ATM.028

Level: UI-ERROR

Short Syntax: ATM.028 Device Driver User Registration Failed, handle = *conn_handle* nt *network ID*, rc *stat*

Long Syntax: ATM.028 Device Driver User Registration Failed, handle = *conn_handle*, on network *network ID*, status = *stat*

Description: atmcharm_init User Registration Failed

ATM.029

Level: UI-ERROR

Short Syntax: ATM.029 Device Driver Wrap User Registration Failed, handle = *conn_handle* nt *network ID*, rc *rc*

Long Syntax: ATM.029 Device Driver Wrap User Registration Failed, handle = *conn_handle*, on network *network ID*, status = *rc*

Description: atmcharm_init Wrap User Registration Failed

ATM.030

Level: UI-ERROR

Short Syntax: ATM.030 ERROR opening Frame SAP, handle = *conn_handle* nt *network ID*, rc *stat*

Long Syntax: ATM.030 ERROR opening Frame SAP, handle = *conn_handle*, on network *network ID*, status = *stat*

Description: Couldn't open Frame SAP

ATM.031

Level: UI-ERROR

Short Syntax: ATM.031 ERROR opening data path for ILMI, handle = *conn_handle* nt *network ID*, rc *stat*

Long Syntax: ATM.031 ERROR opening data path for ILMI, handle = *conn_handle*, on network *network ID*, status = *stat*

Description: Couldn't open data path for ILMI

ATM.032

Level: UI-ERROR

Short Syntax: ATM.032 ERROR opening Wrap Frame SAP, handle = *conn_handle* nt *network ID*, rc *rc*

Long Syntax: ATM.032 ERROR opening Wrap Frame SAP, handle = *conn_handle*, on network *network ID*, status = *rc*

Description: Couldn't open Wrap Frame SAP

ATM.033

Level: UI-ERROR

Short Syntax: ATM.033 ERROR creating objects, handle = *conn_handle* nt *network ID*, rc *rc*

Long Syntax: ATM.033 ERROR creating objects, handle = *conn_handle*, on network *network ID*, status = *rc*

Description: Couldn't create objects (SVC, ILMI, etc.)

ATM.034

Level: UI-ERROR

Short Syntax: ATM.034 Inbound frame discarded, handle = *conn_handle* nt *network ID*, cnt *i_bxfr*

Long Syntax: ATM.034 Inbound frame discarded, handle = *conn_handle*, on network *network ID*, count = *i_bxfr*

Description: Bad byte count in iorb

ATM.035

Level: C-INFO

Short Syntax: ATM.035 Function *function_name* called, nt *network ID*

Long Syntax: ATM.035 Function *function_name* called, on network *network ID*

Description: ATM function called

ATM.036

Level: UI-ERROR

Short Syntax: ATM.036 Could not stop objects, nt *network ID*, rc *rc*

Long Syntax: ATM.036 Could not stop objects, on network *network ID*, return code = *rc*

Description: Error stopping the objects

ATM.037

Level: UI-ERROR

Short Syntax: ATM.037 Connection handle is NULL, nt *network ID*

Long Syntax: ATM.037 Connection handle is NULL, on network *network ID*

Description: conn_handle is NULL

ATM.038

Level: UI-ERROR

Short Syntax: ATM.038 Unable to get buffers, nt *network ID*

Long Syntax: ATM.038 Unable to get buffers on network *network ID*

Description: Could not get a buffer

ATM.039

Level: UI-ERROR

Short Syntax: ATM.039 Connect ioctl failed , nt *network ID* vpi/vci *vpi/ vci*

Long Syntax: ATM.039 Connect ioctl failed , nt *network ID* vpi/vci *vpi/ vci*

Description: Connect ioctl failed

ATM.040

Level: UI-ERROR

Short Syntax: ATM.040 Could not start objects, nt *network ID*, rc *rc*

Long Syntax: ATM.040 Could not start objects, on network *network ID*, return code = *rc*

Description: Error starting the objects

ATM.041

Level: UI-ERROR

Short Syntax: ATM.041 Invalid max SDU size, nt *network ID*, SDU sz *rate*

Long Syntax: ATM.041 Invalid max SDU size, on network *network ID*, max SDU size = *rate*

Description: ATM net handler was passed invalid max SDU size for connection

---

ATM.042

Level: UI-ERROR

Short Syntax: ATM.042 Invalid peak cell rate, nt *network ID*, pk rate *rate*

Long Syntax: ATM.042 Invalid peak cell rate, on network *network ID*, peak cell rate = *rate*

Description: ATM net handler was passed invalid peak cell rate for connection

---

ATM.043

Level: UI-ERROR

Short Syntax: ATM.043 Invalid sustainable cell rate, nt *network ID*, sustn rate *rate*

Long Syntax: ATM.043 Invalid sustainable cell rate, on network *network ID*, sustainable cell rate = *rate*

Description: ATM net handler was passed invalid sustainable cell rate for connection

---

ATM.044

Level: UI-ERROR

Short Syntax: ATM.044 Invalid max burst size, nt *network ID*, brst sz *rate*

Long Syntax: ATM.044 Invalid max burst size, on network *network ID*, max burst size = *rate*

Description: ATM net handler was passed invalid max burst size for connection

---

ATM.045

Level: UI-ERROR

Short Syntax: ATM.045 API, invalid net number, nt *network ID*, dev *devNum*

Long Syntax: ATM.045 API called with invalid net number, on network *network ID*, devNum *devNum*

Description: atmUserRegistration called with invalid devNum

ATM.046

Level: UI-ERROR

Short Syntax: ATM.046 API call failed, no memory, nt *network ID*

Long Syntax: ATM.046 API call failed, no memory, on network *network ID*

Description: ATM API call failed, no memory available

ATM.047

Level: UI-ERROR

Short Syntax: ATM.047 Get UNI version ILMI failed, nt *network ID*, rc *rc*

Long Syntax: ATM.047 Get UNI Version ILMI failed, on network *network ID*, return code = *rc*

Description: atmGetUniVersion ILMI returned get_uni_version_wrap with bad return

---

ATM.048

Level: UI-ERROR

Short Syntax: ATM.048 API, invalid user handle, *userHandle*

Long Syntax: ATM.048 API called with invalid user handle, *userHandle*

Description: atmUserRegistration called with invalid user handle

---

ATM.049

Level: CI-ERROR

Short Syntax: ATM.049 API call failed, net down, nt *network ID*

Long Syntax: ATM.049 API call failed, network down, on network *network ID*

Description: ATM API call failed, network is down

ATM.050

Level: C-INFO

Short Syntax: ATM.050 Address activation ILMI successful, nt *network ID*, rc *rc*

Long Syntax: ATM.050 ATM address activation ILMI successful, on network *network ID*, return code = *rc*

Description: atmAddrActivation ILMI returned alloc_addr_wrap with good return

---

ATM.051

Level: UI-ERROR

Short Syntax: ATM.051 Address activation ILMI failed, nt *network ID*, rc *rc*

Long Syntax: ATM.051 ATM address activation ILMI failed, on network *network ID*, return code = *rc*

Description: atmAddrActivation ILMI returned alloc_addr_wrap with bad return

---

ATM.052

Level: C-INFO

Short Syntax: ATM.052 Sharing ESI/Selector, nt *network ID*, addr *addr*

Long Syntax: ATM.052 Sharing ESI/Selector, on network *network ID*, addr *addr*

Description: atmAddrActivation providing sharing of the ATM address

ATM.053

Level: C-INFO

Short Syntax: ATM.053 conn_mgr dstrc ntrd, nt *network ID*

Long Syntax: ATM.053 Connection Manager destructor entered, on network *network ID*

Description: Connection Manager destructor entered

ATM.054

Level: UI-ERROR

Short Syntax: ATM.054 API, invalid address handle, nt *network ID*, hndl *handle*

Long Syntax: ATM.054 API called with invalid address handle, on network *network ID*, handle = *handle*

Description: atmAddrDeactivation called with invalid address handle

ATM.055

Level: UI-ERROR

Short Syntax: ATM.055 Address deactivation ILMI failed, nt *network ID*, hndl *handle* rc *rc*

Long Syntax: ATM.055 ATM address deactivation ILMI failed, on network *network ID*, handle = *handle*, return code = *rc*

Description: atmAddrDeactivation ILMI returned free_addr_handle_wrap with bad return

ATM.056

Level: UI-ERROR

Short Syntax: ATM.056 Get address by handle ILMI failed, nt *network ID*, hndl *handle* rc *rc*

Long Syntax: ATM.056 Get ATM address by handle ILMI failed, on network *network ID*, handle = *handle*, return code = *rc*

Description: atmGetAddrByHandle ILMI returned get_atm_addr_wrap with bad return

ATM.057

Level: C-INFO

Short Syntax: ATM.057 Get UNI version ILMI successful, nt *network ID*, rc *rc*

Long Syntax: ATM.057 Get UNI Version ILMI successful, on network *network ID*, return code = *rc*

Description: atmGetUniVersion ILMI returned get_uni_version_wrap with good return

ATM.058

Level: C-INFO

Short Syntax: ATM.058 Get LECS address ILMI successful, nt *network ID*, rc *rc*

Long Syntax: ATM.058 Get LECS address ILMI successful, on network *network ID*, return code = *rc*

Description: atmGetLecsAddr ILMI returned get_lecs_list_wrap with good return

ATM.059

Level: UI-ERROR

Short Syntax: ATM.059 Get LECS address ILMI failed, nt *network ID*, rc *rc*

Long Syntax: ATM.059 Get LECS address ILMI failed, on network *network ID*, return code = *rc*

Description: atmGetLecsAddr ILMI returned get_lecs_list_wrap with bad return

ATM.060

Level: UI-ERROR

Short Syntax: ATM.060 call SAP already open, nt *network ID* hndl *handle*

Long Syntax: ATM.060 call SAP already open for this user, network *network ID*, SAP handle = *handle*

Description: call SAP already open for this user

ATM.061

Level: UI-ERROR

Short Syntax: ATM.061 max call SAPs exceeded, nt *network ID*

Long Syntax: ATM.061 max call SAPs exceeded, network *network ID*

Description: Maximum call SAPs already opened.

ATM.062

Level: UI-ERROR

Short Syntax: ATM.062 bad PID count, nt *network ID* cnt *count*

Long Syntax: ATM.062 bad PID count, network *network ID*, count = *count*

Description: PID count in PID list out of range

ATM.063

Level: UI-ERROR

Short Syntax: ATM.063 bad PID, nt *network ID* PID *PID*

Long Syntax: ATM.063 bad PID, network *network ID*, PID = *PID*

Description: Invalid PID in PID list

ATM.064

Level: UI-ERROR

Short Syntax: ATM.064 Conn mgr register caller failed, nt *network ID*, rc *rc*

Long Syntax: ATM.064 Connection manager register caller failed, on network *network ID*, return code = *rc*

Description: Connection Manager call to reg_caller_wrap failed.

ATM.065

Level: UI-ERROR

Short Syntax: ATM.065 API, invalid call SAP hndl, nt *network ID*, hndl *handle*

Long Syntax: ATM.065 API received invalid call SAP handle, on network *network ID*, handle = *handle*

Description: Invalid call SAP handle passed to API

ATM.066

Level: C-INFO

Short Syntax: ATM.066 API, placing call, nt *network ID*, addr *address*

Long Syntax: ATM.066 API, placing call, on network *network ID*, ATM address = *address*

Description: Placing a call to a given address

ATM.067

Level: UI-ERROR

Short Syntax: ATM.067 API, place call failed, nt *network ID*, hndl *handle*, addr *address*, rc *rc*

Long Syntax: ATM.067 API, place call failed, on network *network ID*, handle = *handle*, ATM address = *address*, return code = *rc*

Description: Placing a call to a given address failed

ATM.068

Level: C-INFO

Short Syntax: ATM.068 API, adding leaf, nt *network ID*, addr *address*, conn hndl = *handle*

Long Syntax: ATM.068 API, adding leaf, on network *network ID*, ATM address = *address*, conn handle = *handle*,

Description: Adding a leaf to a multipoint call

ATM.069

Level: UI-ERROR

Short Syntax: ATM.069 API, add leaf failed, nt *network ID*, hndl *handle*, addr *address*, rc *rc*

Long Syntax: ATM.069 API, add leaf failed, on network *network ID*, handle = *handle*, ATM address = *address*, return code = *rc*

Description: Adding a leaf to a multipoint call failed

ATM.070

Level: C-INFO

Short Syntax: ATM.070 API, hangup leaf, nt *network ID*, conn hndl *connHandle*, leaf hndl *leafHandle*

Long Syntax: ATM.070 API, hangup leaf, on network *network ID*, conn handle = *connHandle*, leaf handle = *leafHandle*

Description: Hanging up a leaf

ATM.071

Level: C-INFO

Short Syntax: ATM.071 API, recv call ack, nt *network ID*, conn hndl *connHandle*

Long Syntax: ATM.071 API, receive call ack, on network *network ID*, conn handle = *connHandle*

Description: Receive call ack

ATM.072

Level: C-INFO

Short Syntax: ATM.072 API, hangup call, nt *network ID*, conn hndl *connHandle*

Long Syntax: ATM.072 API, hangup call, on network *network ID*, conn handle = *connHandle*

Description: Hanging up a call

ATM.073

Level: UI-ERROR

Short Syntax: ATM.073 API, invalid frame SAP type, nt *network ID*, type *type*

Long Syntax: ATM.073 API called with invalid frame SAP type, on network *network ID*, type = *type*

Description: Invalid frame SAP type passed to API

ATM.074

Level: UI-ERROR

Short Syntax: ATM.074 API, invalid frame SAP hndl, nt *network ID*, hndl *handle*

Long Syntax: ATM.074 API called with invalid frame SAP handle, on network *network ID*, handle = *handle*

Description: Invalid frame SAP handle passed to API

ATM.075

Level: UI-ERROR

Short Syntax: ATM.075 API, invalid VCC hndl, nt *network ID*, hndl *handle*

Long Syntax: ATM.075 API called with invalid VCC handle, on network *network ID*, handle = *handle*

Description: Invalid VCC handle passed to API

ATM.076

Level: UI-ERROR

Short Syntax: ATM.076 API, invalid MAC offset, nt *network ID*, offset *offset*

Long Syntax: ATM.076 API called with invalid MAC address offset, on network *network ID*, offset = *offset*

Description: Invalid MAC address offset passed to API

ATM.077

Level: UI-ERROR

Short Syntax: ATM.077 API, invalid VCC grp hndl, nt *network ID*, grp hndl *handle*

Long Syntax: ATM.077 API called with invalid VCC group handle, on network *network ID*, group handle = *handle*

Description: Invalid VCC group handle passed to API

ATM.078

Level: UI-ERROR

Short Syntax: ATM.078 API, VCC already in grp, nt *network ID*, vcc hndl *vcchandle*, grp hndl *grphandle*

Long Syntax: ATM.078 API, VCC already in group, on network *network ID*, vcc handle = *vcchandle*, group handle = *grphandle*

Description: Trying to add a VCC to a group it is already a part of

---

ATM.079

Level: UI-ERROR

Short Syntax: ATM.079 API, VCC not in grp, nt *network ID*, vcc hndl *vcchandle*, grp hndl *grphandle*

Long Syntax: ATM.079 API, VCC not in group, on network *network ID*, vcc handle = *vcchandle*, group handle = *grphandle*

Description: VCC not in this group

ATM.080

Level: UI-ERROR

Short Syntax: ATM.080 API, MAC already mapped, nt *network ID*, vcc hndl *vcchandle*, grp hndl *grphandle*, MAC *MACaddr*

Long Syntax: ATM.080 API, MAC already mapped, on network *network ID*, vcc handle = *vcchandle*, group handle = *grphandle*, MAC address *MACaddr*

Description: Trying to map a MAC address to a group it is already mapped to

---

ATM.081

Level: UI-ERROR

Short Syntax: ATM.081 API, MAC not mapped, nt *network ID*, vcc hndl *vcchandle*, grp hndl *grphandle*, MAC *MACaddr*

Long Syntax: ATM.081 API, MAC not mapped, on network *network ID*, vcc handle = *vcchandle*, group handle = *grphandle*, MAC address *MACaddr*

Description: Trying to unmap a MAC address to a group it is not mapped to

---

ATM.082

Level: C-INFO

Short Syntax: ATM.082 addr state change, nt *network ID*, addr *address*, state = *state*

Long Syntax: ATM.082 ATM address state change, on network *network ID*, ATM address = *address*, state= *state*,

Description: Address state change

ATM.083

Level: UI-ERROR

Short Syntax: ATM.083 Connection manager start failed, nt *network ID*, rc *retcode*

Long Syntax: ATM.083 Connection manager start failed, on network *network ID*, return code = *retcode*

Description: Connection Manager start failed.

ATM.084

Level: UI-ERROR

Short Syntax: ATM.084 SAAL wrapper start failed, nt *network ID*, rc *retcode*

Long Syntax: ATM.084 SAAL wrapper start failed, on network *network ID*, return code = *retcode*

Description: SAAL wrapper start failed.

ATM.085

Level: UI-ERROR

Short Syntax: ATM.085 SAAL start failed, nt *network ID*, rc *retcode*

Long Syntax: ATM.085 SAAL start failed, on network *network ID*, return code = *retcode*

Description: SAAL start failed.

ATM.086

Level: UI-ERROR

Short Syntax: ATM.086 SVC start failed, nt *network ID*, rc *retcode*

Long Syntax: ATM.086 SVC start failed, on network *network ID*, return code = *retcode*

Description: SVC start failed.

ATM.087

Level: C-INFO

Short Syntax: ATM.087 Conn mgr stopped, nt *network ID*

Long Syntax: ATM.087 Connection Manager stopped, on network *network ID*

Description: Connection Manager stop entered

ATM.088

Level: P_TRACE

Short Syntax: ATM.088 Trace ATM frame.

Long Syntax: ATM.088 Trace ATM frame.

Description: ATM frame packet tracing.

ATM.089

Level: UI-ERROR

Short Syntax: ATM.089 Conn mgr place call failed, nt *network ID*, rc *retcode*

Long Syntax: ATM.089 Connection manager place call failed, on network *network ID*, rc = *retcode*

Description: Conn Mgr place call failed.

ATM.090

Level: U-INFO

Short Syntax: ATM.090 Call setup failed, SAAL not up yet, nt *network ID*

Long Syntax: ATM.090 Call setup failed, SAAL not up yet, on network *network ID*

Description: Call set-up failed because the SAAL was not established yet.

ATM.091

Level: UI-ERROR

Short Syntax: ATM.091 Conn handle in use, nt *network ID*, hdnl *handle*

Long Syntax: ATM.091 SVC thinks conn handle is in use, on network *network ID*, handle = *handle*

Description: Call set-up failed because the SVC thought the conn handle was in use.

ATM.092

Level: UI-ERROR

Short Syntax: ATM.092 Invalid conn handle, nt *network ID*, hdnl *handle*

Long Syntax: ATM.092 Invalid conn handle, no entry in connection table, on network *network ID*, handle = *handle*

Description: Call set-up failed because the conn handle points to a NULL entry.

ATM.093

Level: UI-ERROR

Short Syntax: ATM.093 Place call ack failed, nt *network ID*, rc *retcode*, hdnl *handle*, vpi *vpi*, vci *vci*

Long Syntax: ATM.093 Place call ack failed, on network *network ID*, rc = *retcode*, handle = *handle*, vpi = *vpi*, vci = *vci*

Description: Place call ack failed

ATM.094

Level: UE-ERROR

Short Syntax: ATM.094 Receive call failed, no such caller, nt *network ID*, hdnl *handle*, vpi *vpi*, vci *vci*

Long Syntax: ATM.094 Receive call failed, no such caller, on network *network ID*, handle = *handle*, vpi = *vpi*, vci = *vci*

Description: No caller found matching call parms.

ATM.095

Level: UE-ERROR

Short Syntax: ATM.095 call rjct, nt *network ID*, rc *retcode*, hdnl *handle*, vpi *vpi*, vci *vci*

Long Syntax: ATM.095 Call rejected, on network *network ID*, rc = *retcode*, handle = *handle*, vpi = *vpi*, vci = *vci*

Description: Called party rejected call.

ATM.096

Level: C-INFO

Short Syntax: ATM.096 Call accepted, nt *network ID*, rc *retcode*, hdnl *handle*, vpi *vpi*, vci *vci*

Long Syntax: ATM.096 Call accepted, on network *network ID*, rc = *retcode*, handle = *handle*, vpi = *vpi*, vci = *vci*

Description: Called party accepted call.

ATM.097

Level: C-INFO

Short Syntax: ATM.097 Receive call ack, nt *network ID*, rc *retcode*, hdnl *handle*

Long Syntax: ATM.097 Receive call ack, on network *network ID*, rc = *retcode*, handle = *handle*

Description: Called party accepts or rejects call.

ATM.098

Level: UI-ERROR

Short Syntax: ATM.098 Bad cnfg prm, n_int *interface*, rc *retcode*, prm *parm*

Long Syntax: ATM.098 Bad configuration parm, n_int = *interface*, rc = *retcode*, parm = *parm*

Description: Conn Mgr attempt to read configuration parameter failed.

ATM.099

Level: UI-ERROR

Short Syntax: ATM.099 Conn mgr, no memory, n_int *interface*, pnt *point*

Long Syntax: ATM.099 Connection Manager, no memory, n_int = *interface*, trace point = *point*

Description: Connection Manager could not get memory to initialize

---

ATM.100

Level: C-INFO

Short Syntax: ATM.100 Conn mgr adding leaf, nt *network ID*, hndl *handle*

Long Syntax: ATM.100 Connection Manager adding leaf, on network *network ID*, handle = *handle*

Description: Connection Manager add leaf entered

ATM.101

Level: UI-ERROR

Short Syntax: ATM.101 Invld conn hndl, not in connect tbl, nt *network ID*, hdnl *handle*, func *function_name*

Long Syntax: ATM.101 Invalid conn handle, no entry in connection table, on network *network ID*, handle = *handle*, function *function_name*

Description: The conn handle points to a NULL entry.

ATM.102

Level: UI-ERROR

Short Syntax: ATM.102 Invld conn hndl, free connection, nt *network ID*, hdnl *handle*, func *function_name*

Long Syntax: ATM.102 Invalid conn handle, free connection, on network *network ID*, handle = *handle*, function = *function_name*

Description: The connection handle points to a inactive entry.

ATM.103

Level: UI-ERROR

Short Syntax: ATM.103 No leaf handle available, nt *network ID*, hdnl *handle*

Long Syntax: ATM.103 No leaf handle available, on network *network ID*, conn handle = *handle*

Description: Could not get leaf handle

ATM.104

Level: UI-ERROR

Short Syntax: ATM.104 Add leaf handle failed, nt *network ID*, hdnl *handle*, rc *retcode*

Long Syntax: ATM.104 Add leaf handle failed, on network *network ID*, conn handle = *handle*, rc = *retcode*

Description: Add leaf handle failed

ATM.105

Level: C-INFO

Short Syntax: ATM.105 Conn mgr add leaf ack, nt *network ID*, hndl *handle*

Long Syntax: ATM.105 Connection Manager add leaf ack, on network *network ID*, handle = *handle*

Description: Connection Manager add leaf ack entered

ATM.106

Level: C-INFO

Short Syntax: ATM.106 Conn mgr registering caller, nt *network ID*

Long Syntax: ATM.106 Connection Manager registering caller, on network *network ID*

Description: Connection Manager register_caller entered

ATM.107

Level: UI-ERROR

Short Syntax: ATM.107 Max callers exceeded, nt *network ID*

Long Syntax: ATM.107 Max callers exceeded, on network *network ID*

Description: Caller tried to register, maximum callers already registered.

ATM.108

Level: UI-ERROR

Short Syntax: ATM.108 dup cllr PID, nt *network ID*

Long Syntax: ATM.108 Duplicate caller PID, on network *network ID*

Description: A caller tried to register with the same PID and address as an existing caller.

ATM.109

Level: UI-ERROR

Short Syntax: ATM.109 dup cllr addr, nt *network ID*

Long Syntax: ATM.109 Duplicate caller address, on network *network ID*

Description: A caller tried to register with the same address as an existing caller.

ATM.110

Level: C-INFO

Short Syntax: ATM.110 Conn mgr deregistering caller, nt *network ID*

Long Syntax: ATM.110 Connection Manager deregistering caller, on network *network ID*

Description: Connection Manager deregister_caller entered

ATM.111

Level: C-INFO

Short Syntax: ATM.111 Conn mgr disconnecting call, nt *network ID*, hndl *handle*

Long Syntax: ATM.111 Connection Manager disconnecting call, on network *network ID*, handle = *handle*

Description: Connection Manager disconnect_call entered

ATM.112

Level: C-INFO

Short Syntax: ATM.112 rmv cnxn, SVC err, nt *network ID*, hndl *handle*

Long Syntax: ATM.112 Conn Mgr removing connection, SVC error, on network *network ID*, handle = *handle*

Description: SVC got a conn handle, found an error, and is giving it back

ATM.113

Level: C-INFO

Short Syntax: ATM.113 Conn mgr reporting failure, nt *network ID*, hndl *handle*

Long Syntax: ATM.113 Connection Manager reporting failure to caller, on network *network ID*, handle = *handle*

Description: Connection Manager report_failure_to_Caller entered

ATM.114

Level: C-INFO

Short Syntax: ATM.114 Conn mgr removing connection, nt *network ID*, hndl *handle*

Long Syntax: ATM.114 Connection Manager removing connection, on network *network ID*, handle = *handle*

Description: Connection Manager remove connection entered

ATM.115

Level: C-INFO

Short Syntax: ATM.115 Conn mgr disconnecting leaf, nt *network ID*, hndl *handle*

Long Syntax: ATM.115 Connection Manager disconnecting leaf, on network *network ID*, handle = *handle*

Description: Connection Manager disconnect_leaf entered

ATM.116

Level: C-INFO

Short Syntax: ATM.116 Conn mgr getting conn hndl, nt *network ID*

Long Syntax: ATM.116 Connection Manager getting conn handle, on network *network ID*

Description: Connection Manager get_conn_handle entered

ATM.117

Level: UI-ERROR

Short Syntax: ATM.117 no conn handles, nt *network ID*

Long Syntax: ATM.117 All connection handles in use, on network *network ID*

Description: No free conn handles

ATM.118

Level: UI-ERROR

Short Syntax: ATM.118 no mem conn obj, nt *network ID*

Long Syntax: ATM.118 No memory for connection object, on network *network ID*

Description: No memory for connection object

ATM.119

Level: C-INFO

Short Syntax: ATM.119 Conn mgr hanging up call, nt *network ID*, hndl *handle*

Long Syntax: ATM.119 Connection Manager hanging up call, on network *network ID*, handle = *handle*

Description: Connection Manager hang_up_call entered

ATM.120

Level: C-INFO

Short Syntax: ATM.120 conn_mgr hanging up leaf, nt *network ID*, hndl *handle*

Long Syntax: ATM.120 Connection Manager hanging up leaf, on network *network ID*, handle = *handle*

Description: Connection Manager hang_up_leaf entered

ATM.121

Level: UI-ERROR

Short Syntax: ATM.121 Hang up leaf failed, nt *network ID*, rc *retcode*

Long Syntax: ATM.121 SVC hang up leaf failed, on network *network ID*, rc = *retcode*

Description: SVC hang_up_leaf failed

ATM.122

Level: C-INFO

Short Syntax: ATM.122 Hang up leaf success, nt *network ID*, rc *retcode*

Long Syntax: ATM.122 SVC hang up leaf success, on network *network ID*, rc = *retcode*

Description: SVC hang_up_leaf successful

ATM.123

Level: C-INFO

Short Syntax: ATM.123 Conn mgr placing call, nt *network ID*

Long Syntax: ATM.123 Connection Manager placing call, on network *network ID*

Description: Connection Manager place_call entered

ATM.124

Level: UI-ERROR

Short Syntax: ATM.124 Max calls exceeded, nt *network ID*

Long Syntax: ATM.124 Max calls exceeded, on network *network ID*

Description: Caller tried to place call, maximum calls already placed.

ATM.125

Level: UI-ERROR

Short Syntax: ATM.125 cllr not reg, nt *network ID*, hndl *handle*

Long Syntax: ATM.125 Caller not registered, on network *network ID*, handle = *handle*

Description: Caller not registered.

ATM.126

Level: C-INFO

Short Syntax: ATM.126 Conn mgr place call ack, nt *network ID*

Long Syntax: ATM.126 Connection Manager place call ack, on network *network ID*

Description: Connection Manager place_call_ack entered

ATM.127

Level: C-INFO

Short Syntax: ATM.127 Conn mgr processing received call, nt *network ID*

Long Syntax: ATM.127 Connection Manager processing received call, on network *network ID*

Description: Connection Manager process_receive_call entered

ATM.128

Level: UE-ERROR

Short Syntax: ATM.128 Cllee not reg, nt *network ID*, hndl *handle*

Long Syntax: ATM.128 Callee not registered, on network *network ID*, conn handle = *handle*

Description: Callee not registered.

ATM.129

Level: C-INFO

Short Syntax: ATM.129 Callee found, nt *network ID*, cllr *caller*, cnxn *handle*

Long Syntax: ATM.129 Callee found, on network *network ID*, caller = *caller*, conn_handle = *handle*

Description: Callee found.

ATM.130

Level: C-INFO

Short Syntax: ATM.130 Conn mgr finding caller id, nt *network ID*

Long Syntax: ATM.130 Connection Manager finding caller id, on network *network ID*

Description: Connection Manager find_caller_id entered

ATM.131

Level: UI-ERROR

Short Syntax: ATM.131 Addr not found, nt *network ID*,addr *handle*

Long Syntax: ATM.131 Address not found by ILMI, on network *network ID*, address handle = *handle*

Description: Address not found by ILMI.

ATM.132

Level: C-INFO

Short Syntax: ATM.132 Callee found, nt *network ID*,hndl *handle*

Long Syntax: ATM.132 Callee found, on network *network ID*, caller handle = *handle*

Description: Callee found

ATM.133

Level: UE-ERROR

Short Syntax: ATM.133 cllee not reg, nt *network ID*, hndl *handle*

Long Syntax: ATM.133 Callee not registered, on network *network ID*, address handle = *handle*

Description: Callee not registered.

ATM.134

Level: UI-ERROR

Short Syntax: ATM.134 Conn handle NULL, nt *network ID*

Long Syntax: ATM.134 Connection handle NULL, on network *network ID*

Description: Connection handle in iorb is NULL.

ATM.135

Level: UI-ERROR

Short Syntax: ATM.135 ILMI ptr NULL, nt *network ID*

Long Syntax: ATM.135 ILMI wrapper function called, ILMI pointer is NULL, on network *network ID*

Description: ILMI wrapper function called, ILMI pointer is NULL.

ATM.136

Level: UI-ERROR

Short Syntax: ATM.136 ILMI wrap ptr NULL, nt *network ID*

Long Syntax: ATM.136 ILMI wrapper function called, ILMI wrapper pointer is NULL, on network *network ID*

Description: ILMI wrapper function called, ILMI wrapper pointer is NULL.

ATM.137

Level: UI-ERROR

Short Syntax: ATM.137 ERROR opening data path for SVC, handle = *user_handle* nt *network ID*, rc *stat*

Long Syntax: ATM.137 ERROR opening data path for SVC, handle = *user_handle*, on network *network ID*, status = *stat*

Description: Couldn't open data path for SVC

ATM.138

Level: UI-ERROR

Short Syntax: ATM.138 Unknown adapter type, nt *network ID*, type *adapter_type*

Long Syntax: ATM.138 Unknown adapter type, on network *network ID*, adapter type = *adapter_type*

Description: The adapter returned an unknown adapter type.

ATM.139

Level: UI-ERROR

Short Syntax: ATM.139 Slftst called, nt *network ID* down

Long Syntax: ATM.139 Selftest called, but network *network ID* is down

Description: Self-test was called, but the adapter is down

ATM.140

Level: CE-ERROR

Short Syntax: ATM.140 Prefix removed, nt *network ID*

Long Syntax: ATM.140 The switch removed its prefix on network *network ID*

Description: The ATM switch removed its prefix.

ATM.141

Level: UE-ERROR

Short Syntax: ATM.141 Cnfgd spd not adapter spd, nt *network ID*, cnfg *config*, adapter *adapter*

Long Syntax: ATM.141 Configured speed different from adapter on network *network ID*, config speed = *config*, adapter speed = *adapter*

Description: The adapter speed is differnet from the configured speed

ATM.142

Level: UI-ERROR

Short Syntax: ATM.142 Bad VCC handle, nt *network ID*, hdnl= *handle*

Long Syntax: ATM.142 Bad VCC handle, on network *network ID*, handle = *handle*

Description: The device driver passed a bad VCC handle to the net handler.

ATM.143

Level: C-INFO

Short Syntax: ATM.143 VCC hdnl, nt *network ID*, hdnl= *handle*, VPI= *vpi*,VCI= *vci*

Long Syntax: ATM.143 VCC handle passed to device driver, on network *network ID*, handle = *handle*, VPI = *vpi*, VCI = *vci*

Description: The net handler passed a handle to the device driver for this VPI/VCI.

ATM.144

Level: UI-ERROR

Short Syntax: ATM.144 No bfr for disc, nt *network ID*, VPI= *vpi*,VCI= *vci*

Long Syntax: ATM.144 No buffer for disconnect, on network *network ID*, VPI = *vpi*, VCI = *vci*

Description: No buffer was available to disconnect a VCC.

ATM.145

Level: UI-ERROR

Short Syntax: ATM.145 Frame recvd while disconn pending, nt *network ID*, hndl= *handle*

Long Syntax: ATM.145 Frame received while disconnect pending, on network *network ID*, handle = *handle*

Description: A frame was received when the VCC was in disconnect pending.

ATM.146

Level: UE-ERROR

Short Syntax: ATM.146 No prefix set, nt *network ID*

Long Syntax: ATM.146 The switch never set its prefix on network *network ID*

Description: The ATM switch never set its prefix.

ATM.147

Level: UI-ERROR

Short Syntax: ATM.147 No mem to rd adptr addr, nt *network ID*

Long Syntax: ATM.147 No memory at init to read adapter address, on network *network ID*

Description: No memory available at initialization to read adapter address.

ATM.148

Level: UI-ERROR

Short Syntax: ATM.148 No mem to rd adptr addr, nt *network ID*

Long Syntax: ATM.148 No memory at init to read adapter address, on network *network ID*

Description: No memory available at initialization to read adapter address.

ATM.153

Level: UI-ERROR

Short Syntax: ATM.153 Timer re-entrancy err, nt *net_no*, flag = *flag*, log pt = *log_point*

Long Syntax: ATM.153 Timer re-entrancy error on net *net_no*, flag = *flag*, logpoint = *log_point*

Description: Timer re-entrancy error.

ATM.154

Level: C-INFO

Short Syntax: ATM.154 Timer set alarm, nt *net_no*, ndx = *index*, callback = *user_ptr*

Long Syntax: ATM.154 Timer set alarm on net *net_no*, index = *index*, callback address = *user_ptr*

Description: Timer set.

ATM.155

Level: C-INFO

Short Syntax: ATM.155 Timer set alarm, nt *net_no*, type = *type*, element = *element*

Long Syntax: ATM.155 Timer set alarm on net *net_no*, type = *type*, element address = *element*

Description: Timer set.

ATM.156

Level: C-INFO

Short Syntax: ATM.156 Timer trace, nt *net_no*, log pt = *logpoint*, rc = *rcode*

Long Syntax: ATM.156 Timer trace on net *net_no*, logpoint = *logpoint*, return code = *rcode*

Description: Timer trace.

ATM.157

Level: C-INFO

Short Syntax: ATM.157 Timer trace, nt *net_no*, log pt = *logpoint*, ndx = *index*, element = *element*

Long Syntax: ATM.157 Timer trace on net *net_no*, logpoint = *logpoint*, index = *index*, element = *element*

Description: Timer trace.

ATM.158

Level: UI-ERROR

Short Syntax: ATM.158 Timer already stopped, nt *net_no*, log pt = *logpoint*, ndx = *index*, element = *element*

Long Syntax: ATM.158 Timer already stopped on net *net_no*, logpoint = *logpoint*, index = *index*, element = *element*

Description: Timer already stopped.

ATM.159

Level: UI-ERROR

Short Syntax: ATM.159 User stopping unowned timer, nt *net_no*, stopper = *stopper*, owner = *owner*

Long Syntax: ATM.159 User stopping unowned timer on net *net_no*, stopper = *stopper*, owner = *owner*

Description: User trying to stop another user's timer.

ATM.160

Level: C-INFO

Short Syntax: ATM.160 Timer cancel alarm, nt *net_no*, logpt = *logpoint*, callback = *callback*

Long Syntax: ATM.160 Timer cancel alarm on net *net_no*, logpoint = *logpoint*, callback address = *callback*

Description: Timer canceled.

ATM.161

Level: UI-ERROR

Short Syntax: ATM.161 Timer SNO, nt *net_no*, logpt = *logpoint*

Long Syntax: ATM.161 Timer should not occur on net *net_no*, logpoint = *logpoint*

Description: Timer element not first, last, or middle.

ATM.162

Level: C-INFO

Short Syntax: ATM.162 Timer tick, nt *net_no*, logpt = *logpoint*, tim = *time*, callback = *callback*

Long Syntax: ATM.162 Timer tick on net *net_no*, logpoint = *logpoint*, time = *time*, callback address = *callback*

Description: Timer tick.

ATM.163

Level: UI-ERROR

Short Syntax: ATM.163 Timer out of elements, nt *net_no*

Long Syntax: ATM.163 Timer out of elements on net *net_no*

Description: Timer out of elements.

ATM.164

Level: UI-ERROR

Short Syntax: ATM.164 Timer tried to free twice, nt *net_no*

Long Syntax: ATM.164 Timer tried to free twice on net *net_no*

Description: Timer tried to free twice.

ATM.165

Level: UI-ERROR

Short Syntax: ATM.165 Tmr elmnts set to max, nt *net_no*

Long Syntax: ATM.165 Number of timer elements capped at maximum on net *net_no*

Description: The number of timer elements would have exceeded the maximum and was capped.

ATM.166

Level: UI-ERROR

Short Syntax: ATM.166 VNET Registration Failed, nt *network id*, rc *stat*

Long Syntax: ATM.166 VNET User Registration Failed, on network *network id*, status = *stat*

Description: atm_vnet_init User Registration Failed

ATM.167

Level: UI-ERROR

Short Syntax: ATM.167 Addr state change, not in API, nt *network ID*, addr *address*, state = *state*

Long Syntax: ATM.167 ATM address state change, not in API, on network *network ID*, ATM address = *address*, state= *state*,

Description: Address state change but API has no record of it.

ATM.168

Level: UI-ERROR

Short Syntax: ATM.168 Addr in use, but not in API, nt *network ID*, addr *address*

Long Syntax: ATM.168 ATM address in use for ILMI, but not API, on network *network ID*, ATM address = *address*

Description: ILMI thinks address is registered but API has no record of it.

ATM.169

Level: UI-ERROR

Short Syntax: ATM.169 No bfr to splice VCC, nt *network ID*, *vpi1- vci1* to *vpi2- vci2*

Long Syntax: ATM.169 No buffer to splice VCC on network *network ID*, *vpi1- vci1* to VCC *vpi2- vci2*

Description: No buffer was available to splice two VCCs.

ATM.170

Level: UI-ERROR

Short Syntax: ATM.170 nt *network ID*, VCC *vpi1- vci1* was spliced *vpi2- vci2*, now *vpi3- vci3*

Long Syntax: ATM.170 on network *network ID*, VCC *vpi1- vci1* was spliced to *vpi2- vci2*, now spliced to *vpi3- vci3*

Description: User spliced an already spliced VCC to a different VCC.

ATM.171

Level: UE-ERROR

Short Syntax: ATM.171 nt *network ID*, no rsp to Restart

Long Syntax: ATM.171 Switch never responded to Restart on net *network ID*

Description: The ATM switch never responded to Restart with RestartAck.

ATM.172

Level: UI-ERROR

Short Syntax: ATM.172 Transmit msg got VCC handle, nt *network ID*

Long Syntax: ATM.172 Transmit message was passed NULL VCC handle on network *network ID*

Description: User called xmit_msg passing NULL VCC handle.

ATM.173

Level: UI-ERROR

Short Syntax: ATM.173 nt *network ID*, *cmd* failed, unsupported protocol: *prt*

Long Syntax: ATM.173 on network *network ID*, *cmd* failed, unsupported protocol: *prt*

Description: An internal routine attempted to add or delete a multicast address for an unsupported protocol.

ATM.174

Level: UI-ERROR

Short Syntax: ATM.174 nt *network ID*, *cmd* failed, protocol *prt*, error code: *err*

Long Syntax: ATM.174 on network *network ID*, *cmd* failed, protocol *prt*, error code: *err*

Description: An internal error occured while attempted to add or remove a multicast address.

ATM.175

Level: UI-ERROR

Short Syntax: ATM.175 nt *network ID*, SVC msg drop, low bfrs: *D2 D3 D4 D5*

Long Syntax: ATM.175 on network *network ID*, SVC message dropped, adapter low on buffers: *D2 D3 D4 D5*

Description: Signaling message dropped because adapter low on buffers.

ATM.176

Level: UI-ERROR

Short Syntax: ATM.176 nt *network ID*, disc conn hndl *caller*, caller *handle* dereg

Long Syntax: ATM.176 on network *network ID*, disconnect call for handle *caller*, caller *handle* already deregistered

Description: Disconnect call received but caller already deregistered

ATM.177

Level: UI-ERROR

Short Syntax: ATM.177 Function *vccmgrHandle* called, bad vccmgr hndl *caller*

Long Syntax: ATM.177 Function *vccmgrHandle* called with invalid vccmgr handle *caller*

Description: VCC manager function called with invalid VCC manager handle

ATM.178

Level: UI-ERROR

Short Syntax: ATM.178 Function *mecHandle* called, bad MEC hndl *caller*

Long Syntax: ATM.178 Function *mecHandle* called with invalid MEC handle *caller*

Description: An invalid MEC handle was found by VCC manager

ATM.179

Level: UI-ERROR

Short Syntax: ATM.179 Function *caller* called, no mem

Long Syntax: ATM.179 Function *caller* called, out of memory

Description: ATM API call failed, no memory available

ATM.181

Level: C-INFO

Short Syntax: ATM.181 VCCMGR, als new entry, vccmgr hndl *vccmgrHandle*

Long Syntax: ATM.181 VCCMGR, new VCC entry created dynamically, vccmgr handle = *vccmgrHandle*

Description: vccmgrGetVCCTableEntry called with a new VCC entry allocated dynamically from system memory

ATM.182

Level: C-INFO

Short Syntax: ATM.182 VCCMGR, new VCC entry, vccmgr hndl *vccmgrHandle*

Long Syntax: ATM.182 VCCMGR, new VCC entry created, vccmgr handle = *vccmgrHandle*

Description: vccmgrGetVCCTableEntry called with a new VCC entry

ATM.183

Level: C-INFO

Short Syntax: ATM.183 VCCMGR, fvs vcc entry, vccmgr hndl *vccmgrHandle*

Long Syntax: ATM.183 VCCMGR, free VCC entry to system, vccmgr handle = *vccmgrHandle*

Description: vccmgrFreeVCCTableEntry called to free VCC entry to system memory

ATM.184

Level: C-INFO

Short Syntax: ATM.184 VCCMGR, free VCC entry, vccmgr hndl *vccmgrHandle*

Long Syntax: ATM.184 VCCMGR, free VCC entry to free list, vccmgr handle = *vccmgrHandle*

Description: vccmgrFreeVCCTableEntry called to free VCC entry to free list

ATM.185

Level: UI-ERROR

Short Syntax: ATM.185 VCCMGR, duplicate entry, pType *pType*, vccmgr hndl *vccmgrHandle*

Long Syntax: ATM.185 VCCMGR, duplicated VCC entry in list, protocol type *pType*, vccmgr handle = *vccmgrHandle*

Description: vccmgrAddVCCToVCCList called for duplicated VCC entry in list

ATM.186

Level: C-INFO

Short Syntax: ATM.186 VCCMGR, add entry, pType *pType*, vccmgr hndl *vccmgrHandle*

Long Syntax: ATM.186 VCCMGR, add VCC entry to VCC list, protocol type *pType*, vccmgr handle = *vccmgrHandle*

Description: vccmgrAddVCCToVCCList called to add VCC entry to the VCC list

ATM.187

Level: C-INFO

Short Syntax: ATM.187 VCCMGR, delete entry, pType *pType*, vccmgr hndl *vccmgrHandle*

Long Syntax: ATM.187 VCCMGR, delete VCC entry from VCC list, protocol type *pType*, vccmgr handle = *vccmgrHandle*

Description: vccmgrDeleteEntryFromVCCList called to delete a VCC entry

ATM.188

Level: C-INFO

Short Syntax: ATM.188 VCCMGR, new MEC entry, mec hndl *mecHandle*

Long Syntax: ATM.188 VCCMGR, new MEC entry created, mec handle = *mecHandle*

Description: vccmgrGetMECTableEntry called with a new MEC entry

ATM.189

Level: C-INFO

Short Syntax: ATM.189 VCCMGR, free MEC entry, mec hndl *mecHandle*

Long Syntax: ATM.189 VCCMGR, free MEC entry, mec handle = *mecHandle*

Description: vccmgrFreeMECTableEntry called to free a MEC entry

ATM.190

Level: UI-ERROR

Short Syntax: ATM.190 VCCMGR, duplicate MEC entry, mec hndl *mecHandle*

Long Syntax: ATM.190 VCCMGR, duplicated entries in MEC list, mec handle = *mecHandle*

Description: vccmgrAddMECEntryToMECList called with duplicated MEC entry

ATM.191

Level: C-INFO

Short Syntax: ATM.191 VCCMGR, add MEC entry, mec hndl *mecHandle*

Long Syntax: ATM.191 VCCMGR, add MEC entry to MEC list, mec handle = *mecHandle*

Description: vccmgrAddMECEntryToMECList called with duplicated MEC entry

ATM.192

Level: C-INFO

Short Syntax: ATM.192 VCCMGR, delete MEC entry, mec hndl *mecHandle*

Long Syntax: ATM.192 VCCMGR, delete MEC entry from MEC list, mec handle = *mecHandle*

Description: vccmgrDeleteMECEntryFromMECList called to delete MEC entry

ATM.193

Level: C-INFO

Short Syntax: ATM.193 VCCMGR, VCC sharing, nt *network ID*, conn hndl *connHandle*

Long Syntax: ATM.193 VCCMGR, place call VCC sharing, on network *network ID*, conn Handle = *connHandle*

Description: vccmgrPlaceCall called with sharing VCC

ATM.194

Level: C-INFO

Short Syntax: ATM.194 VCCMGR, place call ack, vccmgr hndl *vccmgrHandle*, conn hndl *connHandle*

Long Syntax: ATM.194 VCCMGR, place call ack, vccmgr handle = *vccmgrHandle*, conn handle = *connHandle*

Description: vccmgrPlaceCallAck called

ATM.195

Level: C-INFO

Short Syntax: ATM.195 VCCMGR, place call ack for shared vcc, conn hndl *vccmgrHandle*

Long Syntax: ATM.195 VCCMGR, place call ack for shared VCC, conn handle = *vccmgrHandle*

Description: vccmgrPlaceCallAckSimulation called

ATM.196

Level: C-INFO

Short Syntax: ATM.196 VCCMGR, VCC sharing, conn hndl *connHandle*

Long Syntax: ATM.196 VCCMGR, receive call VCC sharing, conn Handle = *connHandle*

Description: vccmgrReceiveCallAck called with sharing VCC

ATM.197

Level: C-INFO

Short Syntax: ATM.197 VCCMGR, frame drop, id *dmuxID* conn hndl *connHandle*

Long Syntax: ATM.197 VCCMGR, data frame dropped, dmux id = *dmuxID*, conn Handle = *connHandle*

Description: vccmgrReceiveFrame called for frame drop

ATM.198

Level: C-INFO

Short Syntax: ATM.198 VCCMGR, ready indicate rcvd, conn hndl *connHandle*

Long Syntax: ATM.198 VCCMGR, first LE ready indicate received, conn Handle = *connHandle*

Description: vccmgrReceiveFrame called for receiving the first ready indicate

ATM.199

Level: C-INFO

Short Syntax: ATM.199 VCCMGR, duplicate VCC found, MEC hndl *mecHandle*, conn hndl *connHandle*

Long Syntax: ATM.199 VCCMGR, duplicate VCC found, MEC handle = *mecHandle*, conn handle = *connHandle*

Description: vccmgrReceiveFrame called for duplicated VCC

ATM.200

Level: C-INFO

Short Syntax: ATM.200 VCCMGR, data frame rcvd, marker *marker*

Long Syntax: ATM.200 VCCMGR, LE data frame received, marker = *marker*

Description: vccmgrReceiveFrame called for receiving data frame

ATM.201

Level: C-INFO

Short Syntax: ATM.201 VCCMGR, control frame rcvd, opcode *opCode*

Long Syntax: ATM.201 VCCMGR, LE control frame received, opCode = *opCode*

Description: vccmgrReceiveFrame called for receiving control frame

ATM.202

Level: C-INFO

Short Syntax: ATM.202 VCCMGR, frame drop, dmuxID *opCode*

Long Syntax: ATM.202 VCCMGR, frame dropped, user not found, dmuxID = *opCode*

Description: vccmgrReceiveFrame called for dropping frame

ATM.203

Level: C-INFO

Short Syntax: ATM.203 VCCMGR, simulating rcvd call, hndl *connHandle*

Long Syntax: ATM.203 VCCMGR, simulating receive call, connHandle = *connHandle*

Description: vccmgrSimulatingReceiveCall called

ATM.204

Level: C-INFO

Short Syntax: ATM.204 VCCMGR, call simulated, hndl *connHandle*

Long Syntax: ATM.204 VCCMGR, receive call simulated, connHandle = *connHandle*

Description: vccmgrSimulatingReceiveCall called

ATM.205

Level: C-INFO

Short Syntax: ATM.205 VCCMGR, share VCC data path, vcc hndl *vccHandle*

Long Syntax: ATM.205 VCCMGR, share VCC data path, vccHandle = *vccHandle*

Description: vccmgrOpenVccDataPath called

ATM.206

Level: C-INFO

Short Syntax: ATM.206 VCCMGR, close shared VCC data path, vcc hndl *vccHandle*

Long Syntax: ATM.206 VCCMGR, close shared VCC data path, vcc handle = *vccHandle*

Description: vccmgrOpenVccDataPath called

ATM.207

Level: C-INFO

Short Syntax: ATM.207 VCCMGR, hangup call by vccmgr, conn hndl *vccHandle*

Long Syntax: ATM.207 VCCMGR, hangup call by vccmgr, conn handle = *vccHandle*

Description: vccmgrReleaseVCC called to hangup call

ATM.208

Level: C-INFO

Short Syntax: ATM.208 VCCMGR, nt *network ID*, hangup call, conn hndl *connHandle*

Long Syntax: ATM.208 VCCMGR, network *network ID*, VCC hangup call, connHandle = *connHandle*

Description: vccmgrHangupCall called

ATM.209

Level: C-INFO

Short Syntax: ATM.209 VCCMGR, nt *network ID*, hangup shared call, conn hndl *connHandle*

Long Syntax: ATM.209 VCCMGR, network *network ID*, hangup shared VCC call, connHandle = *connHandle*

Description: vccmgrHangupCall called

ATM.210

Level: C-INFO

Short Syntax: ATM.210 VCCMGR, new owner, vccmgr hndl *vccmgrHandle*, conn hndl *connHandle*

Long Syntax: ATM.210 VCCMGR, new VCC owner found, vccmgrHandle = *vccmgrHandle*, connHandle = *connHandle*

Description: vccmgrSetNewVccOwner called

ATM.211

Level: C-INFO

Short Syntax: ATM.211 VCCMGR, new owner, vccmgr hndl *vccmgrHandle*, conn hndl *connHandle*

Long Syntax: ATM.211 VCCMGR, new VCC owner found, vccmgrHandle = *vccmgrHandle*, connHandle = *connHandle*

Description: vccmgrDisconnectCallSharedUser called

ATM.212

Level: C-INFO

Short Syntax: ATM.212 VCCMGR, disconnect call, vccmgr hndl *vccmgrHandle*, conn hndl *connHandle*

Long Syntax: ATM.212 VCCMGR, disconnect call, vccmgr handle = *vccmgrHandle*, connHandle = *connHandle*

Description: vccmgrDisconnectCall called

ATM.213

Level: C-INFO

Short Syntax: ATM.213 VCCMGR, close call sap, sap hndl *vccmgrHandle*

Long Syntax: ATM.213 VCCMGR, close call sap, sap handle = *vccmgrHandle*

Description: vccmgrCloseCallSap called

ATM.214

Level: C-INFO

Short Syntax: ATM.214 VCCMGR, disconnect shared call, conn hndl *connHandle*

Long Syntax: ATM.214 VCCMGR, disconnect call for shared users, connHandle = *connHandle*

Description: vccmgrDisconnectCallSharedUser called

ATM.215

Level: C-INFO

Short Syntax: ATM.215 VCCMGR, simulating hangup call, conn hndl *connHandle*

Long Syntax: ATM.215 VCCMGR, simulating hangup call for shared users, connHandle = *connHandle*

Description: vccmgrHangupCallSimulation called

ATM.216

Level: UI-ERROR

Short Syntax: ATM.216 API, invalid bound VCC hndl, nt *network ID*, hndl *handle* bnd hdnl *bound*

Long Syntax: ATM.216 API called with invalid VCC handle, on network *network ID*, handle = *handle* bound handle = *bound*

Description: Invalid bound VCC handle in VCC handle passed to API

ATM.217

Level: UI-ERROR

Short Syntax: ATM.217 C_caller, hangup leaf: dup caller on MP call, nt *network ID*, conn hndl *connHandle*

Long Syntax: ATM.217 C_caller, hangup leaf: duplicate caller on multipoint call, on network *network ID*, conn handle = *connHandle*

Description: Add-party received for a multipoint call for the same caller

ATM.218

Level: U-INFO

Short Syntax: ATM.218 Function *function_name*: Grp-snd cncld nt *network ID* group *groupHandle*

Long Syntax: ATM.218 Function *function_name*: Group-send canceled on network *network ID* group *groupHandle*

Description: Group list(s) was modified during group-send (canceled).

ATM.219

Level: CI-ERROR

Short Syntax: ATM.219 Function *function_name*: Grp not in use nt *network ID* iorb *iorbp*

Long Syntax: ATM.219 Function *function_name*: Group not in use on network *network ID* iorb *iorbp*

Description: Sending on a group and the group status was not set properly.

Panic atmmem

Short Syntax: ATM interface initialization failed, no memory.

Description: The ATM interface failed to allocate sufficient memory to complete initialization.

Action: Contact your customer service representative.

Chapter 9. ATM Interim Local Management Interface (ILMI)

This chapter describes ATM Interim Local Management Interface (ILMI) messages. For information on message content and how to use the message, refer to the Introduction.

ILMI.001

Level: C-INFO

Short Syntax: ILMI.001 nt *net_num* state chng *state*

Long Syntax: ILMI.001 Network *net_num* state changed to: *state*

Description: ILMI state changed.

ILMI.002

Level: C-INFO

Short Syntax: ILMI.002 nt *net_num* ntrd func *function_name*

Long Syntax: ILMI.002 Network *net_num*, entered function *function_name*

Description: ILMI function entered, no trace arguments.

ILMI.003

Level: C-INFO

Short Syntax: ILMI.003 nt *net_num* ntrd func *function_name*, state= *state*

Long Syntax: ILMI.003 Network *net_num*, entered, function *function_name*, state = *state*

Description: ILMI function entered, with ILMI state as an argument.

ILMI.004

Level: C-INFO

Short Syntax: ILMI.004 nt *net_num* ntrd func *function_name value*

Long Syntax: ILMI.004 Network *net_num*, entered function *function_name value*

Description: ILMI function entered, with value x as an argument.

ILMI.005

Level: C-INFO

Short Syntax: ILMI.005 nt *net_num* ntrd func *function_name*, *val1*, *val2*

Long Syntax: ILMI.005 Network *net_num*, entered function *function_name*, *val1*, *val2*

Description: ILMI function entered, with values x and y as arguments.

ILMI.006

Level: P_TRACE

Short Syntax: ILMI.006 Trace ATM ILMI frame.

Long Syntax: ILMI.006 Trace ATM ILMI frame.

Description: ATM ILMI frame packet tracing.

ILMI.007

Level: C-INFO

Short Syntax: ILMI.007 nt *net_num* ntrd func *function_name*,state= *state*,hndl= *info*

Long Syntax: ILMI.007 Network *net_num*, entered function *function_name*, state = *state*, handle = *info*

Description: ILMI function entered, with ILMI state and address handle as arguments.

ILMI.008

Level: C-INFO

Short Syntax: ILMI.008 nt *net_num* recv *cmd_type*

Long Syntax: ILMI.008 Network *net_num*, received a *cmd_type*

Description: ILMI data received with command type.

ILMI.009

Level: UE-ERROR

Short Syntax: ILMI.009 nt *net_num els_msg*, state= *state*

Long Syntax: ILMI.009 Network *net_num*, *els_msg*, state= *state*

Description: ILMI log point of external error with ILMI state.

ILMI.010

Level: UI-ERROR

Short Syntax: ILMI.010 nt *net_num els_msg*, state= *state*

Long Syntax: ILMI.010 Network *net_num*, *els_msg*, state= *state*

Description: ILMI log point of internal error with ILMI state.

ILMI.011

Level: C-INFO

Short Syntax: ILMI.011 nt *net_num els_msg*, state= *state*

Long Syntax: ILMI.011 Network *net_num*, *els_msg*, state= *state*

Description: ILMI log point of information with ILMI state.

ILMI.012

Level: UE-ERROR

Short Syntax: ILMI.012 nt *net_num els_msg*, *info*

Long Syntax: ILMI.012 Network *net_num*, *els_msg info*

Description: ILMI log point of external error with more data.

ILMI.013

Level: UE-ERROR

Short Syntax: ILMI.013 nt *net_num els_msg*

Long Syntax: ILMI.013 Network *net_num*, *els_msg*

Description: ILMI log point of external error with no data.

ILMI.014

Level: UI-ERROR

Short Syntax: ILMI.014 nt *net_num els_msg*

Long Syntax: ILMI.014 Network *net_num*, *els_msg*

Description: ILMI log point of internal error with no data.

ILMI.015

Level: C-INFO

Short Syntax: ILMI.015 nt *net_num* ntrd func *function_name state*, state= *info*

Long Syntax: ILMI.015 Network *net_num*, entered function *function_name state*, state = *info*

Description: ILMI function entered, with value y and ILMI state as arguments.

ILMI.016

Level: C-INFO

Short Syntax: ILMI.016 nt *net_num els_msg value*

Long Syntax: ILMI.016 Network *net_num*, *els_msg value*

Description: ILMI log point of information with value.

ILMI.017

Level: C-INFO

Short Syntax: ILMI.017 nt *net_num els_msg*

Long Syntax: ILMI.017 Network *net_num*, *els_msg*

Description: ILMI log point of information with no data.

ILMI.018

Level: UI-ERROR

Short Syntax: ILMI.018 nt *net_num els_msg*, *value*

Long Syntax: ILMI.018 Network *net_num*, *els_msg value*

Description: ILMI log point of internal error with more data.

ILMI.019

Level: C-INFO

Short Syntax: ILMI.019 nt *net_num els_msg*, *val1*, *val2*

Long Syntax: ILMI.019 Network *net_num*, *els_msg*, *val1*, *val2*

Description: ILMI log point of information with two values.

ILMI.020

Level: C-INFO

Short Syntax: ILMI.020 nt *net_num* snt *cmd_type*

Long Syntax: ILMI.020 Network *net_num*, sent a *cmd_type*

Description: ILMI data sent with command type.

ILMI.021

Level: C-INFO

Short Syntax: ILMI.021 nt *net_num* net pref= *addr1 addr2 addr3 addr4*

Long Syntax: ILMI.021 Network *net_num*, network prefix= *addr1 addr2 addr3 addr4*

Description: ILMI received the network prefix from the switch.

ILMI.022

Level: C-INFO

Short Syntax: ILMI.022 nt *net_num* ntrd func *function_name*, *version*

Long Syntax: ILMI.022 Network *net_num*, entered, function *function_name*, UNI version = *version*

Description: ILMI returned UNI version

ILMI.023

Level: C-INFO

Short Syntax: ILMI.023 Reg ESI, nt *net_num*, func *function_name*, addr= *addr1 addr2*,sel= *sel*

Long Syntax: ILMI.023 Registering ESI on Network *net_num*, function *function_name*, ESI= *addr1 addr2*, Selector = *sel*

Description: ILMI registering ESI with Selector

Chapter 10. ATM LLC (ALLC)

This chapter describes ATM LLC (ALLC) messages. For information on message content and how to use the message, refer to the Introduction.

ALLC.001

Level: C-INFO

Short Syntax: ALLC.001 Fn *function_name* called, nt *network id*

Long Syntax: ALLC.001 Function *function_name* called, on network *network id*

Description: ATM 1483 function called

ALLC.002

Level: CI-ERROR

Short Syntax: ALLC.002 error rtn, fn *function_name* rc *return_code* (*descriptor_string*) nt *network ID*

Long Syntax: ALLC.002 error rtn, fn *function_name* rc *return_code* (*descriptor_string*) nt *network ID*

Description: A downcall to the ATM LLC layer returned error

Cause: Could be bad input parameters, or an erroneous condition the return code will be printed alongwith a short string describing the error. The function name returning error is also printed. Possible error strings: ''Invalid net num'' : Invalid net number was passed to the API ''NULL clientFunctions'' : NULL clientFunctions ptr passed ''NULL clientHandlePtr'' : NULL clientHandlePtr passed ''Inv hdr length'' : Invalid packet header length passed ''Invalid addr scheme'' : Invalid ATM addressing scheme requested ''net down (reg)'' : Net is down, but client was registered (no error) ''net down (no reg)'' : Net is down, no client registration performed (no error) ''dup non-shared EPs'' : Two non shareable endpoints configured with the same atm address tried to register. (This indicates a possible configuration error) ''addr actvn procdng'' : ATM address activation is in progress (no error) ''bad client handle'' : An invalid client handle was passed to the ATM LLC API. ''bad input parms'' : An invalid input parameter was passed to the ATM LLC API. ''ep not up'' : A caller tried to open a channel on an endpoint which is not yet up. ''bad channel handle'' : An invalid channel handle was passed to the ATM LLC API. ''not chnl user'' : A caller to the ATM LLC API tried to use a channel without being a user of it.

ALLC.003

Level: C-INFO

Short Syntax: ALLC.003 clnt (*atm1483ClientStruct_ptr*) added to exstng EP (*atm1483EpBlkStruct_ptr*) (total *num_clnts*), nt *network id*

Long Syntax: ALLC.003 client (*atm1483ClientStruct_ptr*) added to existing EP (*atm1483EpBlkStruct_ptr*) (total *num_clnts*), nt *network id*

Description: A client has been successfully registered with an existing point, the new number of clients on this endpoint is printed.

ALLC.004

Level: C-INFO

Short Syntax: ALLC.004 Clnt (*atm1483ClientStruct_ptr*) added to new EP (*atm1483EpBlkStruct_ptr*), nt *network id*

Long Syntax: ALLC.004 Client (*atm1483ClientStruct_ptr*) added to newly created endpoint (*atm1483EpBlkStruct_ptr*), nt *network id*

Description: A new endpoint has been created and a client has successfully registered with it.

ALLC.005

Level: C-INFO

Short Syntax: ALLC.005 Fn *function_name* rtng SUCCESS, nt *network id*

Long Syntax: ALLC.005 Function *function_name* returning SUCCESS, on network *network id*

Description: ATM 1483 function returning SUCCESS

ALLC.006

Level: C-INFO

Short Syntax: ALLC.006 Fn *function_name* called

Long Syntax: ALLC.006 Function *function_name* called

Description: ATM 1483 function called

ALLC.007

Level: CI-ERROR

Short Syntax: ALLC.007 Error rtn, fn *function_name* rc

return_code (*descriptor_string*)

Long Syntax: ALLC.007 Error rtn, fn *function_name* rc *return_code* (*descriptor_string*)

Description: A downcall to the ATM LLC layer returned error

Cause: Could be bad input parameters, or an erroneous condition the return code will be printed alongwith a short string describing the error. Possible descriptor_strings are the same as in ALLC_2.

ALLC.008

Level: UI_ERROR

Short Syntax: ALLC.008 Failed mem allocn, fn *function_name* (*descriptor_string*)

Long Syntax: ALLC.008 Failed memory allocation in function *function_name* (*descriptor_string*)

Description: An attempt to allocate dynamic memory failed

Cause: This indicates that the router is running out of dynamic memory This should be addressed the same way that other memory allocation failures are addressed.

ALLC.009

Level: UI_ERROR

Short Syntax: ALLC.009 Matching PVC (vpi= *vpi* vci= *vci*) unusable (*descriptor_string*), nt *network id*

Long Syntax: ALLC.009 matching PVC (vpi= *vpi* vci= *vci*) unusable (*descriptor_string*), nt *network id*

Description: A client tried to open an existing PVC but sharing flags do not allow this or there is an SDU mismatch (this is indicated in the descriptor string).

Cause: This may indicate a configuration error for these PVCs.

ALLC.010

Level: C-INFO

Short Syntax: ALLC.010 new user on chnl (vpi *vpi*/ vci *vci*) (total *new_total*) nt *network id*

Long Syntax: ALLC.010 Added new user to chnl (vpi *vpi*/ vci *vci*) (total *new_total*) nt *network id*

Description: A new user has been added to a vcc. The new total number of users of this vcc is printed.

ALLC.011

Level: C-INFO

Short Syntax: ALLC.011 PVC up (vpi= *vpi* vci= *vci*), nt *network id*

Long Syntax: ALLC.011 New PVC activated (vpi= *vpi* vci= *vci*), nt *network id*

Description: A new PVC has been activated

ALLC.012

Level: UI-ERROR

Short Syntax: ALLC.012 Failed PVC bring up (vpi= *vpi* vci= *vci*, redial= *redial_flag*), nt *network id*

Long Syntax: ALLC.012 Failed PVC bring up (vpi= *vpi* vci= *vci*, redial= *redial_flag*), nt *network id*

Description: Indicates failure to bring up a PVC. redial if non zero indicates subsequent activation will be re-attempted.

ALLC.013

Level: CE-ERROR

Short Syntax: ALLC.013 Failed SVC bring up (dstn *atm_address*, redial= *redial_flag*), nt *network id*

Long Syntax: ALLC.013 Failed SVC bring up (dstn ATM addr *atm_address*, redial= *redial_flag*), nt *network id*

Description: Indicates failure to bring up a SVC to specified destination ATM address. ''redial'' being non zero indicates subsequent activation will be re-attempted.

ALLC.014

Level: C-INFO

Short Syntax: ALLC.014 SVC call placed (dstn *atm_address*), nt *network id*

Long Syntax: ALLC.014 SVC call placed (dstn ATM addr *atm_address*), nt *network id*

Description: Indicates that a call was successfully placed for an SVC to the specified destination atm address.

ALLC.016

Level: C-INFO

Short Syntax: ALLC.016 SVC hung up (vpi *vpi*/ vci *vci*, dstn *atm_address*) nt *network id*

Long Syntax: ALLC.016 SVC hung up (vpi *vpi*/ vci *vci*, dstn *atm_address*) nt *network id*

Description: Indicates that an SVC was hung up after the last user of this SVC closed this channel.

ALLC.017

Level: C-INFO

Short Syntax: ALLC.017 atm1483SendData success (vpi *vpi*/ vci *vci*), nt *network id*

Long Syntax: ALLC.017 atm1483SendData success(vpi *vpi*/ vci *vci*), nt *network id*

Description: The ″slow path″ data transmission function on this vcc was successful

---

ALLC.018

Level: UI-ERROR

Short Syntax: ALLC.018 atm1483SendData failed (vpi *vpi*/ vci *vci*), nt *network id*

Long Syntax: ALLC.018 atm1483SendData failed (vpi *vpi*/ vci *vci*), nt *network id*

Description: The ″slow path″ data transmission function vcc was unsuccessful

---

ALLC.019

Level: C-INFO

Short Syntax: ALLC.019 EP actvn attempt (ESI *esi* Sel *selector*), nt *network id*

Long Syntax: ALLC.019 EP activation attempt (ESI *esi* Sel *selector*), nt *network id*

Description: An attempt is being made to activate an endpoint. The endpoint is described by its ESI (either actual or ″Burned In″ and the selector byte).

ALLC.020

Level: CI-ERROR

Short Syntax: ALLC.020 ATM downcall fail (*function_name* rc *return_code*)

Long Syntax: ALLC.020 ATM downcall from 1483 failed (*function_name* rc *return_code*)

Description: A downcall from the 1483 layer to the ATM driver returned something other than SUCCESS, the function name and return code are printed.

ALLC.021

Level: C-INFO

Short Syntax: ALLC.021 EP up (*atm_address*), nt *network id*

Long Syntax: ALLC.021 Local endpoint activated (*atm_address*), nt *network id*

Description: A local ATM 1483 endpoint has been activated

ALLC.022

Level: C-INFO

Short Syntax: ALLC.022 ATM addr state chg (ESI *esi* Sel *selector*, state *newstate*), nt *network id*

Long Syntax: ALLC.022 ATM addr state chg upcall (ESI *esi* Sel *selector*, state *newstate*), nt *network id*

Description: An upcall was received indicating a change in state of an ATM address The new state is indicated by the value of ″state″ state = 0 => address deactivated state = 1 => address activated state = 2 => address refused state = 3 => address wrap

---

ALLC.023

Level: UE-ERROR

Short Syntax: ALLC.023 Hanging up incoming call (caller *remote_atm_address*, *hangup_descriptor_string*), nt *network id*

Long Syntax: ALLC.023 Hanging up incoming call (caller *remote_atm_address*, *hangup_descriptor_string*), nt *network id*

Description: An incoming call has been hung up. Caller's atm address and a string describing the reason are printed

---

ALLC.024

Level: C-INFO

Short Syntax: ALLC.024 Valid call recvd (caller *remote_atm_address*, vpi *vpi*/ vci *vci*, total *num_chnls*, sdu chg *sdu_chg*)

Long Syntax: ALLC.024 Valid call recvd (caller *remote_atm_address*, vpi *vpi*/ vci *vci*, total *num_chnls*, sdu chg *sdu_chg*)

Description: An valid incoming call has been received and acknowledged. In addition to the caller's atm address, vpi and vci, the new total number of active SVCs on this interface (using ALLC) is printed and whether SDU negotiation was performed (if ″sdu chg″ is non zero).

ALLC.025

Level: UE-ERROR

Short Syntax: ALLC.025 Hanging up acked call (destn *remote_atm_address*, *hangup_descriptor_string*, redial *redial_flag*), nt *network id*

Long Syntax: ALLC.025 Hanging up acked call (destn *remote_atm_address*, *hangup_descriptor_string*, redial *redial_flag*), nt *network id*

Description: An outgoing call which has received an ack has been hung up. Destination atm address and a string describing the reason are printed. If ″redial″ flag is non zero, indicates that the channel will be redialled.

ALLC.026

Level: C-INFO

Short Syntax: ALLC.026 VCC setup complete (destn *remote_atm_address*, vpi *vpi*/ vci *vci*, total *num_chnls*), nt *network id*

Long Syntax: ALLC.026 VCC setup complete (destn *remote_atm_address*, vpi *vpi*/ vci *vci*, total *num_chnls*), nt *network id*

Description: A VCC which was initiated by the local endstation has been successfully setup. The destination ATM address, vpi, vci are printed as well as the new total number of active SVCs on this interface (using ALLC).

ALLC.027

Level: CE-ERROR

Short Syntax: ALLC.027 Recvd remote discon (from *remote_atm_address*, vpi *vpi*/vci *vci*), nt *network id*

Long Syntax: ALLC.027 Recvd remote discon (from *remote_atm_address*, vpi *vpi*/vci *vci*), nt *network id*

Description: The remote ATM station disconnected the VCC. The remote ATM address and vpi/vci of the VCC are printed.

ALLC.028

Level: CE-ERROR

Short Syntax: ALLC.028 Remote VCC Disconn (rsn *reason_code*, cause *cause_code*, diagLen *diag_len*, diagData[0] *diag_data*)

Long Syntax: ALLC.028 Remote VCC Disconn (rsn *reason_code*, cause *cause_code*, diagLen *diag_len*, diagData[0] *diag_data*)

Description: This is used to indicate the reason and cause codes for the VCC disconnect indicated by ALLC_27

ALLC.029

Level: C-INFO

Short Syntax: ALLC.029 EP cleaned up (ESI *esi* Sel *selector*), nt *network id*

Long Syntax: ALLC.029 EP cleaned up (ESI *esi* Sel *selector*), nt *network id*

Description: An Endpoint which is no longer needed is being cleaned up. The ESI and Selector byte which define the endpoint are displayed.

ALLC.030

Level: UI_ERROR

Short Syntax: ALLC.030 Internal ATM downcall fail (*function_name* rc *return_code*)

Long Syntax: ALLC.030 ATM downcall for a local function from 1483 failed (*function_name* rc *return_code*)

Description: A downcall to the ATM API which only involved local functions and should normally always succeed, failed. This indicates possibly an error in the ATMLLC and/or ATM driver code.

ALLC.031

Level: C-INFO

Short Syntax: ALLC.031 Packet recvd (0-3 *first_4_bytes* 4-7 *next_4_bytes* 8-9 *next_2_bytes*), nt *network id*

Long Syntax: ALLC.031 Packet received by ATMLLC (0-3 *first_4_bytes* 4-7 *next_4_bytes* 8-9 *next_2_bytes*), nt *network id*

Description: A packet has been received by the ATMLLC module on this net; the first 10 bytes of the packet (containing LLC SNAP information) are printed.

ALLC.032

Level: UI-ERROR

Short Syntax: ALLC.032 invalid clnt (*atm1483ClientStruct_ptr*) on chnl (vpi *vpi*/vci *vci*), nt *network id*

Long Syntax: ALLC.032 invalid client (*atm1483ClientStruct_ptr*) still on chnl (vpi *vpi*/vci *vci*), nt *network id*

Description: An invalid client is listed as a user of a channel. This could have been an old client which has since been deleted or which exitted this channel earlier. This indicates an internal coding error.

ALLC.033

Level: U-INFO

Short Syntax: ALLC.033 chnl disconn recvd with null correlator

Long Syntax: ALLC.033 chnl disconn recvd with null correlator

Description: The ATMLLC module received a disconnect for a channel before receiving the setup itself.

ALLC.034

Level: C-INFO

Short Syntax: ALLC.034 netup recvd by ep *atm1483EpBlkStruct_ptr* (*num_clients* clnts), nt *network id*

Long Syntax: ALLC.034 netup recvd by endpoint *atm1483EpBlkStruct_ptr* (*num_clients* clients), nt *network id*

Description: A netup was received by the endpoint specified. The number of clients currently registered with the endpoint is printed.

ALLC.035

Level: C-INFO

Short Syntax: ALLC.035 netdwn recvd by ep *atm1483EpBlkStruct_ptr* (*num_clients* clnts), nt *network id*

Long Syntax: ALLC.035 netdown recvd by endpoint *atm1483EpBlkStruct_ptr* (*num_clients* clients), nt *network id*

Description: A net down was received by the endpoint specified. The number of clients currently registered with the endpoint is printed.

ALLC.036

Level: C-INFO

Short Syntax: ALLC.036 clnt (*atm1483ClientStruct_ptr*) deleted from ep (*atm1483EpBlkStruct_ptr*), *n_clients* remain, nt *network id*

Long Syntax: ALLC.036 client (*atm1483ClientStruct_ptr*) deleted from endpt (*atm1483EpBlkStruct_ptr*), *n_clients* remain, nt *network id*

Description: This message is printed whenever a client is deregistered from an endpoint. IDs of the client and endpoint are printed as well as the number of clients remaining on the endpoint.

Chapter 11. ATM Signalling (SVC)

This chapter describes ATM Signalling (SVC) messages. For information on message content and how to use the message, refer to the Introduction.

SVC.001

Level: C-INFO

Short Syntax: SVC.001 *LOGATM_STRING*

Long Syntax: SVC.001 *LOGATM_STRING*

Description: generic information log.

SVC.002

Level: C-INFO

Short Syntax: SVC.002 *LOGATM_STRING D2*

Long Syntax: SVC.002 *LOGATM_STRING D2*

Description: generic information log with one argument.

SVC.003

Level: C-INFO

Short Syntax: SVC.003 *LOGATM_STRING, D2, D3*

Long Syntax: SVC.003 *LOGATM_STRING, D2, D3*

Description: generic information log for two arguments.

SVC.004

Level: C-INFO

Short Syntax: SVC.004 *LOGATM_STRING, D2, D3, D4*

Long Syntax: SVC.004 *LOGATM_STRING, D2, D3, D4*

Description: generic information log with three arguments.

SVC.005

Level: UI-ERROR

Short Syntax: SVC.005 *LOGATM_STRING*

Long Syntax: SVC.005 *LOGATM_STRING*

Description: generic internal error log.

SVC.006

Level: UI-ERROR

Short Syntax: SVC.006 *LOGATM_STRING D2*

Long Syntax: SVC.006 *LOGATM_STRING D2*

Description: generic internal error log with one argument.

SVC.007

Level: UI-ERROR

Short Syntax: SVC.007 *LOGATM_STRING D2 D3*

Long Syntax: SVC.007 *LOGATM_STRING D2 D3*

Description: generic internal error log with two arguments.

SVC.008

Level: UI-ERROR

Short Syntax: SVC.008 *LOGATM_STRING D2 D3 D4*

Long Syntax: SVC.008 *LOGATM_STRING D2 D3 D4*

Description: generic internal error log with three arguments.

SVC.009

Level: UE-ERROR

Short Syntax: SVC.009 *LOGATM_STRING*

Long Syntax: SVC.009 *LOGATM_STRING*

Description: generic external error log.

SVC.010

Level: UE-ERROR

Short Syntax: SVC.010 *LOGATM_STRING D2*

Long Syntax: SVC.010 *LOGATM_STRING D2*

Description: generic external error log with one argument.

SVC.011

Level: UE-ERROR

Short Syntax: SVC.011 *LOGATM_STRING, D2, D3*

Long Syntax: SVC.011 *LOGATM_STRING, D2, D3*

Description: generic external error log with two arguments.

SVC.012

Level: C-INFO

Short Syntax: SVC.012 Enter *LOGATM_STRING*

Long Syntax: SVC.012 Entered function *LOGATM_STRING*

Description: SVC function entered

SVC.013

Level: C-INFO

Short Syntax: SVC.013 Enter function *LOGATM_STRING D2*

Long Syntax: SVC.013 Entered function *LOGATM_STRING D2*

Description: SVC function entered, with one argument.

SVC.014

Level: C-INFO

Short Syntax: SVC.014 Enter function *LOGATM_STRING*, *D2*, *D3*

Long Syntax: SVC.014 Entered function *LOGATM_STRING*, *D2*, *D3*

Description: SVC function entered, with two arguments.

SVC.015

Level: C-INFO

Short Syntax: SVC.015 Enter function *LOGATM_STRING*, *D2*, *D3*, *D4*

Long Syntax: SVC.015 Entered function *LOGATM_STRING*, *D2*, *D3*, *D4*

Description: SVC function entered, with three arguments.

SVC.016

Level: C-INFO

Short Syntax: SVC.016 Exit *LOGATM_STRING*

Long Syntax: SVC.016 Exited Function *LOGATM_STRING*

Description: SVC function exited

SVC.017

Level: C-INFO

Short Syntax: SVC.017 Exit *LOGATM_STRING D2*

Long Syntax: SVC.017 Exited Function *LOGATM_STRING D2*

Description: SVC function exited, with one argument

SVC.018

Level: C-INFO

Short Syntax: SVC.018 Exit *LOGATM_STRING*, *D2*, *D3*

Long Syntax: SVC.018 Exited Function *LOGATM_STRING D2 D3*

Description: SVC function exited, with two arguments

SVC.019

Level: C-INFO

Short Syntax: SVC.019 Exit *LOGATM_STRING*, *D2*, *D3*, *D4*

Long Syntax: SVC.019 Exited Function *LOGATM_STRING D2 D3 D4*

Description: SVC function exited, with three arguments

SVC.020

Level: C-INFO

Short Syntax: SVC.020 Received signalling message *LOGATM_STRING*,conn hndl= *D2*,ID= *D3*,state= *D4*

Long Syntax: SVC.020 Received signalling message *LOGATM_STRING*, conn handle = *D2*, Call Ref ID = *D3*, call state = *D4*

Description: Signalling message received.

SVC.021

Level: UE-ERROR

Short Syntax: SVC.021 Timer *LOGATM_STRING* expired, conn hndl= *D2*,leaf hndl= *D3*

Long Syntax: SVC.021 Timer *LOGATM_STRING* expired, conn handle = *D2*, leaf handle = *D3*

Description: Timer expired.

SVC.022

Level: P_TRACE

Short Syntax: SVC.022 Trace ATM SVC frame.

Long Syntax: SVC.022 Trace ATM SVC frame.

Description: ATM SVC frame packet tracing

SVC.023

Level: UE-ERROR

Short Syntax: SVC.023 Timer *LOGATM_STRING* expired

Long Syntax: SVC.023 Timer *LOGATM_STRING* expired

Description: Timer expired.

SVC.024

Level: C-INFO

Short Syntax: SVC.024 Received signalling message, *LOGATM_STRING* type= *D2*

Long Syntax: SVC.024 Received signalling message, *LOGATM_STRING* type = *D2*

Description: Signalling message received.

SVC.025

Level: C-INFO

Short Syntax: SVC.025 *LOGATM_STRING D2 D3 D4 D5*

Long Syntax: SVC.025 *LOGATM_STRING D2 D3 D4 D5*

Description: generic information log with string argument.

SVC.026

Level: UE-ERROR

Short Syntax: SVC.026 *LOGATM_STRING*, *D2*, *D3*, *D4*

Long Syntax: SVC.026 *LOGATM_STRING*, *D2*, *D3*, *D4*

Description: generic external error log with three arguments.

SVC.027

Level: UI-ERROR

Short Syntax: SVC.027 *LOGATM_STRING*

Long Syntax: SVC.027 *LOGATM_STRING*

Description: Call clearing error

Chapter 12. ATM Signalling ATM Adaptation Layer (SAAL)

This chapter describes ATM Signalling ATM Adaptation Layer (SAAL) messages. For information on message content and how to use the message, refer to the Introduction.

SAAL.001

Level: C-INFO

Short Syntax: SAAL.001 nt *n_net* Function *LOGATM_STRING* entered

Long Syntax: SAAL.001 Net *n_net* Function *LOGATM_STRING* entered

Description: SAAL function entered

SAAL.002

Level: C-INFO

Short Syntax: SAAL.002 nt *n_net* Function *LOGATM_STRING* extd

Long Syntax: SAAL.002 Net *n_net* Function *LOGATM_STRING* exited

Description: SAAL function exited

SAAL.003

Level: UI-ERROR

Short Syntax: SAAL.003 nt *n_net LOGATM_STRING*

Long Syntax: SAAL.003 Net *n_net LOGATM_STRING*

Description: SAAL internal error

SAAL.004

Level: C-INFO

Short Syntax: SAAL.004 nt *n_net* SSCF state change, *LOGATM_STRING D2*

Long Syntax: SAAL.004 Net *n_net* SSCF state change, *LOGATM_STRING D2*

Description: SSCF state change

SAAL.005

Level: C-INFO

Short Syntax: SAAL.005 nt *n_net* SSCF *LOGATM_STRING D2*

Long Syntax: SAAL.005 Net *n_net* SSCF *LOGATM_STRING D2*

Description: SSCF state change with one arg

SAAL.006

Level: C-INFO

Short Syntax: SAAL.006 nt *n_net LOGATM_STRING*

Long Syntax: SAAL.006 Net *n_net LOGATM_STRING*

Description: SSCF transmit packet

SAAL.007

Level: C-INFO

Short Syntax: SAAL.007 nt *n_net LOGATM_STRING*

Long Syntax: SAAL.007 Net *n_net LOGATM_STRING*

Description: SSCF receive packet

SAAL.008

Level: UI-ERROR

Short Syntax: SAAL.008 nt *n_net LOGATM_STRING*

Long Syntax: SAAL.008 Net *n_net LOGATM_STRING*

Description: SSCF internal error

SAAL.009

Level: UI-ERROR

Short Syntax: SAAL.009 nt *n_net* SSCF state change *LOGATM_STRING D2*

Long Syntax: SAAL.009 Net *n_net* SSCF state change *LOGATM_STRING D2*

Description: SSCF unusual state change

SAAL.010

Level: C-INFO

Short Syntax: SAAL.010 nt *n_net* SSCOP state change, *LOGATM_STRING D2*

Long Syntax: SAAL.010 Net *n_net* SSCOP state change, *LOGATM_STRING D2*

Description: SSCOP state change

SAAL.011

Level: UE-ERROR

Short Syntax: SAAL.011 nt *n_net LOGATM_STRING*

Long Syntax: SAAL.011 Net *n_net LOGATM_STRING*

Description: SSCF external error log

SAAL.012

Level: UE-ERROR

Short Syntax: SAAL.012 nt *n_net LOGATM_STRING D2*

Long Syntax: SAAL.012 Net *n_net LOGATM_STRING D2*

Description: SSCF external error log with one arg

SAAL.013

Level: UI-ERROR

Short Syntax: SAAL.013 nt *n_net* SSCOP state change *LOGATM_STRING D2*

Long Syntax: SAAL.013 Net *n_net* SSCOP state change *LOGATM_STRING D2*

Description: SSCOP unusual state change with one arg

SAAL.014

Level: UI-ERROR

Short Syntax: SAAL.014 nt *n_net LOGATM_STRING*

Long Syntax: SAAL.014 Net *n_net LOGATM_STRING*

Description: SSCOP internal error

SAAL.015

Level: UI-ERROR

Short Syntax: SAAL.015 nt *n_net* SSCOP state change *LOGATM_STRING*

Long Syntax: SAAL.015 Net *n_net* SSCOP state change *LOGATM_STRING*

Description: SSCOP unusual state change

SAAL.016

Level: C-INFO

Short Syntax: SAAL.016 nt *n_net* recv *LOGATM_STRING*, seq= *seq*,len= *len*

Long Syntax: SAAL.016 Net *n_net* receive *LOGATM_STRING*, sequence number = *seq*, length= *len*

Description: SSCOP receive sequenced data

SAAL.017

Level: UE-ERROR

Short Syntax: SAAL.017 nt *n_net LOGATM_STRING*

Long Syntax: SAAL.017 Net *n_net LOGATM_STRING*

Description: SSCOP external error

SAAL.018

Level: UE-ERROR

Short Syntax: SAAL.018 nt *n_net LOGATM_STRING D2*

Long Syntax: SAAL.018 Net *n_net LOGATM_STRING D2*

Description: SSCOP external error with one arg

SAAL.019

Level: C-INFO

Short Syntax: SAAL.019 nt *n_net LOGATM_STRING*, *sequence*, *size*

Long Syntax: SAAL.019 Net *n_net LOGATM_STRING*, *sequence*, *size*

Description: SSCOP transmit packet with sequence number and size

SAAL.020

Level: C-INFO

Short Syntax: SAAL.020 nt *n_net* SSCOP *LOGATM_STRING* timeout

Long Syntax: SAAL.020 Net *n_net* SSCOP *LOGATM_STRING* timeout

Description: SSCOP timeout

SAAL.021

Level: UE-ERROR

Short Syntax: SAAL.021 nt *n_net* SSCOP rcv err, *LOGATM_STRING*

Long Syntax: SAAL.021 Net *n_net* SSCOP rcv err, *LOGATM_STRING*

Description: SSCOP receive error

SAAL.022

Level: U-INFO

Short Syntax: SAAL.022 nt *n_net* xmit *LOGATM_STRING*: *D2 D3 D4 D5*, len= *len*

Long Syntax: SAAL.022 Net *n_net* transmit *LOGATM_STRING*: *D2 D3 D4 D5*, length= *len*

Description: SSCOP transmit data

SAAL.023

Level: U-INFO

Short Syntax: SAAL.023 nt *n_net* recv
LOGATM_STRING: *D2 D3 D4 D5*, len= *len*

Long Syntax: SAAL.023 Net *n_net* receive
LOGATM_STRING: *D2 D3 D4 D5*, length= *len*

Description: SSCOP receive data

SAAL.024

Level: P_TRACE

Short Syntax: SAAL.024 Trace SAAL packet

Long Syntax: SAAL.024 Trace SAAL packet

Description: Trace SAAL packet

SAAL.025

Level: C-INFO

Short Syntax: SAAL.025 nt *n_net* xmit
LOGATM_STRING: *D2 D3 D4 D5*, len= *len*

Long Syntax: SAAL.025 Net *n_net* transmit
LOGATM_STRING: *D2 D3 D4 D5*, length= *len*

Description: SSCOP transmit poll or status

SAAL.026

Level: C-INFO

Short Syntax: SAAL.026 nt *n_net* recv
LOGATM_STRING: *D2 D3 D4 D5*, len= *len*

Long Syntax: SAAL.026 Net *n_net* receive
LOGATM_STRING: *D2 D3 D4 D5*, length= *len*

Description: SSCOP receive poll or status

Chapter 13. Auto Install Functions (AI)

This chapter describes Auto Install Functions (AI) messages. For information on message content and how to use the message, refer to the Introduction.

AI.001

Level: ALWAYS

Short Syntax: AI.001 Changed params on ifc *ifNum* (*subsytemName*), from *oldParams* to *newParams*.

Long Syntax: AI.001 Changed parameters on interface *ifNum* (*subsytemName*) from *oldParams* to *newParams*

Description: Subsystem parameters changed during EasyStart configuration download attempt.

Chapter 14. Bandwidth Reservation System (BRS)

This chapter describes Bandwidth Reservation System (BRS) messages. For information on message content and how to use the message, refer to the Introduction.

BRS.001

Level: C_INFO

Short Syntax: BRS.001 pkt *iob* prot/filt *msg prot/filt type* size *size* queued in class *class name* prio *priority* nt *interface number* int *network ID*

Long Syntax: BRS.001 packet with Id *iob* for protocol/filter *msg prot/filt type* size *size* placed in class *class name* at priority *priority* network *interface number* int *network ID*

Description: A packet is placed in the class at a priority based on its protocol/filter.

BRS.002

Level: C_INFO

Short Syntax: BRS.002 pkt of prot *msg prot type* size *size* is disc'ed by overflow nt *interface number* int *network ID*

Long Syntax: BRS.002 a packet of protocol *msg prot type* size *size* is discarded because of queue overflow network *interface number* int *network ID*

Description: Notifies on all packet overflows

BRS.003

Level: C_INFO

Short Syntax: BRS.003 zero length pkt of prot *msg prot type* is disc'ed nt *interface number* int *network ID*

Long Syntax: BRS.003 a zero length packet of protocol *msg prot type* is discarded network *interface number* int *network ID*

Description: msg when zero length pkts are dumped

BRS.004

Level: C_INFO

Short Syntax: BRS.004 pkt *iob* prot/filt *protocol or filter name* size *size* xmit from class *class name* nt *interface number* int *network ID*

Long Syntax: BRS.004 packet with Id *iob* for prot or filter *protocol or filter name* size *size* is transmitted from class *class name* network *interface number* int *network ID*

Description: A packet is placed from handler struct to driver queue for xmit.

BRS.005

Level: C_INFO

Short Syntax: BRS.005 Lost prior *other items affected* mappings of *prot or filt* for nt *interface number* int *network ID*

Long Syntax: BRS.005 Lost priority *other items affected* mappings of *prot or filt* for network *interface number* int *network ID*

Description: The configuration record for protocol or filter mappings is not present in SRAM. Default mappings have been assumed.

Cause: Either the original configuration record for protocol or filter mappings is not supported by this level of software or configuration memory has been corrupted.

Action: Either reconfigure the mappings or use the configuration tool upgrade facility when the original configuration record is not supported by the current level of software. Contact customer service if configuration memory has been corrupted.

BRS.006

Level: C_INFO

Short Syntax: BRS.006 No memory to create BRS structure. BRS not enabled on nt *other items affected* int *interface number*

Long Syntax: BRS.006 No memory to create BRS structure. BRS not enabled on network *other items affected* int *interface number*

Description: BRS encountered memory allocation error in attempt to allocate storage required for BRS operation.

Cause: Out of memory

Action: Contact customer service.

BRS.008

Level: C_INFO

Short Syntax: BRS.008 pkt *iob* prot/filt *protocol or filter name* size *size* intercepted by Frame Relay from class *class name* nt *interface number* int *network ID*

Long Syntax: BRS.008 packet with Id *iob* for prot or filter *protocol or filter name* size *size* intercepted by FR

from class *class name* network *interface number* int *network ID*

Description: A packet is intercepted by Frame Relay for fragmentation.

BRS.009

Level: C_INFO

Short Syntax: BRS.009 Frag *fragment* of *total frags* for prot/filt *protocol/filter name* size *size* xmit from class *class name* nt *interface number* int *network ID*

Long Syntax: BRS.009 Fragment *fragment* of *total frags* for prot or filter *protocol/filter name* size *size* is transmitted from class *class name* network *interface number* int *network ID*

Description: A Frame Relay fragment is placed from handler struct to driver queue for xmit.

BRS.010

Level: C_INFO

Short Syntax: BRS.010 Interleave higher prio single seg pkt for prot/filt *prot/filt type* size *size* from class *class name* network *interface number* int *network id*

Long Syntax: BRS.010 Fragment sequence interrupted to send higher priority single segment packet for prot/filt *prot/filt type* size *size* from class *class name* network *interface number* int *network id*

Description: Interleave higher priority single segment packet from same class as current fragmentation sequence.

BRS.011

Level: UE_ERROR

Short Syntax: BRS.011 BRS disabled for FH circuit *name* DLCI *number* nt *network ID*

Long Syntax: BRS.011 BRS can not be enabled for frame relay frame handler circuits. The BRS definition for circuit *name* DLCI *number* has been ignored, nt *network ID*

Description: BRS definition invalid and ignored.

BRS.012

Level: C_INFO

Short Syntax: BRS.012 Whole pkt size *PKT size* Frag *fragment* of *total frags* size *size* xmit from BRS to DD n_opq.nq= *n_opq.nq* nt *interface number* int *network ID*

Long Syntax: BRS.012 Whole Pkt Size *PKT size* Fragment *fragment* of *total frags* size *size* xmit from BRS to DD n_opq.nq= *n_opq.nq* network *interface number* int *network ID*

Description: A Frame Relay fragment is placed from handler struct to driver queue for xmit.

BRS.013

Level: C_INFO

Short Syntax: BRS.013 Frag *fragment* of *total frags* size *size* xmit from BRS to DD. n_opq.nq= *n_opq.nq* nt *interface number* int *network ID*

Long Syntax: BRS.013 Fragment *fragment* of *total frags* size *size* xmit from BRS to DD. n_opq.nq= *n_opq.nq* network *interface number* int *network ID*

Description: A Frame Relay fragment is placed from handler struct to driver queue for xmit.

BRS.014

Level: C_INFO

Short Syntax: BRS.014 cml_OSTART netp= *net* n_opc_inc= *n_opc* n_opq.nq= *n_opq.nq* nt *interface number* int *network ID*

Long Syntax: BRS.014 cml_Ostart netp= *net* n_opc_inc= *n_opc* n_opq.nq= *n_opq.nq* network *interface number* int *network ID*

Description: enter cml ostart.

BRS.015

Level: C_INFO

Short Syntax: BRS.015 cml_OCMP netp= *net* n_opc_dec= *n_opc* n_opq.nq= *n_opq.nq*

Long Syntax: BRS.015 cml_OCMP netp= *net* n_opc_dec= *n_opc* n_opq.nq= *n_opq.nq*

Description: enter cml ocmp

BRS.016

Level: C_INFO

Short Syntax: BRS.016 cml_OSTRTERR netp= *net* n_opc_dec= *n_opc* n_opq.nq= *n_opq.nq*

Long Syntax: BRS.016 cml_OSTRTERR netp= *net* n_opc_dec= *n_opc* n_opq.nq= *n_opq.nq*

Description: enter cml ocmp

Chapter 15. Banyan Vines (VN)

This chapter describes Banyan Vines (VN) messages. For information on message content and how to use the message, refer to the Introduction.

VN.001

Level: P-TRACE

Short Syntax: VN.001 *source_vines_network*: *source_vines_subnet* -> *destination_vines_network*: *destination_vines_subnet*

Long Syntax: VN.001 Accepting packet from *source_vines_network*: *source_vines_subnet* for *destination_vines_network*: *destination_vines_subnet*

Description: This message is generated for each VINES packet that successfully passes through the forwarder.

VN.002

Level: CI-ERROR

Short Syntax: VN.002 drp pkt *source_vines_network*: *source_vines_subnet* -> *destination_vines_network*: *destination_vines_subnet* rsn *reason_code*, nt *Network ID*

Long Syntax: VN.002 Dropping packet from *source_vines_network*: *source_vines_subnet* to *destination_vines_network*: *destination_vines_subnet* for reason *reason_code*, net *Network ID*

Description: This message is generated when a packet is not accepted for transmission on a network. The reason code specifies the reason that the packet was dropped.

VN.003

Level: C-TRACE

Short Syntax: VN.003 no rte for pkt *source_vines_network*: *source_vines_subnet* -> *destination_vines_network*: *destination_vines_subnet*

Long Syntax: VN.003 No route for packet from *source_vines_network*: *source_vines_subnet* to *destination_vines_network*: *destination_vines_subnet*

Description: This message is generated when no route can be found for a data packet.

VN.004

Level: U-TRACE

Short Syntax: VN.004 cant alloc for bcast frm *source_vines_network*: *source_vines_subnet*

Long Syntax: VN.004 Cannot allocate buffer to broadcast packet from *source_vines_network*: *source_vines_subnet*

Description: This message is generated when the router receives a broadcast packet and cannot broadcast it out all its interfaces because of a buffer shortage.

VN.005

Level: UE-ERROR

Short Syntax: VN.005 pkt ln *packet_length* too small *source_vines_network*: *source_vines_subnet* -> *destination_vines_network*: *destination_vines_subnet* nt *Network ID*

Long Syntax: VN.005 Packet length (*packet_length*) under minimum VINES packet size from *source_vines_network*: *source_vines_subnet* to *destination_vines_network*: *destination_vines_subnet* net *Network ID*

Description: A packet with a length less than the minimum VINES length was received.

Cause: Problem with source node.

Action: If problem persists, check source node.

VN.006

Level: UE-ERROR

Short Syntax: VN.006 pkt ln *packet_length* too large *source_vines_network*: *source_vines_subnet* -> *destination_vines_network*: *destination_vines_subnet* nt *Network ID*

Long Syntax: VN.006 Packet length (*packet_length*) over maximum VINES packet size from *source_vines_network*: *source_vines_subnet* to *destination_vines_network*: *destination_vines_subnet* net *Network ID*

Description: A packet with a length greater than the maximum VINES length was received.

Cause: Problem with source node.

Action: If problem persists, check source node.

VN.007

Level: UE-ERROR

Short Syntax: VN.007 pkt trunc *specified_length* pkt ln *true_length source_vines_network*: *source_vines_subnet* ->

destination_vines_network: *destination_vines_subnet* nt *Network ID*

Long Syntax: VN.007 Packet truncated from *specified_length* to *true_length* bytes from *source_vines_network*: *source_vines_subnet* for *destination_vines_network*: *destination_vines_subnet* net *Network ID*

Description: This message is generated when the packet length specified in the header is greater than the packet buffer length.

Cause: Packet corruption in transit.

Action: If problem persists, check networks and routers.

Cause: Programming error in remote note.

VN.008

Level: CE-ERROR

Short Syntax: VN.008 hop cnt zero *source_vines_network*: *source_vines_subnet* -> *destination_vines_network*: *destination_vines_subnet*

Long Syntax: VN.008 Hop count expired on packet from *source_vines_network*: *source_vines_subnet* for *destination_vines_network*: *destination_vines_subnet*

Description: This message is generated when a packet is discarded because the hop count expired.

VN.009

Level: C-TRACE

Short Syntax: VN.009 snd ICP pkt for unrch dest *source_vines_network*: *source_vines_subnet* -> *destination_vines_network*: *destination_vines_subnet*

Long Syntax: VN.009 Sending ICP unreachable packet to source *source_vines_network*: *source_vines_subnet* for destination *destination_vines_network*: *destination_vines_subnet*

Description: This message is generated when an ICP packet is returned to the source of a packet with an unreachable destination.

VN.010

Level: UE-ERROR

Short Syntax: VN.010 bd hdr cks frm *source_vines_network*: *source_vines_subnet*, expct *expected_checksum*, gt *actual_checksum*, nt *Network ID*

Long Syntax: VN.010 Bad header checksum in packet from *source_vines_network*: *source_vines_subnet*, expected *expected_checksum*, got *actual_checksum*, nt *Network ID*

Description: This message is generated when a packet destined for the local router has an invalid checksum.

Cause: Most likely, this is a damaged packet. It may be that another node is building an incorrect header.

Action: If the problem persists, examine a line trace to determine where the packet is being damaged.

VN.011

Level: U-INFO

Short Syntax: VN.011 q ovrf *source_vines_network*: *source_vines_subnet* -> *destination_vines_network*: *destination_vines_subnet* net *network ID*

Long Syntax: VN.011 Queue overflow on packet from *source_vines_network*: *source_vines_subnet* for *destination_vines_network*: *destination_vines_subnet* from net *network ID*

Description: This message is generated when the forwarder must discard a packet because of a queue overflow.

VN.012

Level: UI-ERROR

Short Syntax: VN.012 cant alloc for ICP to *destination_vines_network*: *destination_vines_subnet*

Long Syntax: VN.012 Cannot allocate a buffer for an ICP message to node *destination_vines_network*: *destination_vines_subnet*

Description: This message is generated when the router cannot send an ICP message due to no buffers.

VN.013

Level: C-INFO

Short Syntax: VN.013 rcv echo frm *neighbor_hardware_address* nt *network ID*

Long Syntax: VN.013 Received echo packet from *neighbor_hardware_address* net *network ID*

Description: This message is generated when the router receives a VINES IP Echo packet.

VN.014

Level: CI-ERROR

Short Syntax: VN.014 cant snd echo rpl to *neighbor_hardware_address* rsn *reason_code* nt *network ID*

Long Syntax: VN.014 Cannot send echo packet to *neighbor_hardware_address* for reason *reason_code* net *network ID*

Description: This message is generated when the router receives a VINES IP Echo packet and cannot respond to it. The reason code specifies the reason that the router could not send the response.

VN.015

Level: P-TRACE

Short Syntax: VN.015 dsc pkt *source_vines_network*: *source_vines_subnet* -> *destination_vines_network*: *destination_vines_subnet* nt *Network ID* no VINES

Long Syntax: VN.015 Discarded packet from *source_vines_network*: *source_vines_subnet* for *destination_vines_network*: *destination_vines_subnet* net *Network ID*, no VINES forwarder

Description: This message is generated by the fake VINES forwarder for each packet which is received on an interface that is not enabled for VINES.

VN.016

Level: UE-ERROR

Short Syntax: VN.016 bd brdc hdr cks frm *source_vines_network*: *source_vines_subnet*, expct *expected_checksum*, gt *actual_checksum*, nt *Network ID*

Long Syntax: VN.016 Bad broadcast header checksum in packet from *source_vines_network*: *source_vines_subnet*, expected *expected_checksum*, got *actual_checksum*, net *Network ID*

Description: This message is generated when a packet that is part of a VINES link level broadcast arrives at the router with an invalid checksum.

Cause: Most likely, this is a damaged packet. It may be that another node is building an incorrect header.

Action: If the problem persists, examine a line trace to determine where the packet is being damaged.

VN.017

Level: U-INFO

Short Syntax: VN.017 rcv pkt frm *source_vines_network*: *source_vines_subnet* prot *protocol* no srvr nt *Network ID*

Long Syntax: VN.017 Packet from *source_vines_network*: *source_vines_subnet*, protocol *protocol*; no server net *Network ID*

Description: This message is generated when a packet arrives for an unknown protocol. The packet is destined for the local router.

VN.018

Level: C-TRACE

Short Syntax: VN.018 brd pkt *source_vines_network*: *source_vines_subnet* prot *protocol* no srvr nt *Network ID*

Long Syntax: VN.018 Broadcast packet from *source_vines_network*: *source_vines_subnet*, protocol *protocol*; no server net *Network ID*

Description: This message is generated when a VINES IP broadcast packet arrives for an unknown protocol.

VN.019

Level: CE-ERROR

Short Syntax: VN.019 rcv unk nonbrd pkt typ *packet_type* trprt ctrl *transport_control* info *information* for lcl rtr frm *source_vines_network*: *source_vines_subnet* nt *Network ID*

Long Syntax: VN.019 Received unknown non-broadcast packet of type *packet_type* transport control *transport_control* with info *information* for the local router from node *source_vines_network*: *source_vines_subnet* net *Network ID*

Description: This message is generated when a data packet is received that is destined for the local router. This should not happen, because the router does not support any VINES protocols higher than level three. The transport control field is from the VINES IP header. If the packet type is IPC (type 1) the info field contains the Destination Port number, and an IPC Error message is returned to the source node. Otherwise, the information field is meaningless.

VN.020

Level: C-INFO

Short Syntax: VN.020 rcv unk brd pkt typ *packet_type* trprt ctrl *transport_control* info *information* frm *source_vines_network*: *source_vines_subnet* nt *Network ID*

Long Syntax: VN.020 Received unknown broadcast data packet type *packet_type* transport control *transport_control* with info *information* from node *source_vines_network*: *source_vines_subnet* net *Network ID*

Description: This message is generated when a VINES IP broadcast data packet is received by the local router. The packet is discarded. The transport control field is from the VINES IP header. If the packet type is IPC (type 1) the info field contains the Destination Port number, and an IPC Error message is returned to the source node. Otherwise, the information field is meaningless.

VN.021

Level: C-INFO

Short Syntax: VN.021 rcv netrpc call msg frm *source_vines_network*: *source_vines_subnet* nt *Network ID*, no such nbr

Long Syntax: VN.021 Received netrpc call message from node *source_vines_network*: *source_vines_subnet* net *Network ID*, no such neighbor

Description: This message is generated when a VINES NetRPC packet containing a Call message is received by the local router, but the router does not have a record of the neighbor that the source node is

requesting information about. A NetRPC Abort message is returned to the source node.

VN.022

Level: C-INFO

Short Syntax: VN.022 rcv netrpc call msg frm *source_vines_network*: *source_vines_subnet* port *port* nt *Network ID*, nbr exists

Long Syntax: VN.022 Received netrpc call message from node *source_vines_network*: *source_vines_subnet* port *port* net *Network ID*, neighbor exists

Description: This message is generated when a VINES NetRPC packet containing a Call message is received by the local router on the given IPC port. A NetRPC Return message is returned to the source node.

VN.023

Level: U-INFO

Short Syntax: VN.023 rcv rte pkt with hop cnt gtr than zero frm *source_vines_network*: *source_vines_subnet* nt *Network ID*

Long Syntax: VN.023 Received a routing packet with a hop count greater than zero from *source_vines_network*: *source_vines_subnet* net *Network ID*

Description: This message is generated when a routing update or routing response packet with a hop count of greater than zero was received. The packet is discarded.

VN.024

Level: C-TRACE

Short Syntax: VN.024 snd rte rsp to *destination_vines_network*: *destination_vines_subnet*

Long Syntax: VN.024 Sending routing response packet to node *destination_vines_network*: *destination_vines_subnet*

Description: This message is generated when this router is about to send a routing response packet.

VN.025

Level: UI-ERROR

Short Syntax: VN.025 cant alloc for rte rsp to nt *destination_vines_network*

Long Syntax: VN.025 Cannot allocate a buffer to send a routing response to network *destination_vines_network*

Description: This message is generated when the router attempts to send a routing response packet but cannot because no buffers are available.

VN.026

Level: CE-ERROR

Short Syntax: VN.026 rcv rte pkt on uncng int frm *source_vines_network*: *source_vines_subnet* nt *Network ID*

Long Syntax: VN.026 Received a routing packet on interface not configured for VINES from *source_vines_network*: *source_vines_subnet* net *Network ID*

Description: This message is generated when a routing update or routing response was received on an interface that is not configured to run VINES.

VN.027

Level: P-TRACE

Short Syntax: VN.027 snd rte pkt typ *routing_packet_type*

Long Syntax: VN.027 Sending a routing packet of type *routing_packet_type*

Description: This message is generated when the router is sending a routing packet. A type of 0 means the update contains only routing entries that have changed recently. A type of 1 means it is a full routing update. A type of 2 means it is a routing request packet (only sent on X.25 circuits). A type of 3 means the update contains changes intended only for X.25 circuits.

VN.028

Level: U-TRACE

Short Syntax: VN.028 rcv rte rsp frm *destination_vines_network*: *destination_vines_subnet*

Long Syntax: VN.028 Received routing response packet from *destination_vines_network*: *destination_vines_subnet*

Description: This message is generated when a routing response packet was received. The packet is accepted and processed.

VN.029

Level: P-TRACE

Short Syntax: VN.029 rcv rte upd frm *destination_vines_network*: *destination_vines_subnet* nt *Network ID*

Long Syntax: VN.029 Received routing update packet from *destination_vines_network*: *destination_vines_subnet* net *Network ID*

Description: This message is generated when a routing update packet is received.

VN.030

Level: U-INFO

Short Syntax: VN.030 cant alloc nbr tbl ent for *neighbor_vines_network*: *neighbor_vines_subnet*

Long Syntax: VN.030 Cannot allocate a neighbor table entry for neighbor *neighbor_vines_network*: *neighbor_vines_subnet*

Description: This message is generated when there are no neighbor table entries on the free list to hold information about the routing update that was just received. The routing update containing that information is discarded.

VN.031

Level: U-INFO

Short Syntax: VN.031 cant alloc nbr cache ent for *neighbor_vines_network*: *neighbor_vines_subnet*

Long Syntax: VN.031 Cannot allocate a neighbor cache entry for neighbor *neighbor_vines_network*: *neighbor_vines_subnet*

Description: This message is generated when there are no neighbor cache entries on the free list to hold information about neighbor Client nodes of the Service node that generated the routing update. The routing update containing that information is discarded.

VN.032

Level: U-INFO

Short Syntax: VN.032 add eql cst rte to nbr *neighbor_vines_network*: *neighbor_vines_subnet* nt *Network ID*

Long Syntax: VN.032 Adding an equal cost route to neighbor *neighbor_vines_network*: *neighbor_vines_subnet* net *Network ID*

Description: This message is generated when an additional, equal cost route to the same neighbor node is added. At this point, there will be at least two, equal cost routes to the same neighbor.

VN.033

Level: U-INFO

Short Syntax: VN.033 cant alloc net tbl ent for *destination_vines_network*

Long Syntax: VN.033 Cannot allocate a network table entry network *destination_vines_network*

Description: This message is generated when there are no network table entries on the free list to hold information about the routing update that was just received. The routing update containing that information is discarded.

VN.034

Level: U-INFO

Short Syntax: VN.034 add eql cst rte for node *vines_network* nt *Network ID*

Long Syntax: VN.034 Adding an equal cost route for node *vines_network* net *Network ID*

Description: This message is generated when an additional, equal cost route to the same network node is added. At this point, there will be at least two, equal cost routes to the same network.

VN.035

Level: U-INFO

Short Syntax: VN.035 updt nt *destination_vines_network* mtrc *metric* via same *next_hop_vines_network*: *next_hop_vines_subnet*

Long Syntax: VN.035 update route to net *destination_vines_network* at metric *metric* via same neighbor *next_hop_vines_network*: *next_hop_vines_subnet*

Description: This message is generated when a new (better) route to the given destination has been learned via a routing update and has been installed.

VN.036

Level: U-INFO

Short Syntax: VN.036 nt *destination_vines_network* unrch inc met

Long Syntax: VN.036 Marking network *destination_vines_network* unreachable due to increased metric

Description: This message is generated when a RTP packet was received that announced an increased metric to a destination network. The destination network is marked unreachable.

VN.037

Level: C-TRACE

Short Syntax: VN.037 rcv rte req frm *destination_vines_network*: *destination_vines_subnet*

Long Syntax: VN.037 Received routing request packet from *destination_vines_network*: *destination_vines_subnet*

Description: This message is generated when a routing request packet is received.

VN.038

Level: UE-ERROR

Short Syntax: VN.038 rcv rte red frm *destination_vines_network*: *destination_vines_subnet*

Long Syntax: VN.038 Received routing redirect packet from *destination_vines_network*: *destination_vines_subnet*

Description: This message is generated when a routing redirect packet is received. The packet is ignored.

VN.039

Level: UE-ERROR

Short Syntax: VN.039 rcv unkn rte pkt frm *destination_vines_network*: *destination_vines_subnet* nt *Network ID*

Long Syntax: VN.039 Received unknown sub-type of routing packet from *destination_vines_network*: *destination_vines_subnet* net *Network ID*

Description: This message is generated when a routing packet with an unknown sub-type is received.

Cause: Confused remote node.

Action: If this problem persists, debug the remote node.

VN.040

Level: C-INFO

Short Syntax: VN.040 rmv nbr entry node *neighbor_vines_network*: *neighbor_vines_subnet*

Long Syntax: VN.040 Removing neighbor entry for node *neighbor_vines_network*: *neighbor_vines_subnet*

Description: This message is generated when the local router has not received a routing packet from a neighbor node for six minutes.

VN.041

Level: C-INFO

Short Syntax: VN.041 rmv rtng entry node *destination_vines_network*: *destination_vines_subnet*

Long Syntax: VN.041 Removing routing entry for node *destination_vines_network*: *destination_vines_subnet*

Description: This message is generated when the local router has not received a routing packet about a Service node for six minutes.

VN.042

Level: UI-ERROR

Short Syntax: VN.042 cant alloc for rte upd

Long Syntax: VN.042 Cannot allocate a buffer to send a routing update.

Description: This message is generated when the router attempts to send a routing update packet but cannot because no buffers are available.

VN.043

Level: UI-ERROR

Short Syntax: VN.043 cant alloc for redir *source_vines_network*: *source_vines_subnet* -> *destination_vines_network*: *destination_vines_subnet* nt *Network ID*

Long Syntax: VN.043 Cannot allocate buffer for redirect packet for packet from *source_vines_network*: *source_vines_subnet* for *destination_vines_network*: *destination_vines_subnet* net *Network ID*

Description: This message is generated when this router tries to send a redirect packet, but cannot because no buffers are available.

VN.044

Level: C-INFO

Short Syntax: VN.044 snd redir *source_vines_network*: *source_vines_subnet* -> *destination_vines_network*: *destination_vines_subnet* nt *Network ID*

Long Syntax: VN.044 Sending redirect packet for packet from *source_vines_network*: *source_vines_subnet* for *destination_vines_network*: *destination_vines_subnet* net *Network ID*

Description: This message is generated when this router sends a redirect packet

Cause: The neighbor node sent a packet to this router, when it could have sent the packet directly to the destination.

Action: If this occurs repeatedly, the neighbor node should be fixed.

VN.045

Level: U-INFO

Short Syntax: VN.045 rcv rte upd frm orphn *source_vines_network*: *source_vines_subnet* nt *Network ID*

Long Syntax: VN.045 Received a routing update packet from orphan client node *source_vines_network*: *source_vines_subnet* net *Network ID*

Description: This message is generated when the router receives a routing update from a neighbor Client node whose associated Service node is not operational. This can happen if the associated Service node has recently gone down.

VN.046

Level: U-TRACE

Short Syntax: VN.046 cant alloc mem fr hdr fr rte upd frm clnt *source_vines_network*: *source_vines_subnet*

Long Syntax: VN.046 Cannot allocate memory for link level header for routing update from Client

source_vines_network: *source_vines_subnet*

Description: This message is generated when the router receives a routing update from a neighbor Client node but cannot allocate memory to hold the link level header of the Client node for its routing table entry.

VN.047

Level: U-TRACE

Short Syntax: VN.047 cant alloc mem fr hdr fr rte upd frm srvr *source_vines_network*: *source_vines_subnet*

Long Syntax: VN.047 Cannot allocate memory for link level header for routing update from Server *source_vines_network*: *source_vines_subnet*

Description: This message is generated when the router receives a routing update from a neighbor Server node but cannot allocate memory to hold the link level header of the Server node for its routing table entry.

VN.048

Level: U-TRACE

Short Syntax: VN.048 cant alloc mem fr hdr fr rte upd nw int frm srvr *source_vines_network*: *source_vines_subnet*

Long Syntax: VN.048 Cannot allocate memory for link level header for routing update on a new interface from Server *source_vines_network*: *source_vines_subnet*

Description: This message is generated when the router receives a routing update from a neighbor Server node that it already has a record of, but on a new interface, and cannot allocate memory to hold the link level header of the Server node for its routing table entry.

VN.049

Level: C-TRACE

Short Syntax: VN.049 no rte for redr pkt *source_vines_network*: *source_vines_subnet* -> *destination_vines_network*: *destination_vines_subnet*

Long Syntax: VN.049 No route for redirect packet from *source_vines_network*: *source_vines_subnet* to *destination_vines_network*: *destination_vines_subnet*

Description: This message is generated when the router has established that a RTP Redirect packet should be sent to the source of a packet, but cannot find a routing entry for the destination of the packet.

VN.050

Level: C-INFO

Short Syntax: VN.050 rmv int rec for nbr node *neighbor_vines_network*: *neighbor_vines_subnet* nt *Network ID*

Long Syntax: VN.050 Removing interface record for neighbor node *neighbor_vines_network*: *neighbor_vines_subnet* net *Network ID*

Description: This message is generated when the local router has not received a routing packet from a neighbor node on a particular interface for six minutes. The neighbor node may still be directly reachable via a different interface.

VN.051

Level: C-INFO

Short Syntax: VN.051 rmv int rec for dest node *vines_network* nt *Network ID*

Long Syntax: VN.051 Removing interface record for destnation node *vines_network* net *Network ID*

Description: This message is generated when the local router has received a routing packet indicating a greater cost to a remote node than the router has in its database for that remote node. However, there remains at least one additional route to the remote node.

VN.052

Level: UE-ERROR

Short Syntax: VN.052 rcv rte pkt with unk X.25 addr *dte_address* frm *source_vines_network*: *source_vines_subnet* nt *Network ID*

Long Syntax: VN.052 Received a routing packet with an unknown X.25 address *dte_address* from *source_vines_network*: *source_vines_subnet* net *Network ID*

Description: This message is generated when a routing update is received from an X.25 node, but the address of the node has not been configured in the local router. The packet is discarded.

VN.053

Level: P-TRACE

Short Syntax: VN.053 rcv ARP qr pkt frm *neighbor_hardware_adrress* nt *Network ID*

Long Syntax: VN.053 Received an ARP query request packet from neighbor *neighbor_hardware_adrress* net *Network ID*

Description: This message is generated when an ARP query request packet is received.

VN.054

Level: UE-ERROR

Short Syntax: VN.054 rcv ARP qr with IP addr frm *neighbor_hardware_address* nt *Network ID*

Long Syntax: VN.054 Received an ARP query request packet with a non-null IP address from

neighbor_hardware_address net *Network ID*

Description: This message is generated when an ARP query request packet is received which contains a non-null IP address. The packet is still processed as normal.

Cause: Confused neighbor node.

Action: If this problem persists, debug the neighbor node.

VN.055

Level: UI-ERROR

Short Syntax: VN.055 no free nbr tbl entries for *neighbor_hardware_address*

Long Syntax: VN.055 No free neighbor table entries for neighbor *neighbor_hardware_address*

Description: This message is generated when an ARP query request is received but no neighbor table entries are available to resolve the neighbor's VINES IP address.

VN.056

Level: UI-ERROR

Short Syntax: VN.056 no free mem for hw addr for *neighbor_hardware_address*

Long Syntax: VN.056 No free memory to hold hardware address for ARP packet from *neighbor_hardware_address*

Description: This message is generated when an ARP query request packet is received, but memory cannot be allocated to save the hardware address of the node which sent the query request.

VN.057

Level: UI-ERROR

Short Syntax: VN.057 no free mem for RIF for *neighbor_hardware_address*

Long Syntax: VN.057 No free memory to hold routing information field for ARP packet from *neighbor_hardware_address*

Description: This message is generated when an ARP query request packet is received, but memory cannot be allocated to save the routing information field of the node which sent the query request.

VN.058

Level: C-TRACE

Short Syntax: VN.058 rcv ARP qr frm *neighbor_hardware_address* while rslving addr nt *Network ID*

Long Syntax: VN.058 Received an ARP query request packet from *neighbor_hardware_address* while resolving address for another node net *Network ID*

Description: This message is generated when an ARP query request packet is received, from one node and the router is in the middle of resolving a VINES IP address from a different node.

VN.059

Level: P-TRACE

Short Syntax: VN.059 rcv ARP ar pkt frm *neighbor_hardware_adrress* nt *Network ID*

Long Syntax: VN.059 Received an ARP assignment request packet from neighbor *neighbor_hardware_adrress* net *Network ID*

Description: This message is generated when an ARP assignment request packet is received.

VN.060

Level: UE-ERROR

Short Syntax: VN.060 rcv ARP ar with IP addr frm *neighbor_hardware_address* nt *Network ID*

Long Syntax: VN.060 Received an ARP assignment request packet with a non-null IP address from *neighbor_hardware_address* net *Network ID*

Description: This message is generated when an ARP assignment request packet is received which contains a non-null IP address. The packet is still processed as normal.

Cause: Confused neighbor node.

Action: If this problem persists, debug the neighbor node.

VN.061

Level: UI-ERROR

Short Syntax: VN.061 cant instl ARP entry frm *neighbor_hardware_address*

Long Syntax: VN.061 Cannot install an ARP entry for address resolution from *neighbor_hardware_address*

Description: This message is generated when an ARP assignment request packet is received, but the router does not have the memory resources to assign a VINES IP address.

VN.064

Level: UE-ERROR

Short Syntax: VN.064 rcv bad ARP subtyp pkt frm *neighbor_hardware_address* nt *Network ID*

Long Syntax: VN.064 Received an ARP packet with a

bad sub-type field from *neighbor_hardware_address* net *Network ID*

Description: This message is generated when an ARP packet is received with an illegal sub-type field.

Cause: Confused neighbor node.

Action: If this problem persists, debug the neighbor node.

VN.066

Level: P-TRACE

Short Syntax: VN.066 snd ARP sr pkt to *neighbor_hardware_adrress* nt *Network ID*

Long Syntax: VN.066 Sending an ARP service response packet to neighbor *neighbor_hardware_adrress* net *Network ID*

Description: This message is generated when an ARP service response packet is generated in response to an ARP query request packet.

VN.067

Level: P-TRACE

Short Syntax: VN.067 snd ARP ar pkt to *neighbor_hardware_adrress* nt *Network ID*

Long Syntax: VN.067 Sending an ARP assignment response packet to neighbor *neighbor_hardware_adrress* net *Network ID*

Description: This message is generated when an ARP assignment response packet is generated in response to an ARP assignment request packet.

VN.068

Level: C-INFO

Short Syntax: VN.068 rcv ICP exc not frm *source_vines_network*: *source_vines_subnet*

Long Syntax: VN.068 Received ICP exception notification packet from node *source_vines_network*: *source_vines_subnet*

Description: This message is generated when the router receives an ICP exception notification packet.

VN.069

Level: CE-ERROR

Short Syntax: VN.069 rcv ICP metr not frm *source_vines_network*: *source_vines_subnet*

Long Syntax: VN.069 Received ICP metric notification packet from node *source_vines_network*: *source_vines_subnet*

Description: This message is generated when an ICP metric notification packet is received. This should never happen, because the router will never generate a metric request packet.

Cause: Confused neighbor node.

Action: If this problem persists, debug the destination node.

VN.070

Level: UE-ERROR

Short Syntax: VN.070 rcv ICP illeg subtyp frm *source_vines_network*: *source_vines_subnet*

Long Syntax: VN.070 Received ICP packet with illegal sub-type from node *source_vines_network*: *source_vines_subnet*

Description: This message is generated when an ICP packet is received with an illegal sub-type field.

Cause: Confused destination node.

Action: If this problem persists, debug the destination node.

VN.071

Level: C-TRACE

Short Syntax: VN.071 snd rte cost icp pkt to *destination_vines_network*: *destination_vines_subnet*

Long Syntax: VN.071 Sending routing cost ICP packet to node *destination_vines_network*: *destination_vines_subnet*

Description: This message is generated when an ICP packet is sent to a node that requested the routing cost from this router to one of its neighbors.

VN.072

Level: C-TRACE

Short Syntax: VN.072 snd no rte icp pkt to *destination_vines_network*: *destination_vines_subnet*

Long Syntax: VN.072 Sending No Route ICP packet to node *destination_vines_network*: *destination_vines_subnet*

Description: This message is generated when an ICP packet is sent to a node because it sent a packet to an unreachable destination.

VN.073

Level: C-TRACE

Short Syntax: VN.073 rcv icp echo pkt frm *source_vines_network*: *source_vines_subnet*

Long Syntax: VN.073 Received ICP Echo packet from *source_vines_network*: *source_vines_subnet*

Description: This message is generated when an ICP Echo Request packet is received. The router responds

with an ICP Echo Reply packet.

VN.074

Level: C-INFO

Short Syntax: VN.074 VINES init nt *network_number*, rtl tbl sz *routing_table_entries*, max svc nbrs *max_service_neighbors*, max clt nbrs *max_client_neighbors*

Long Syntax: VN.074 The VINES protocol is initializing with network number *network_number*, max routing table entries *routing_table_entries*, max service node neighbors *max_service_neighbors*, max client node neighbors *max_client_neighbors*

Description: This message is generated when the VINES protocol runs its initialization code.

VN.075

Level: U-INFO

Short Syntax: VN.075 No VINES IP addr

Long Syntax: VN.075 No VINES IP address is configured for this router

Description: This message is generated when VINES is enabled on the router, but the user has not assigned a VINES IP address to the router. The VINES protocol will not be initialized.

VN.076

Level: U-INFO

Short Syntax: VN.076 int dlt but not VINES nt *network ID* dlt

Long Syntax: VN.076 Interface record deleted, but VINES interface record net *network ID* not deleted

Description: This message is generated when the user has deleted a router interface record without deleting the VINES record for that interface.

VN.077

Level: U-INFO

Short Syntax: VN.077 int max pkt sz too sml nt *network ID*

Long Syntax: VN.077 The maximum packet size of net *network ID* is smaller than the maximum VINES packet size

Description: This message is generated when an interface has a maximum packet size smaller than the maximum VINES packet size. This can happen if the user configures the interface for a maximum packet size smaller than its default. The interface will not be enabled for VINES.

VN.079

Level: U-INFO

Short Syntax: VN.079 No Int cfg

Long Syntax: VN.079 No Interfaces have been configured, so Vines will not be started.

Description: Vines must detect that there are interfaces defined for the router (even if they will not be used for Vines) and also must have at least one interface or X.25 address to talk with or else the protocol will not start.

Chapter 16. Bisync Relay (BRLY)

This chapter describes Bisync Relay (BRLY) messages. For information on message content and how to use the message, refer to the Introduction.

BRLY.001

Level: C-INFO

Short Syntax: BRLY.001 BRLY interface initialization starting network *networkID*

Long Syntax: BRLY.001 BRLY initialization started on network *networkID*

Description: BRLY forwarder has started initialization on the relay interface.

BRLY.002

Level: C-INFO

Short Syntax: BRLY.002 BRLY interface initialization complete network *networkID*

Long Syntax: BRLY.002 BRLY initialization completed on network *networkID*

Description: BRLY forwarder has completed initialization on the relay interface.

BRLY.003

Level: C-TRACE

Short Syntax: BRLY.003 BRLY frame received on network *networkID*

Long Syntax: BRLY.003 BRLY frame received on network *networkID*

Description: BSC Relay frame received.

BRLY.004

Level: C-INFO

Short Syntax: BRLY.004 BRLY frame sent on network *networkID*

Long Syntax: BRLY.004 BRLY frame sent on network *networkID*

Description: BSC Relay frame transmitted.

BRLY.005

Level: UI-ERROR

Short Syntax: BRLY.005 BRLY frame discarded for group *group_number* on network *networkID* - *discard_reason*

Long Syntax: BRLY.005 BRLY frame discarded for group *group_number* on network *networkID* - *discard_reason*

Description: BRLY frame discarded.

BRLY.006

Level: CI-ERROR

Short Syntax: BRLY.006 BRLY memory allocation failed

Long Syntax: BRLY.006 A BRLY memory allocation request failed

Description: A BRLY memory allocation request for resources has failed.

BRLY.007

Level: CI-ERROR

Short Syntax: BRLY.007 BRLY port defined for non-BSC net or invalid hdw - net *networkID*

Long Syntax: BRLY.007 BRLY port defined for non-BSC net or invalid hdw - network *networkID*

Description: BRLY port defined on non-BSC net or using invalid hardware - disabled.

BRLY.008

Level: CI-ERROR

Short Syntax: BRLY.008 Configuration error for group *group_number* - *configuration_error* - group internally disabled

Long Syntax: BRLY.008 Configuration error for group *group_number* - *configuration_error* - group internally disabled

Description: Group configuration error.

BRLY.009

Level: CI-ERROR

Short Syntax: BRLY.009 BRLY frame discarded on network *networkID* - *discard_reason*

Long Syntax: BRLY.009 BRLY frame discarded on network *networkID* - *discard_reason*

Description: BRLY frame discarded.

BRLY.010

Level: CI-ERROR

Short Syntax: BRLY.010 BRLY frame discarded - *discard_reason*

Long Syntax: BRLY.010 BRLY frame discarded - *discard_reason*

Description: BRLY frame discarded.

Panic brlyudperr

Short Syntax: BSC Relay UDP port not available\r\n

Description: Another application registered previously with BSC Relay's UDP port.

Action: Contact customer service.

Chapter 17. Bootp (BTP)

This chapter describes Bootp (BTP) messages. For information on message content and how to use the message, refer to the Introduction.

BTP.001

Level: U-TRACE

Short Syntax: BTP.001 rcvd rqst frm (*client_IP_address*, nt *Network ID*)

Long Syntax: BTP.001 received request from (*client_IP_address*, net *Network ID*)

Description: A BOOTP request has been received on a particular interface. The client IP address is included in the message, but may be unknown at this time, in which case it will show up as 0.0.0.0.

BTP.002

Level: UE-ERROR

Short Syntax: BTP.002 bd rqst frm (*client_IP_address*, nt *Network ID*): *reason*

Long Syntax: BTP.002 bad request from (*client_IP_address*, net *Network ID*): *reason*

Description: A BOOTP request has been received on a particular interface. The client IP address is included in the message, but may be unknown at this time, in which case it will show up as 0.0.0.0. The request is bad for the stated reason, and is therefore discarded.

BTP.003

Level: U-TRACE

Short Syntax: BTP.003 fwd rqst to *server_IP_address*

Long Syntax: BTP.003 Forwarding request to *server_IP_address*

Description: A BOOTP request is being forwarded to a particular server.

BTP.004

Level: U-TRACE

Short Syntax: BTP.004 fwd rply *server_IP_address* -> *client_IP_address*

Long Syntax: BTP.004 Forwarding reply from *server_IP_address* to *client_IP_address*

Description: A BOOTP reply is being forwarded from a particular server back to the client, using the router as a relay agent.

BTP.005

Level: UE-ERROR

Short Syntax: BTP.005 bad reply *server_IP_address* -> *client_IP_address*: *reason*

Long Syntax: BTP.005 bad reply from *server_IP_address* to *client_IP_address*: *reason*

Description: A BOOTP reply was received. We are unable to relay the reply to the client due to the stated error. The BOOTP reply has been discarded.

BTP.006

Level: ALWAYS

Short Syntax: BTP.006 net *Network ID*, gw *source_IP_address*: Client reply packet in error; *error*

Long Syntax: BTP.006 net *Network ID*, gw *source_IP_address*: Client reply packet in error; *error*

Description: A reply was received from a BOOTP server that was either the incorrect packet type or it was too short. The gatway address is the router that did the final relay from the server to this client. It could be the server address.

BTP.007

Level: ALWAYS

Short Syntax: BTP.007 net *Network ID*, Valid Resp, Server: *serverName*(*serverIp*), Bootfile: *bootfile* VendOpts config file: *cfgFile* IpAddr *ourIp*, gwAddr *gwAddr*

Long Syntax: BTP.007 net *Network ID*, Valid Resp, server: *serverName*/ *serverIp*, bootfile: *bootfile*, vendor options config File: *cfgFile*, ipAddr *ourIp*, gwAddr *gwAddr*

Description: A valid BOOTP reply packet was received from a server.

BTP.008

Level: ALWAYS

Short Syntax: BTP.008 net *Network ID* No cfg file name (srv: *serverName*/ *serverIp*)

Long Syntax: BTP.008 net *Network ID* No config file name in packet (srv: *serverName*/ *serverIp*)

Description: A reply was received from a server without a configuration filename in the vendor

extension field or in the boot filename field.

BTP.009

Level: ALWAYS

Short Syntax: BTP.009 net *Network ID*, Failed to snd client req (htype: *htype*)

Long Syntax: BTP.009 net *Network ID*, Failed to send client request (htype: *htype*)

Description: An attempt to send the BOOTP request failed.

BTP.010

Level: ALWAYS

Short Syntax: BTP.010 net *Network ID*, Sent client request (htype: *htype*)

Long Syntax: BTP.010 net *Network ID*, Sent client request (htype: *htype*)

Description: A BOOTP client request was successfully sent.

BTP.011

Level: ALWAYS

Short Syntax: BTP.011 net *Network ID*, Could not snd client req because: *error*

Long Syntax: BTP.011 net *Network ID*, Could not send client request because: *error*

Description: An attempt to send the BOOTP request failed because the output device does not support BOOTP, the device is not up, or a buffer could not be allocated.

BTP.012

Level: ALWAYS

Short Syntax: BTP.012 net *Network ID* No cfile in vendOptions, using bootfile fld instead

Long Syntax: BTP.012 net *Network ID* No cfile in vendOptions, using bootfile fld instead

Description: The vendor extensions for the configuration filename was not in the response. The router will use the bootfile name field in its place.

BTP.013

Level: ALWAYS

Short Syntax: BTP.013 net *Network ID* Unsupported vend tag: *vendTag*, len: *vendLen*

Long Syntax: BTP.013 net *Network ID* Reply received with unsupported vendor tag field: *vendTag*, len *vendLen*

Description: The server sent a BOOTP reply packet with a vendor field containing an unsupported vendor specific option. This is not critical; it only means that the BOOTP server is not configured correctly for this BOOTP client.

BTP.014

Level: U-TRACE

Short Syntax: BTP.014 cached rqst frm *client_hardware_address*

Long Syntax: BTP.014 cached request from *client_hardware_address*

Description: A BOOTP client request was cached.

BTP.015

Level: U-TRACE

Short Syntax: BTP.015 rmvd cached rqst frm *client_hardware_address*: *reason*

Long Syntax: BTP.015 removed cached request from *client_hardware_address*: *reason*

Description: A BOOTP client request was removed from the cache.

Panic btpudperr

Short Syntax: bootp udp port not avail

Description: Another application registered previously with bootp's UDP port.

Action: Contact customer service.

Chapter 18. Border Gateway Protocol (BGP)

This chapter describes Border Gateway Protocol (BGP) messages. For information on message content and how to use the message, refer to the Introduction.

BGP.001

Level: UI-ERROR

Short Syntax: BGP.001 Bad sec code in OPEN, from *neighbor*

Long Syntax: BGP.001 BGP security code in OPEN message is incorrect from neighbor *neighbor*

Description: The BGP RFC specifies only a single acceptable security code of 0. This message is printed if another code is received.

Cause: Neighbor sent a security code in the OPEN message that is non null.

Action: Use a router that adheres more closely to the BGP specification.

BGP.002

Level: UI-ERROR

Short Syntax: BGP.002 Bad msg hdr len, from *neighbor*

Long Syntax: BGP.002 BGP message header length is incorrect from neighbor *neighbor*

Description: The speaker received a message in which the header length was incorrect.

Cause: Neighbor sent an OPEN message that is of incorrect length.

Action: Use a router that adheres to the BGP specification.

BGP.003

Level: U-INFO

Short Syntax: BGP.003 Unsupported BGP version, from *neighbor*

Long Syntax: BGP.003 Unsupported BGP version request from neighbor *neighbor*

Description: The current version supported by BGP is version 4. No other version support exists. This message is printed when a neighbor requests a lower version of BGP.

Cause: Neighbor is requesting a version of BGP, which is unsupported.

Action: Neighbor router must be configured for the proper version.

BGP.004

Level: UI-ERROR

Short Syntax: BGP.004 Bad marker fld, from *neighbor*

Long Syntax: BGP.004 Marker field is incorrect from neighbor *neighbor*

Description: The only supported marker field is 16 octets of all ones. This message is printed when any other value is received.

Cause: Neighbor is using an incorrect marker field.

Action: Use a router that adheres to the BGP specification.

BGP.005

Level: UI-ERROR

Short Syntax: BGP.005 Bad AS num, from *neighbor*

Long Syntax: BGP.005 Bad AS number from neighbor *neighbor*

Description: This message is printed when the neighbor's AS number in OPEN message does not match the configured value for that neighbor.

Cause: Neighbor is using an AS that does not match the configured value.

Action: Make sure that the neighbor and this router have properly configured AS numbers.

BGP.006

Level: UI-ERROR

Short Syntax: BGP.006 Bad BGP ID, from *neighbor*

Long Syntax: BGP.006 Bad BGP identifier from neighbor *neighbor*

Description: This message is printed when the neighbor and this speaker have the same BGP identifier. Since this is used to resolve TCP connection collisions, this is an illegal configuration.

Cause: Neighbor is using a BGP identifier that is the same as this one.

Action: Make sure that the neighbor and this router have properly configured BGP identifiers.

BGP.007

Level: U-INFO

Short Syntax: BGP.007 Conn err to *neighbor*; clsg with notify

Long Syntax: BGP.007 Closing connection to neighbor *neighbor* with notification

Description: Some error in the connection Finite State Machine resulted in this message.

Cause: An error in the connection Finite State Machine resulted in connection termination.

Action: Note other connection errors that occur with this event.

BGP.008

Level: U-INFO

Short Syntax: BGP.008 Conn err to *neighbor*; clsg with no notify

Long Syntax: BGP.008 Closing connection to neighbor *neighbor* without notification

Description: Some error in the connection Finite State Machine resulted in this message, usually because this speaker received a NOTIFICATION message and there is no reason to send another one back to the neighbor who sent this.

Cause: An error in the connection Finite State Machine resulted in connection termination.

Action: Note other connection errors that occur along with this one.

BGP.009

Level: UI-ERROR

Short Syntax: BGP.009 Foreign close from *neighbor* sprt *sourceport* dprt *destinationport*

Long Syntax: BGP.009 Foreign close from neighbor *neighbor* source port *sourceport* destination port *destinationport*

Description: The speaker just received a foreign close.

Cause: Neighbor is issuing a close.

Action: Neighbor should issue a close after a notification or during BGP ID negotiation. If this is the case, no action is necessary. If a connection closes for reasons other than these, the neighbor is in error.

BGP.010

Level: U-INFO

Short Syntax: BGP.010 Reinit BGP conn to *neighbor*

Long Syntax: BGP.010 Reinitialize the BGP connection to neigbor *neighbor*

Description: If a previous connection to this neighbor resulted in termination, the speaker reinitiates the connection. This message is printed when this occurs.

Cause: Speaker is reinitializing a connection to this neighbor after an initial failure.

Action: None, unless this happens many times with no connection to the neighbor.

BGP.011

Level: U-INFO

Short Syntax: BGP.011 Conn to *neighbor* clsg with no notify

Long Syntax: BGP.011 Connection to neighbor *neighbor* closing with no notification

Description: Probably in response to a NOTIFICATION message received from the other end, the router is closing the BGP connection to the neighbor without sending a notify.

BGP.012

Level: UI-ERROR

Short Syntax: BGP.012 No conn listen can be done

Long Syntax: BGP.012 No connection listen can be done

Description: Something is preventing the speaker from issuing a listen.

Cause: Probably an internal error in the TCP subsystem. Also, the router could be low on memory.

Action: Check for low memory. If memory is low, check the BGP config statistics for memory utilization. A large number of neighbor connections can conceivably use up memory.

BGP.013

Level: UI-ERROR

Short Syntax: BGP.013 TCP open fail to *neighbor*

Long Syntax: BGP.013 TCP open failure to BGP neighbor *neighbor*

Description: The BGP speaker initiates a tcp_listen request in order to receive connection requests from neighbors. This message is printed when the invocation to this function fails.

Cause: The open to the TCP subsystem failed.

Action: Serious problem. Check amount of heap memory available to router.

BGP.014

Level: U-INFO

Short Syntax: BGP.014 Conn timer fired for *neighbor*

Long Syntax: BGP.014 Connection timer fired for neighbor *neighbor*

Description: A connection timer is used to continue attempts to make active connections from this speaker to this neighbor. The firing of this timer causes the speaker to quit the previous tcp_open and issue another tcp_open.

Cause: The connection timer fired because no neighbor connection was completed in the specified time.

Action: None. Connection process will continue until connection to neighbor completes.

BGP.015

Level: U-INFO

Short Syntax: BGP.015 conn to *neighbor* open on sprt *sourceport* dprt *destinationport*

Long Syntax: BGP.015 connection to neighbor *neighbor* open on soure port *sourceport* destination port *destinationport*

Description: An OPEN message has been received on this connection for this neighbor.

Cause: The connection to the neighbor has completed successfully.

Action: None. This is an informational message.

BGP.016

Level: U-INFO

Short Syntax: BGP.016 OPEN sent to *neighbor*

Long Syntax: BGP.016 OPEN message sent to neighbor *neighbor*

Description: When a connection is opened, the speaker sends an OPEN message to the neighbor. This message is printed when this happens.

Cause: This is part of the connection process.

Action: None. This is an informational message.

BGP.017

Level: UI-ERROR

Short Syntax: BGP.017 Bad msg len from *neighbor* sprt *sourceport* dprt *destinationport*

Long Syntax: BGP.017 Bad message length received from neighbor *neighbor* source port *sourceport* destination port *destinationport*

Description: The message length is checked when received. This message is printed if the length of the received message is smaller than the expected message header size.

Cause: This is probably caused by some device driver error or defect in the software either with the speaker or the neighbor.

Action: Determine if this happens with other neighbors. If yes, suspect some problem with this router; else, there is probably a problem with the neighbor. This is a serious error that might require information from many sources.

BGP.019

Level: UI-ERROR

Short Syntax: BGP.019 Bad msg type from *neighbor* sprt *sourceport* dprt *destinationport*

Long Syntax: BGP.019 Bad message type from neighbor *neighbor* source port *sourceport* destination port *destinationport*

Description: BGP messages can be only of four types: OPEN, UPDATE, NOTIFICATION, and KEEPALIVE. This message is printed if the type is something other than the ones expected.

Cause: Since message types are among the most basic pieces of BGP information, this is probably the result of a garbled message.

Action: Determine if this happens with other neighbors. If yes, suspect some problem with this router; else, there is probably a problem with the neighbor. This is a serious error that requires information from many sources.

BGP.020

Level: U-INFO

Short Syntax: BGP.020 BGP init

Long Syntax: BGP.020 BGP initialization

Description: This message is printed when BGP has been enabled.

BGP.023

Level: U-INFO

Short Syntax: BGP.023 Nbr *neighbor* disabled or deleted

Long Syntax: BGP.023 Neighbor *neighbor* is disabled or deleted

Description: The neighbor record has been found, but the neighbor is disabled or deleted.

Cause: The user has disabled or deleted the neighbor.

Action: None.

BGP.024

Level: UI-ERROR

Short Syntax: BGP.024 Attr len too long from *neighbor*, len *length*

Long Syntax: BGP.024 Attribute length too long from neighbor *neighbor*, length *length*

Description: The length of the path attributes exceeds the length in the header.

Cause: Either the speaker or the neighbor has garbled the message.

Action: The user should suspect data corruption with the speaker or neighbor. Check the quality of link.

BGP.025

Level: UI-ERROR

Short Syntax: BGP.025 mand attr without trans bit set from *neighbor*, attr *attribute_type*

Long Syntax: BGP.025 mandatory attribute without transitive bit set from neighbor *neighbor*, attribute type *attribute_type*

Description: The neighbor has sent a mandatory attribute with the non-transitive bit set. This is a violation of the specification.

Cause: This is so basic to the protocol that the user would have to suspect some data corruption in the neighbor or the speaker.

Action: The user should suspect data corruption with the speaker or neighbor. Check the quality of link.

BGP.026

Level: UI-ERROR

Short Syntax: BGP.026 Mand attr with partial bit set from *neighbor*, attr *attribute_type*

Long Syntax: BGP.026 Mandatory attribute with partial bit set from neighbor *neighbor*, attribute type *attribute_type*

Description: The neighbor has sent a mandatory attribute with the partial bit set. This is a violation of the specification.

Cause: This is so basic to the protocol that the user would have to suspect some data corruption in the neighbor or the speaker.

Action: The user should suspect data corruption with the speaker or neighbor. Check the quality of link.

BGP.027

Level: UI-ERROR

Short Syntax: BGP.027 Opt non-trans attr with partial bit set from *neighbor*, attr *attribute_type*

Long Syntax: BGP.027 Optional non-transitive attribute with partial bit set from neighbor *neighbor*, attribute *attribute_type*

Description: The neighbor has sent an optional attribute with the partial bit set. This is a violation of the specification.

Cause: This is a basic protocol violation and the user should suspect data corruption in the neighbor or the speaker.

Action: The user should suspect data corruption with the speaker or neighbor. Check the quality of link.

BGP.028

Level: UI-ERROR

Short Syntax: BGP.028 Origin path attr with bad len from *neighbor*, len *length*

Long Syntax: BGP.028 Origin path attribute has bad length from neighbor *neighbor*, length *length*

Description: The origin attribute must be one byte long. This attribute has a different length.

Cause: This is a basic protocol violation and the user should suspect data corruption in the neighbor or the speaker.

Action: The user should suspect data corruption with the speaker or neighbor. Check the quality of link.

BGP.029

Level: UI-ERROR

Short Syntax: BGP.029 Origin path attr with bad type from *neighbor*, origin *origin_type*

Long Syntax: BGP.029 Origin path attribute with bad type from neighbor *neighbor*, origin *origin_type*

Description: The origin attribute contains an unidentified origin type.

Cause: This is a basic protocol violation.

Action: Use a router that adheres to the BGP specification.

BGP.030

Level: UI-ERROR

Short Syntax: BGP.030 Dupl AS in path attr from *neighbor*, pathlen *AS_path_length*

Long Syntax: BGP.030 Duplicate AS in path attribute from neighbor *neighbor*, path length *AS_path_length*

Description: The neighbor has sent an AS path attribute with a duplicate.

Cause: The AS path attribute contains a loop as evidenced by a duplicate AS. A speaker should never advertise a path with a duplicate AS.

Action: The probability of data corruption causing a duplicate is low. The problem may be with the neighbor. Since this is a core function of BGP, the neighbor may be operating with a defective implementation and must be corrected.

BGP.031

Level: UI-ERROR

Short Syntax: BGP.031 Bad next hop attr len from *neighbor*, len *length*

Long Syntax: BGP.031 Next hop attribute with bad length from neighbor *neighbor*, length *length*

Description: The next hop should be the length of an IP address. This attribute has an incorrect length.

Cause: The neighbor has sent a next hop attribute with an incorrect length. This could be the result of data corruption.

Action: If the length field is completely garbled, suspect data corruption with the speaker or the neighbor. If the length field is off by a byte, suspect a protocol violation by the neighbor.

BGP.032

Level: UI-ERROR

Short Syntax: BGP.032 Bad next hop attr from *neighbor*, next hop *next_hop_attribute*

Long Syntax: BGP.032 Bad next hop attribute from neighbor *neighbor*, next hop *next_hop_attribute*

Description: The next hop attribute is of proper length, but has been determined to be incorrect.

Cause: The neighbor has sent a next hop address, which is ours or a subnet address.

Action: If the address is our address, the neighbor is in definite violation of the protocol. If the address is a subnet, the neighbor is probably in violation.

BGP.033

Level: UI-ERROR

Short Syntax: BGP.033 Bad mult exit disc attr len from *neighbor*, len *length*

Long Syntax: BGP.033 Bad mult exit disc attribute length from neighbor *neighbor*, length *length*

Description: The mult exit disc attribute length is incorrect.

Cause: The neighbor has sent a mult exit disc attribute with the incorrect length.

Action: If there is a wide discrepancy between the expected and the received length, suspect data corruption in the speaker or the neighbor; otherwise, if the difference in length is only one, the neighbor is probably in violation of the protocol.

BGP.034

Level: UI_ERROR

Short Syntax: BGP.034 Bad local pref attr len from *neighbor*, len *length*

Long Syntax: BGP.034 Local preference attribute has bad length from neighbor *neighbor*, length *length*

Description: The local preference attribute length is incorrect.

Cause: The neighbor has sent a local preference with an incorrect length.

Action: If there is a wide discrepancy between the expected and the received length, suspect data corruption in the speaker or the neighbor; otherwise, if the difference in length is only one, the neighbor is probably in violation of the protocol.

BGP.035

Level: UI-ERROR

Short Syntax: BGP.035 Bad atom aggr attr len from *neighbor*, len *length*

Long Syntax: BGP.035 Atomic aggregate attribute has bad length from neighbor *neighbor*, length *length*

Description: The atomic aggregate attribute should be of length 0, but has a length different than 0.

Cause: The neighbor has sent an incorrectly formatted atomic aggregate attribute.

Action: If there is a wide discrepancy between the expected and the received length, suspect data corruption in the speaker or the neighbor; otherwise, if the difference in length is only one, the neighbor is probably in violation of the protocol.

BGP.036

Level: UI-ERROR

Short Syntax: BGP.036 Bad aggr attr len from *neighbor*, len *length*

Long Syntax: BGP.036 Aggregator attribute has bad length from neighbor *neighbor* length *length*

Description: The aggregator attribute has an incorrect length.

Cause: The neighbor has sent an aggregator attribute with the incorrect length.

Action: If there is a wide discrepancy between the expected and the received length, suspect data corruption in the speaker or the neighbor; otherwise, if the difference in length is only one, the neighbor is probably in violation of the protocol.

BGP.039

Level: UI-ERROR

Short Syntax: BGP.039 Unrecog well knwn attr from *neighbor*, attr *attribute_type*

Long Syntax: BGP.039 Unrecognized well-known attribute from neighbor *neighbor*, attribute *attribute_type*

Description: The well-known attribute is unrecognized.

Cause: The neighbor has sent a well-known attribute that is unrecognized.

Action: Since this would be a basic protocol violation, the user should suspect data corruption with the speaker or the neighbor.

BGP.040

Level: UI-ERROR

Short Syntax: BGP.040 Dupl attr from *neighbor*, attr *attribute_type*

Long Syntax: BGP.040 Multiple attributes from neighbor *neighbor*, attribute *attribute_type*

Description: Duplicate path attributes were found in the UPDATE message.

Cause: The neighbor has sent an UPDATE message with a duplicate path attribute.

Action: The neighbor should be checked, since this is a protocol violation.

BGP.042

Level: UI-ERROR

Short Syntax: BGP.042 No NLRI in UPDATE from *neighbor*

Long Syntax: BGP.042 No Network Layer Routing Information in UPDATE from neighbor *neighbor*

Description: The UPDATE message had no network layer routing information.

Cause: The neighbor sent an UPDATE message with path attributes but no routing information.

Action: The neighbor should be checked for a protocol violation.

BGP.043

Level: U-INFO

Short Syntax: BGP.043 NLRI *NLRI* rej by ext policy from *neighbor*

Long Syntax: BGP.043 Network Layer Routing Information *NLRI* rejected by external policy from neighbor *neighbor*

Description: The Network Layer Routing Information described by the path attribute has been rejected after applying policy.

Cause: Policy configuration commands have resulted in this NLRI described by the path attribute to be rejected.

Action: None, unless this NLRI should have been included.

BGP.044

Level: U-INFO

Short Syntax: BGP.044 New or updtd RIB entry *NLRI* from *neighbor*

Long Syntax: BGP.044 New or updated RIB entry *NLRI* from neighbor *neighbor*

Description: A NLRI has passed filters and is being put into the Routing Information Base.

Cause: The neighbor has sent an UPDATE message with NLRI and path that is acceptable by external policy rule definitions.

Action: None, unless this NLRI should have been excluded.

BGP.046

Level: U-INFO

Short Syntax: BGP.046 Notify rcvd from *neighbor*, err *error_code*: *sub_code*

Long Syntax: BGP.046 Notify received from *neighbor*, error code *error_code*, subcode *sub_code*

Description: A NOTIFICATION message has been received from the neighbor. This terminates the BGP connection, and usually indicates some kind of error. The error code and subcode can be found in the BGP specification, giving the exact reason for the notification.

BGP.047

Level: U-INFO

Short Syntax: BGP.047 Accept dest *destination* from IP fw tbl

Long Syntax: BGP.047 Destination *destination* from IP forwarding table included

Description: The speaker has just included this destination, as directed by internal policy.

Cause: The internal policy can specifically include destinations.

Action: None, unless the internal policy should have excluded this destination.

BGP.048

Level: UI-ERROR

Short Syntax: BGP.048 BGP spkr unable to get mem

Long Syntax: BGP.048 BGP speaker unable to get memory

Description: BGP was unable to allocate the necessary memory. BGP is unable to run because of this.

Cause: There is a shortage in heap memory, possibly because too many memory intensive forwarders/protocols are running.

Action: Disable unnecesary forwarders/protocols or get more memory.

BGP.049

Level: U-INFO

Short Syntax: BGP.049 Closing conn to *neighbor* sprt *sourceport* dprt *destinationport*; conn collision

Long Syntax: BGP.049 closing connection to neighbor *neighbor* source port *sourceport* destination port *destinationport* because of connection collision

Description: BGP is removing a duplicate connection to this neighbor because of a connection collision.

Cause: Multiple TCP connections can form during the original neighbor connection establishment.

Action: None. Collisions can occur and the BGP RFC describes procedures to decide which connection wins.

BGP.050

Level: U-INFO

Short Syntax: BGP.050 UPDATE(s) sent to *neighbor*, len *message_length*

Long Syntax: BGP.050 UPDATE(s) sent to neighbor *neighbor*, length *message_length*

Description: One or more BGP UPDATE messages are being queued to the given neighbor. This occurs only on topology changes. The length of the entire collection of UPDATE messages is displayed.

BGP.052

Level: U-INFO

Short Syntax: BGP.052 UPDATE rcvd from *neighbor*, len *message_length*

Long Syntax: BGP.052 UPDATE received from neighbor *neighbor*, length *message_length*

Description: BGP UPDATE message of a given length has been received from the given neighbor. This indicates some kind of topology change.

BGP.053

Level: U-INFO

Short Syntax: BGP.053 Del BGP route to *network*

Long Syntax: BGP.053 Deleted BGP route to network *network*

Description: The BGP route to the given network is no longer valid, and has been deleted from the IP routing table.

BGP.055

Level: UI-ERROR

Short Syntax: BGP.055 Ext nbr *neighbor* not on cmn net

Long Syntax: BGP.055 External neighbor *neighbor* is not on common network

Description: External neighbors must share a common network with the router, else the neighbor will be ignored. The neighbor's address on the common network must be configured in the "BGP Config> add neighbor" command.

Cause: May be the neighbor common network address is not configured.

Action: Check the neighbor address configuration.

BGP.056

Level: U-INFO

Short Syntax: BGP.056 OPEN rcvd from *neighbor*

Long Syntax: BGP.056 OPEN received from neighbor *neighbor*

Description: BGP OPEN message has been received from the given neighbor. This indicates that the neighbor wishes to initiate a conversation.

BGP.057

Level: P-TRACE

Short Syntax: BGP.057 KEEPALIVE rcvd from *neighbor*

Long Syntax: BGP.057 KEEPALIVE received from neighbor *neighbor*

Description: BGP KEEPALIVE message has been received from the given neighbor. These are sent and received periodically in order to ensure that the BGP connection is still in tact.

BGP.058

Level: U-INFO

Short Syntax: BGP.058 Notify sent to *neighbor*

Long Syntax: BGP.058 Notify sent to *neighbor*

Description: A NOTIFICATION message has been sent to the neighbor. This terminates the BGP connection, and means that we have encountered an unrecoverable error, probably the reception of bad data from the neighbor. A previously displayed ELS message indicates the exact nature of the error.

BGP.059

Level: P-TRACE

Short Syntax: BGP.059 KEEPALIVE sent to *neighbor*

Long Syntax: BGP.059 KEEPALIVE sent to neighbor *neighbor*

Description: BGP KEEPALIVE message has been sent to the given neighbor. These are sent and received periodically in order to ensure that the BGP connection is still in tact.

BGP.060

Level: U-INFO

Short Syntax: BGP.060 Couldn't add net *network* mask *mask*

Long Syntax: BGP.060 Couldn't add network *network* mask *mask* to routing table

Description: Router unable to add a network that was received in a BGP UPDATE message to its routing table. This is either because the routing table overflowed, or because the network number was badly formed.

BGP.061

Level: U-INFO

Short Syntax: BGP.061 No mem for UPDATE to *neighbor*

Long Syntax: BGP.061 No memory for UPDATE to neighbor *neighbor*

Description: Unable to get memory to send an UPDATE message to peer. Router will continue to retry. If message persists, router may have run out of available memory.

BGP.062

Level: UI-ERROR

Short Syntax: BGP.062 Rej nbr *neighbor*, not in nbr tbl

Long Syntax: BGP.062 External neighbor *neighbor* is not in the neighbor table

Description: External neighbor is trying to establish a BGP connection with this speaker, which does not have the neighbor in the configuration.

Cause: Neighbor parmeters are not configured in both speaker.

Action: Check the neighbor configuration in both speakers.

BGP.063

Level: U-INFO

Short Syntax: BGP.063 Pasv conn exists for *neighbor*; new pasv conn closed

Long Syntax: BGP.063 Passive connection already exists for neighbor *neighbor*; new passive connection is closed

Description: A passive TCP connection already exists for this neighbor, but the neighbor has tried for another passive connection. The new connection will be closed.

BGP.067

Level: UI-ERROR

Short Syntax: BGP.067 Hold tmr exp for *neighbor* clsng conn

Long Syntax: BGP.067 Hold timer expired for neighbor *neighbor*; closing connection

Description: No KEEPALIVE message has been received from this neighbor. Thus, the KEEPALIVE Timer expires and the connection will be closed.

Cause: See description

Action: Make sure neighbor is up and running BGP.

BGP.069

Level: U-INFO

Short Syntax: BGP.069 BGP state change; nbr *neighbor* ev *event* oldst *oldstate* newst *newstate*

Long Syntax: BGP.069 BGP state change; neighbor *neighbor* event *event* old state *oldstate* new state *newstate*

Description: The state of the connection to this neighbor has just changed.

BGP.070

Level: UI-ERROR

Short Syntax: BGP.070 Unexp event; nbr *neighbor* ev *event* st *state*

Long Syntax: BGP.070 Unexpected event; neighbor *neighbor* event *event* state *state*

Description: An event not handled by this BGP implementation has occurred. This indicates a software error, and should be reported to Customer Service.

Cause: See description

Action: See description

BGP.071

Level: UE-ERROR

Short Syntax: BGP.071 Bad aggregate net *aggregate_net* mask *aggregate_mask*

Long Syntax: BGP.071 Bad aggregate net *aggregate_net* mask *aggregate_mask*

Description: An aggregate has been configured that the router cannot use. This is probably due to misconfiguration. The aggregate is ignored.

BGP.072

Level: P-TRACE

Short Syntax: BGP.072 Add NLRI *destination_net* len *destination_mask_len* updt for nbr *neighbor*

Long Syntax: BGP.072 Add NLRI *destination_net* len *destination_mask_len* UPDATE for neighbor *neighbor*

Description: A new Network Layer Reachability Information has been added to the list of NLRIs associated with a particular attribute list in the new UPDATE message being constructed for this neighbor.

BGP.073

Level: P-TRACE

Short Syntax: BGP.073 Wdra NLRI *destination_net* len *destination_mask_len* updt for nbr *neighbor*

Long Syntax: BGP.073 Withdraw NLRI *destination_net* len *destination_mask_len* UPDATE for neighbor *neighbor*

Description: The Network Layer Reachability Information has been added to the list of unfeasible routes in the new UPDATE message being constructed for this neighbor.

BGP.074

Level: UI-ERROR

Short Syntax: BGP.074 Bad hold tim val *timer_value* from *neighbor*

Long Syntax: BGP.074 Received bad hold timer value *timer_value* from neighbor *neighbor*

Description: The speaker received an OPEN message that has unacceptable hold timer value.

Cause: Neighbor sent an OPEN message that has incorrect hold timer value.

Action: Use a router that adheres to the BGP specification.

BGP.075

Level: U-INFO

Short Syntax: BGP.075 Conn Cls to *neighbor*; clsg with notify cease

Long Syntax: BGP.075 Closing connection to neighbor *neighbor* with notification cease

Description: User disabled the neighbor and hence this message.

Cause: User disabled the neighbor and hence this message.

Action: None.

Chapter 19. Bridge Routing (BR)

This chapter describes Bridge Routing (BR) messages. For information on message content and how to use the message, refer to the Introduction.

BR.001

Level: C-INFO

Short Syntax: BR.001 *source_mac*-> *dest_mac* drp, port block/list, nt *network*

Long Syntax: BR.001 Frame from *source_mac* to *dest_mac* dropped, received on blocked or listening port, network *network*

Description: A MAC frame has been received by the hardware, but is being dropped because the port on which it was received is in the "blocking" or "listening" state. Frames are only processed when the port is in the "learning" or "forwarding" state.

Cause: Normal on port bringup.

Action: Wait for port to transition to "learning" and "forwarding" states.

BR.002

Level: P-TRACE

Short Syntax: BR.002 *source_mac*-> *dest_mac* drp, dst same LAN, nt *network*

Long Syntax: BR.002 Frame from *source_mac* to *dest_mac* dropped, destination on same LAN, network *network*

Description: A MAC frame has been received whose destination address is known to be on the same side of the bridge as the packet came from. It is dropped by the filtering logic since it does not need to be bridged. Note that this event is not counted by ELS for performance reasons. A counter is kept in ASRT, it is the ''Dropped, dest addr filtering'' entry in the ASRT>LIST SOURCE-ROUTE COUNTERS and ASRT>LIST TRANSPARENT COUNTERS commands.

Cause: Normal local traffic on network.

BR.003

Level: UE-ERROR

Short Syntax: BR.003 *source_mac*-> *dest_mac*, brdg encap for rout prot IPX (802.3), drp, nt *network*

Long Syntax: BR.003 Frame from *source_mac* to *dest_mac*, WAN bridge encapsulation for routed protocol IPX (802.3), dropped, network *network*

Description: A frame has been received over a WAN interface in the Ethernet bridge encapsulation, but the IPX protocol (in 802.3 encapsulation) is routed by this node.

Cause: Configuration error at local or remote node with respect to bridging or routing particular protocols over the WAN link.

Action: All hosts on a WAN link must agree on whether to bridge or route a given protocol. Reconfigure as appropriate.

BR.004

Level: P-TRACE

Short Syntax: BR.004 *source_mac*-> *dest_mac* prt IPX (802.3) filt, drp, nt *network*

Long Syntax: BR.004 Frame from *source_mac* to *dest_mac*, protocol IPX (802.3) filtered, dropped, network *network*

Description: A frame has been received for the IPX protocol (in 802.3 encapsulation), but the IPX protocol is being administratively filtered by the bridge. The frame will be dropped.

Cause: Frame received for filtered protocol.

BR.005

Level: P-TRACE

Short Syntax: BR.005 *source_mac*-> *dest_mac* SNAP *protocol_identifier* filt, drp, nt *network*

Long Syntax: BR.005 Frame from *source_mac* to *dest_mac*, IEEE 802 SNAP Protocol Identifier *protocol_identifier* filtered, dropped, network *network*

Description: A frame has been received for the specified IEEE 802 Subnetwork Access Protocol (SNAP) Protocol Identifier (PID), but this PID is being administratively filtered by the bridge. The frame will be dropped.

Cause: Frame received for filtered protocol.

BR.006

Level: U-TRACE

Short Syntax: BR.006 Unreg dst *source_mac*-> *dest_mac* SNAP *protocol_identifier*, drp, nt *network*

Long Syntax: BR.006 Frame from *source_mac* to unregistered destination MAC address *dest_mac*, IEEE

802 SNAP Protocol Identifier *protocol_identifier*, dropped, network *network*

Description: A frame has been received for the IEEE 802 Subnetwork Access Protocol (SNAP) Protocol Identifier (PID) which corresponds with an enabled protocol, but the destination MAC address is not registered in the bridge. The frame will be dropped.

Cause: If dest_mac is a unicast address, a station on the LAN is sending frames for this protocol to the wrong next hop MAC address.

Action: Correct action of remote station.

Cause: If dest_mac is a multicast address, a station on the LAN may be sending frames to the wrong multicast address, or perhaps just to one that this router does not have enabled. Depending on the protocol, this may or may not be an error.

Action: Correct action of remote station, if necessary.

BR.007

Level: P-TRACE

Short Syntax: BR.007 *source_mac*-> *dest_mac* SNAP *protocol_identifier*, endnode, nt *network*

Long Syntax: BR.007 Frame from *source_mac* to *dest_mac*, IEEE 802 SNAP Protocol Identifier *protocol_identifier* for endnode protocol, network *network*

Description: A multicast frame has been received for the IEEE 802 Subnetwork Access Protocol (SNAP) Protocol Identifier (PID) which corresponds with an endnode protocol. The frame will be both bridged and locally processed by the endnode protocol.

BR.008

Level: UE-ERROR

Short Syntax: BR.008 *source_mac*-> *dest_mac*, brdg encap for rout SNAP *protocol_identifier*, drp, nt *network*

Long Syntax: BR.008 Frame from *source_mac* to *dest_mac*, WAN bridge encapsulation for routed IEEE 802 SNAP Protocol Identifier *protocol_identifier*, dropped, network *network*

Description: An IEEE 802.2 frame has been received over a WAN interface in a bridge encapsulation, but its IEEE 802 Subnetwork Access Protocol (SNAP) Protocol Identifier (PID) is one that is being routed by this node. The frame will be dropped.

Cause: Configuration error at local or remote node with respect to bridging or routing particular protocols over the WAN link.

Action: All hosts on a WAN link must agree on whether to bridge or route a given protocol. Reconfigure as appropriate.

BR.009

Level: UE-ERROR

Short Syntax: BR.009 BPDU *source_mac*-> *dest_mac*, wrng dst, drp, nt *network*

Long Syntax: BR.009 IEEE 802.1D BPDU *source_mac* to *dest_mac*, wrong destination, dropped, network *network*

Description: An IEEE 802.1D Bridge Protocol Data Unit (BPDU) was received at the wrong destination address. It is supposed to be addressed to a particular multicast address. The BPDU will be dropped.

Cause: Programming error at remote node.

Action: Correct software in remote node.

Cause: Node speaking IBM Token-Ring proprietary source-routing spanning tree protocol, which uses a non-standard destination address for BPDUs.

Action: Ignore message or reconfigure source-routing node.

BR.010

Level: P-TRACE

Short Syntax: BR.010 *source_mac*-> *dest_mac* DSAP *destination_service_access_point* filt, drp, nt *network*

Long Syntax: BR.010 Frame from *source_mac* to *dest_mac*, IEEE 802.2 DSAP *destination_service_access_point* filtered, dropped, network *network*

Description: A frame has been received for the specified IEEE 802.2 Destination Service Access Point (DSAP), but this DSAP is being administratively filtered by the bridge. The frame will be dropped.

Cause: Frame received for filtered protocol.

BR.011

Level: U-TRACE

Short Syntax: BR.011 Unreg dst *source_mac*-> *dest_mac* DSAP *destination_service_access_point*, drp, nt *network*

Long Syntax: BR.011 Frame from *source_mac* to unregistered destination MAC address *dest_mac*, IEEE 802.2 DSAP *destination_service_access_point*, dropped, network *network*

Description: A frame has been received for the IEEE 802.2 Destination Service Access Point (DSAP) which corresponds with an enabled protocol, but the destination MAC address is not registered in the bridge. The frame will be dropped.

Cause: If dest_mac is a unicast address, a station on the LAN is sending frames for this protocol to the wrong next hop MAC address.

Action: Correct action of remote station.

Cause: If dest_mac is a multicast address, a station on the LAN may be sending frames to the wrong multicast address, or perhaps just to one that this router does not have enabled. Depending on the protocol, this may or may not be an error.

Action: Correct action of remote station, if necessary.

BR.012

Level: P-TRACE

Short Syntax: BR.012 *source_mac*-> *dest_mac* DSAP *destination_service_access_point*, endnode, nt *network*

Long Syntax: BR.012 Frame from *source_mac* to *dest_mac*, IEEE 802.2 DSAP *destination_service_access_point* for endnode protocol, network *network*

Description: A multicast frame has been received for the IEEE 802.2 Destination Service Access Point (DSAP) which corresponds with an endnode protocol. The frame will be both bridged and locally processed by the endnode protocol.

BR.013

Level: UE-ERROR

Short Syntax: BR.013 *source_mac*-> *dest_mac*, brdg encap for rout DSAP *destination_service_access_point*, drp, nt *network*

Long Syntax: BR.013 Frame from *source_mac* to *dest_mac*, WAN bridge encapsulation for routed IEEE 802.2 DSAP *destination_service_access_point*, dropped, network *network*

Description: An IEEE 802.2 frame has been received over a WAN interface in a bridge encapsulation, but its IEEE 802.2 Destination Service Access Point (DSAP) is one that is being routed by this node. The frame will be dropped.

Cause: Configuration error at local or remote node with respect to bridging or routing particular protocols over the WAN link.

Action: All hosts on a WAN link must agree on whether to bridge or route a given protocol. Reconfigure as appropriate.

BR.014

Level: P-TRACE

Short Syntax: BR.014 *source_mac*-> *dest_mac* Etype *Ethernet_type* filt, drp, nt *network*

Long Syntax: BR.014 Frame from *source_mac* to *dest_mac*, Ethernet type *Ethernet_type* filtered, dropped, network *network*

Description: A frame has been received for the specified Ethernet type, but this type is being administratively filtered by the bridge. The frame will be dropped.

Cause: Frame received for filtered protocol.

BR.015

Level: U-TRACE

Short Syntax: BR.015 Unreg dst *source_mac*-> *dest_mac* Etype *Ethernet_type*, drp, nt *network*

Long Syntax: BR.015 Frame from *source_mac* to unregistered destination MAC address *dest_mac*, Ethernet type *Ethernet_type*, dropped, network *network*

Description: A frame has been received for the Ethernet type which corresponds with an enabled protocol, but the destination MAC address is not registered in the bridge. The frame will be dropped.

Cause: If dest_mac is a unicast address, a station on the LAN is sending frames for this protocol to the wrong next hop MAC address.

Action: Correct action of remote station.

Cause: If dest_mac is a multicast address, a station on the LAN may be sending frames to the wrong multicast address, or perhaps just to one that this router does not have enabled. Depending on the protocol, this may or may not be an error.

Action: Correct action of remote station, if necessary.

BR.016

Level: P-TRACE

Short Syntax: BR.016 *source_mac*-> *dest_mac* Etype *Ethernet_type*, endnode, nt *network*

Long Syntax: BR.016 Frame from *source_mac* to *dest_mac*, Ethernet type *Ethernet_type* for endnode protocol, network *network*

Description: A multicast frame has been received for the Ethernet type which corresponds with an endnode protocol. The frame will be both bridged and locally processed by the endnode protocol.

BR.017

Level: UE-ERROR

Short Syntax: BR.017 *source_mac*-> *dest_mac*, brdg encap for rout Etype *Ethernet_type*, drp, nt *network*

Long Syntax: BR.017 Frame from *source_mac* to *dest_mac*, WAN bridge encapsulation for routed Ethernet type *Ethernet_type*, dropped, network *network*

Description: An Ethernet frame has been received over a WAN interface in the Ethernet bridge encapsulation, but its Ethernet type is one that is being routed by this node. The frame will be dropped.

Cause: Configuration error at local or remote node

with respect to bridging or routing particular protocols over the WAN link.

Action: All hosts on a WAN link must agree on whether to bridge or route a given protocol. Reconfigure as appropriate.

BR.018

Level: P-TRACE

Short Syntax: BR.018 SR *source_mac*-> *dest_mac* DSAP *destination_service_access_point* filt, drp, nt *network*

Long Syntax: BR.018 Source-routed frame from *source_mac* to *dest_mac*, IEEE 802.2 DSAP *destination_service_access_point* filtered, dropped, network *network*

Description: A source-routed frame has been received for the specified IEEE 802.2 Destination Service Access Point (DSAP), but this DSAP is being administratively filtered by the bridge. The frame will be dropped.

Cause: Frame received for filtered protocol.

BR.019

Level: U-TRACE

Short Syntax: BR.019 SR unreg dst *source_mac*-> *dest_mac* DSAP *destination_service_access_point*, drp, nt *network*

Long Syntax: BR.019 Source-routed frame from *source_mac* to unregistered destination MAC address *dest_mac*, IEEE 802.2 DSAP *destination_service_access_point*, dropped, network *network*

Description: A source-routed frame has been received for the IEEE 802.2 Destination Service Access Point (DSAP) which corresponds with an enabled protocol, but the destination MAC address is not registered in the bridge. The frame will be dropped.

Cause: If dest_mac is a unicast address, a station on the LAN is sending frames for this protocol to the wrong next hop MAC address.

Action: Correct action of remote station.

Cause: If dest_mac is a multicast address, a station on the LAN may be sending frames to the wrong multicast address, or perhaps just to one that this router does not have enabled. Depending on the protocol, this may or may not be an error.

Action: Correct action of remote station, if necessary.

BR.020

Level: P-TRACE

Short Syntax: BR.020 SR *source_mac*-> *dest_mac* DSAP *destination_service_access_point*, endnode, nt *network*

Long Syntax: BR.020 Source-routed frame from *source_mac* to *dest_mac*, IEEE 802.2 DSAP *destination_service_access_point* for endnode protocol, network *network*

Description: A multicast source-routed frame has been received for the IEEE 802.2 Destination Service Access Point (DSAP) which corresponds with an endnode protocol. The frame will be both bridged and locally processed by the endnode protocol.

BR.021

Level: P-TRACE

Short Syntax: BR.021 SR *source_mac*-> *dest_mac* SNAP *protocol_identifier* filt, drp, nt *network*

Long Syntax: BR.021 Source-routed frame from *source_mac* to *dest_mac*, IEEE 802 SNAP Protocol Identifier *protocol_identifier* filtered, dropped, network *network*

Description: A source-routed frame has been received for the specified IEEE 802 Subnetwork Access Protocol (SNAP) Protocol Identifier (PID), but this PID is being administratively filtered by the bridge. The frame will be dropped.

Cause: Frame received for filtered protocol.

BR.022

Level: U-TRACE

Short Syntax: BR.022 SR unreg dst *source_mac*-> *dest_mac* SNAP *protocol_identifier*, drp, nt *network*

Long Syntax: BR.022 Source-routed frame from *source_mac* to unregistered destination MAC address *dest_mac*, IEEE 802 SNAP Protocol Identifier *protocol_identifier*, dropped, network *network*

Description: A source-routed frame has been received for the IEEE 802 Subnetwork Access Protocol (SNAP) Protocol Identifier (PID) which corresponds with an enabled protocol, but the destination MAC address is not registered in the bridge. The frame will be dropped.

Cause: If dest_mac is a unicast address, a station on the LAN is sending frames for this protocol to the wrong next hop MAC address.

Action: Correct action of remote station.

Cause: If dest_mac is a multicast address, a station on the LAN may be sending frames to the wrong multicast address, or perhaps just to one that this router does not have enabled. Depending on the protocol, this may or may not be an error.

Action: Correct action of remote station, if necessary.

BR.023

Level: P-TRACE

Short Syntax: BR.023 SR *source_mac*-> *dest_mac* SNAP *protocol_identifier*, endnode, nt *network*

Long Syntax: BR.023 Source-routed frame from *source_mac* to *dest_mac*, IEEE 802 SNAP Protocol Identifier *protocol_identifier* for endnode protocol, network *network*

Description: A source-routed multicast frame has been received for the IEEE 802 Subnetwork Access Protocol (SNAP) Protocol Identifier (PID) which corresponds with an endnode protocol. The frame will be both bridged and locally processed by the endnode protocol.

BR.025

Level: P-TRACE

Short Syntax: BR.025 *source_mac*-> *dest_mac* drp, src add flt, nt *network*

Long Syntax: BR.025 Frame from *source_mac* to *dest_mac* dropped, source address filtered, network *network*

Description: A MAC frame has been received by the hardware, but is being dropped because the source MAC address is being administratively filtered by the bridge. The frame will be dropped.

Cause: Receipt of frame whose source MAC address matches the destination filter.

BR.026

Level: P-TRACE

Short Syntax: BR.026 SR *source_mac*-> *dest_mac* drp, dst add flt, nt *network*

Long Syntax: BR.026 Frame from *source_mac* to *dest_mac* dropped, destination address filtered, network *network*

Description: A source-routed MAC frame has been received by the hardware, but is being dropped because the destination MAC address is being administratively filtered by the bridge. The frame will be dropped.

Cause: Receipt of frame whose destination MAC address matches the destination filter.

BR.027

Level: P-TRACE

Short Syntax: BR.027 SR *source_mac*-> *dest_mac* drp, src add flt, nt *network*

Long Syntax: BR.027 Frame from *source_mac* to *dest_mac* dropped, source address filtered, network *network*

Description: A source-routed MAC frame has been received by the hardware, but is being dropped because the source MAC address is being administratively filtered by the bridge. The frame will be dropped.

Cause: Receipt of frame whose source MAC address matches the destination filter.

BR.028

Level: UI-ERROR

Short Syntax: BR.028 No buf for endnode bridge, *source_mac*-> *dest_mac*, nt *network*, not bridged

Long Syntax: BR.028 No buffer to copy packet for endnode bridge and process, from *source_mac* to *dest_mac*, network *network*, not bridged

Description: A Multicast frame has been received for an endnode protocol that is both bridged and locally processed. There was no buffer to make two copies of the frame for both types of processing, so it will not be bridged, only locally processed.

Cause: Severe packet buffer shortage.

Action: Check memory statistics in GWCON to verify packet buffer level. If possible, make routing or bridging tables smaller. If tables cannot be made smaller, increase memory size.

Cause: Traffic peak using all available buffers.

Action: This is the problem if this message occurs very infrequently.

BR.029

Level: C-TRACE

Short Syntax: BR.029 NB inp pkt fltd - *source_mac*-> *dest_mac*, prt *port*, nt *network*

Long Syntax: BR.029 NETBIOS Input Packet Filtered - *source_mac*-> *dest_mac*, port *port*, network *network*

Description: A NETBIOS packet has matched the criteria specified in a NETBIOS Filter configuration record. The packet is dropped.

BR.030

Level: U-TRACE

Short Syntax: BR.030 Rcvd tkr brg pkt but no tkr hnd

Long Syntax: BR.030 Received tkr bridge packet over WAN, but router has no handler to process it.

Description: A remote router sent a packet over a WAN bridge port to the local router, the frame was in Token-Ring format, but the local router does not contain a handler for Token-Ring frames. The packet was dropped.

Chapter 20. Bridging Broadcast Manager (BBCM)

This chapter describes Bridging Broadcast Manager (BBCM) messages. For information on message content and how to use the message, refer to the Introduction.

BBCM.001

Level: U_INFO

Short Syntax: BBCM.001 *instance_str*initlzd

Long Syntax: BBCM.001 *instance_str*initialized

Description: Bridging Broadcast Manager has been initialized

BBCM.002

Level: U_INFO

Short Syntax: BBCM.002 *instance_str*HALTED

Long Syntax: BBCM.002 *instance_str*HALTED

Description: Bridging Broadcast Manager has been halted. No protocols are active

BBCM.003

Level: U_INFO

Short Syntax: BBCM.003 *instance_str*STARTED/RESTARTED prtcl *protocol_name*, age out= *age_out* min

Long Syntax: BBCM.003 *instance_str*STARTED/RESTARTED protocol *protocol_name*, age out= *age_out* min

Description: BBCM has been started (or restarted) for the given protocol

BBCM.004

Level: U_INFO

Short Syntax: BBCM.004 *instance_str*STOPPED prtcl *protocol_name*

Long Syntax: BBCM.004 *instance_str*STOPPED protocol *protocol_name*

Description: BBCM has been stopped for the given protocol Frames will not be processed by BBCM for the protocol, existing protocol entries will be aged out over time

BBCM.005

Level: U_INFO

Short Syntax: BBCM.005 *instance_str*SHUT DOWN BBCM for prtcl *protocol_name*

Long Syntax: BBCM.005 *instance_str*SHUT DOWN BBCM for protocol *protocol_name*

Description: BBCM has been shut down for the given protocol. Frames will not be processed by BBCM for the protocol, all existing protocol entries have been deleted. This is likely a result of BBCM running out of memory for adding additional protocol entries. BBCM's memory is now free for other functions to use.

BBCM.006

Level: U_INFO

Short Syntax: BBCM.006 *instance_str*deleted all *protocol_name* prtcl entries

Long Syntax: BBCM.006 *instance_str*deleted all entries for protocol *protocol_name*

Description: All protocol entries for the given protocol were deleted.

BBCM.007

Level: UI_ERROR

Short Syntax: BBCM.007 *instance_str*add to *protocol_name* cache failed. prtcl CB alloc err

Long Syntax: BBCM.007 *instance_str*add to *protocol_name* cache failed. protocol control block allocation error

Description: BBCM could not add a new protocol address because an error occurred while trying to allocate memory for the protocol control block. Given the lack of availability of memory, BBCM will shut down.

Action: Contact your customer service representative.

BBCM.008

Level: C_INFO

Short Syntax: BBCM.008 *instance_str*added *protocol_type_string protocol_address* on MAC addr x *MAC_address* to cache

Long Syntax: BBCM.008 *instance_str*added *protocol_type_string protocol_address* on MAC address x *MAC_address* to cache

Description: BBCM added a protocol address with the

given MAC address to its cache.

BBCM.009

Level: C_INFO

Short Syntax: BBCM.009 *instance_str*aged *protocol_type_string protocol_address* on MAC addr x *MAC_address* from cache

Long Syntax: BBCM.009 *instance_str*aged *protocol_type_string protocol_address* on MAC address x *MAC_address* from cache

Description: BBCM aged out the given protocol address on the given MAC address from its cache.

BBCM.010

Level: C_INFO

Short Syntax: BBCM.010 *instance_str*set *protocol_type_string protocol_address* age to *age*

Long Syntax: BBCM.010 *instance_str*set *protocol_type_string protocol_address* age to *age*

Description: The given protocol address age was set to the given age.

BBCM.011

Level: U_INFO

Short Syntax: BBCM.011 *instance_str*Warning: MAC addr x *MAC_address* replaced MAC addr x *MAC_address* for *protocol_type_string protocol_address*

Long Syntax: BBCM.011 *instance_str*Warning: MAC address x *MAC_address* replaced MAC address x *MAC_address* for *protocol_type_string protocol_address*

Description: BBCM has discovered that two MAC addresses are using the same protocol address. The first MAC address displayed was detected more recently and will now be associated with the protocol address.

Action: This may be a misconfiguration of one of the devices.

BBCM.012

Level: U_INFO

Short Syntax: BBCM.012 *instance_str*Warning: MAC addr x *MAC_address* conflicts w/ Permanent Entry MAC addr x *MAC_address*, *protocol_type_string protocol_address*

Long Syntax: BBCM.012 *instance_str*Warning: MAC address x *MAC_address* conflicts with Permanent Entry MAC address x *MAC_address*, *protocol_type_string protocol_address*

Description: BBCM has detected that the first MAC address is using the same protocol address as the Permanent Entry shown. The Permanent Entry remains intact.

Action: This may be a misconfiguration of a device, or the Permanent Entry.

BBCM.013

Level: UI_ERROR

Short Syntax: BBCM.013 *instance_str*INIT FAILED

Long Syntax: BBCM.013 *instance_str*INITIALIZATION FAILED

Description: Bridging Broadcast Manager initialization has failed. An error occurred while trying to allocate memory for BBCM initialization.

BBCM.014

Level: UI_ERROR

Short Syntax: BBCM.014 *instance_str*ERROR STARTING PROTOCOL *protocol_name*

Long Syntax: BBCM.014 *instance_str*ERROR STARTING PROTOCOL *protocol_name*

Description: Bridging Broadcast Manager for the given protocol could not be started successfully.

BBCM.015

Level: UI_ERROR

Short Syntax: BBCM.015 *instance_str*No assoc. Super ELAN

Long Syntax: BBCM.015 *instance_str*No associated Super ELAN exists

Description: A Bridging Broadcast Manager request was made but the associated Super ELAN could not be found.

Action: Contact your customer service representative.

Chapter 21. Channel Associated Signaling

This chapter describes Channel Associated Signaling messages. For information on message content and how to use the message, refer to the Introduction.

CAS.001

Level: C-INFO

Short Syntax: CAS.001 Adapter CAS code initialized, protocol *net* nt *protocol* int /

Long Syntax: CAS.001 Adapter CAS code initialized, protocol *net* nt *protocol* int /

Description: Adapter CAS code initialized for specified protocol.

CAS.002

Level: C-INFO

Short Syntax: CAS.002 Adapter CAS code reset, net *net* int /

Long Syntax: CAS.002 Adapter CAS code reset, net *net* int /

Description: Adapter CAS code reset.

CAS.003

Level: C-INFO

Short Syntax: CAS.003 Chn *channel* FSM st *state1* ev *event -> state2* nt *network ID*

Long Syntax: CAS.003 Channel *channel* FSM transition occurred: old state *state1*, event *event*, new state *state2* on network *network ID*.

Description: An FSM transition occurred.

CAS.004

Level: C-INFO

Short Syntax: CAS.004 Chn *channel* FSM st *state1* ev *event -> state2*

Long Syntax: CAS.004 Channel *channel* FSM transition occurred: old state *state1*, event *event*, new state *state2*.

Description: An FSM transition occurred.

Chapter 22. Component Not Present Functions (NOT)

This chapter describes Component Not Present Functions (NOT) messages. For information on message content and how to use the message, refer to the Introduction.

NOT.001

Level: UINFO

Short Syntax: NOT.001 *source_net/ source_node -> destination_net/ destination_node* nt *network* ign

Long Syntax: NOT.001 *source_net/ source_node -> destination_net/ destination_node* net *network* ignored

Description: An AppleTalk packet was recognized but ignored because AppleTalk forwarding is not in this load.

NOT.004

Level: UINFO

Short Syntax: NOT.004 DECnet pkt ign

Long Syntax: NOT.004 DECnet packet ignored, no DECnet forwarder

Description: A DECnet packet was received, but no DECnet forwarder is installed in the gateway.

NOT.005

Level: UINFO

Short Syntax: NOT.005 dsc pkt *source_ip_address -> destination_ip_address* nt *Network ID* no IP

Long Syntax: NOT.005 Discarded packet from *source_ip_address* for *destination_ip_address* net *Network ID*, no IP forwarder

Description: This message is generated by the fake IP forwarder for each packet which is received.

Cause: Received IP packet, but no IP forwarder.

NOT.007

Level: UINFO

Short Syntax: NOT.007 dsc pkt *source_vines_network*: *source_vines_subnet -> destination_vines_network*: *destination_vines_subnet* nt *Network ID* no IP

Long Syntax: NOT.007 Discarded packet from *source_vines_network*: *source_vines_subnet* for *destination_vines_network*: *destination_vines_subnet* net *Network ID*, no VINES forwarder

Description: This message is generated by the fake VINES forwarder for each packet which is received when VINES is not enabled on the router.

NOT.008

Level: UINFO

Short Syntax: NOT.008 FAKE: pkt dscrd frm hst *source_address*

Long Syntax: NOT.008 FAKE: packet discarded from host *source_address*

Description: This message is generated by the SNMP fake-out routine.

Cause: An SNMP packet arrived and the router does not have SNMP support.

NOT.009

Level: UINFO

Short Syntax: NOT.009 FAKE: EGP neighbor *IP_address* lost

Long Syntax: NOT.009 FAKE: EGP neighbor *IP_address* lost

Description: This message is generated by the SNMP, EGP neighbor loss fake-out routine.

Cause: EGP tried to generate a neighbor loss event but the router does not have SNMP installed.

NOT.011

Level: UINFO

Short Syntax: NOT.011 Bridge *source_mac-> dest_mac*, no fwd, nt *network*

Long Syntax: NOT.011 Bridge frame from *source_mac* to *dest_mac*, no forwarder, network *network*

Description: Bridge frame received, but there is no bridging available in this load. The frame will be ignored.

Cause: Receiving a frame to 802.2 destination SAP 42.

NOT.014

Level: UINFO

Short Syntax: NOT.014 *source_net/ source_node -> dest_net/ dest_node* ign

Long Syntax: NOT.014 Packet from *source_net/ source_node* for *dest_net/ dest_node* ignored

Description: An IPX packet arrived on a network and the IPX forwarder is not installed.

NOT.015

Level: UINFO

Short Syntax: NOT.015 disc frm *src_SRLY_addr*H -> *dst_SRLY_addr*H nt *networkID*

Long Syntax: NOT.015 discarded frame with source addr *src_SRLY_addr*H and destination addr *dst_SRLY_addr*H on network *networkID*

Description: A frame had been discarded due to SDLC relay not configured on interface noted.

Cause: The null or fake forwarder is configured on the interface, all received SDLC relay frames are discarded.

NOT.016

Level: UINFO

Short Syntax: NOT.016 dsc pkt *source_ip_address* -> *destination_ip_address* nt *Network ID* no IPSec

Long Syntax: NOT.016 Discarded packet from *source_ip_address* for *destination_ip_address* net *Network ID*, no IP Security

Description: This message is generated by the IPSec stubs for each packet which is received for IP Security.

Cause: Received IP packet for IPSec, but no IP Security.

NOT.017

Level: UINFO

Short Syntax: NOT.017 no NAT to trans pkt *source_ip_address* -> *destination_ip_address* Dir= *direction*

Long Syntax: NOT.017 Did not translate packet from *source_ip_address* to *destination_ip_address* direction *direction*, no NAT

Description: This message is generated by the NAT stub for each packet which is received for NAT.

Cause: Received IP packet for NAT, but no NAT.

NOT.018

Level: UINFO

Short Syntax: NOT.018 dsc pkt *source_ip_address* -> *destination_ip_address* nt *Network ID* no IPSec

Long Syntax: NOT.018 Discarded packet from *source_ip_address* for *destination_ip_address* net *Network ID*, no IP Security for IPv6

Description: This message is generated by the IPSec stubs for each IPv6 packet which is received for IP Security.

Cause: Received IPv6 packet for IPSec, but no IP Security.

NOT.019

Level: UINFO

Short Syntax: NOT.019 dsc pkt *source_ip_address* -> *destination_ip_address* nt *Network ID* no DiffServ

Long Syntax: NOT.019 Discarded packet from *source_ip_address* for *destination_ip_address* net *Network ID*, no DiffServ

Description: This message is generated by the DiffServ stubs for each packet which is received for DiffServ action.

Cause: Received IP packet for DiffServ, but no DiffServ.

Chapter 23. CPU Utilization Monitor (PERF)

This chapter describes CPU Utilization Monitor (PERF) messages. For information on message content and how to use the message, refer to the Introduction.

PERF.001

Level: ALWAYS

Short Syntax: PERF.001 CPU Loading *iob= packet load factor* Max CPU Loading %%= %3u

Long Syntax: PERF.001 CPU Loading *iob= packet load factor* Max CPU Loading %%= %3u

Description: CPU Loading (% of cpu packet handling ability)

PERF.002

Level: ALWAYS

Short Syntax: PERF.002 CPU Util *iob= utilization* Max CPU Util %%= %3u

Long Syntax: PERF.002 Processor *iob* Utilization= *utilization* Max Processor %% Utilization= %3u

Description: CPU utilization (non-linear with respect to packet load)

PERF.003

Level: ALWAYS

Short Syntax: PERF.003 RX Packets Dropped= *iob* TX Packets Dropped= *Inboud packets dropped by router*

Long Syntax: PERF.003 Inbound Packets Dropped= *iob* Outbound Packets Dropped= *Inboud packets dropped by router*

Description: Output of the Packet Statistics monitor

PERF.004

Level: ALWAYS

Short Syntax: PERF.004 *String*

Long Syntax: PERF.004 *String*

Description: This ELS is for the TN Testool only. Testtool only enabled with DEBUG compile

Chapter 24. Data Compression Engines (COMP)

This chapter describes Data Compression Engines (COMP) messages. For information on message content and how to use the message, refer to the Introduction.

COMP.005

Level: UE-ERROR

Short Syntax: COMP.005 *algo,* bad FCS nt *network ID*

Long Syntax: COMP.005 *algo,* bad FCS, net *network ID*.

Description: Decompressor found packet was corrupt on input, a bad CRC or similar sort of check value was invalid.

COMP.008

Level: UE-ERROR

Short Syntax: COMP.008 LZS_Decomp returned *got*.

Long Syntax: COMP.008 LZS_Decomp returned *got*.

Description: Stacker decomp returned something faulty.

COMP.009

Level: UE-ERROR

Short Syntax: COMP.009 *alg,*/compress err *rc,* doing *doing,*, nt *network ID*

Long Syntax: COMP.009 *alg,*/compress error *rc,* doing *doing,* on network *network ID*

Description: Compressor returned an error code. The "doing" parameter indicates what the compressor was working on.

COMP.010

Level: UE-ERROR

Short Syntax: COMP.010 *alg,*/decompress err *rc,* doing *doing,*, nt *network ID*

Long Syntax: COMP.010 *alg,*/decompress error *rc,* doing *doing,* on network *network ID*

Description: Decompressor returned an error code. The "doing" parameter indicates what the decompressor was working on.

COMP.011

Level: UE-ERROR

Short Syntax: COMP.011 *alg,* err nobuf net *network ID*

Long Syntax: COMP.011 *alg,* error, can't get buffer on network *network ID*

Description: Compression routine couldn't obtain work buffer.

COMP.012

Level: P-TRACE

Short Syntax: COMP.012 *alg,* nocomp cc *cc,* pktlen *pktlen,* cmplen *cmplen,* net *network ID*

Long Syntax: COMP.012 *alg,* nocompress cond code *cc,*, pkt-len *pktlen,* -> cmp-len *cmplen,*, on network *network ID*

Description: Packet was incompressible.

COMP.013

Level: P-TRACE

Short Syntax: COMP.013 *alg,* cmp: pkt len *pktlen,* -> send len *cmplen,*, net *network ID*

Long Syntax: COMP.013 alg *alg,* compress: original pkt len *pktlen,*, compressed pkt len *cmplen,*, on network *network ID*

Description: Per-packet trace compression results.

COMP.014

Level: P-TRACE

Short Syntax: COMP.014 *alg,* exp: pkt.len *pktlen,* <- recv len *cmplen,*, net *network ID*

Long Syntax: COMP.014 alg *alg,* expand: result pkt len *pktlen,*, received pkt len *cmplen,*, on network *network ID*

Description: Per-packet trace expansion results.

COMP.019

Level: UI-ERROR

Level: OOM

Short Syntax: COMP.019 Init fail: no mem for contexts; CMP disabled.

Long Syntax: COMP.019 Unable to allocate memory for compression contexts.

Description: The compression system was not able to

allocate memory for the configured number of compression ″contexts″. The compression subsystem is inoperative.

Cause: The system does not have enough RAM, or too many contexts were configured.

Action: Reduce the number of contexts which are configured to be allocated, or reduce the memory requirements used by other parts of the system. Otherwise, must upgrade the amount of RAM in the router.

COMP.020

Level: C-INFO

Short Syntax: COMP.020 CMP Init: max contexts = 0, CMP disabled.

Long Syntax: COMP.020 No compression contexts were configured. Compression is disabled.

Description: Compression has never been configured, or the number of contexts to allocate was set to zero. No contexts were allocated; and compression is disabled.

COMP.021

Level: C_INFO

Short Syntax: COMP.021 CMP Init: allocated *num_ctx* contexts.

Long Syntax: COMP.021 Compression subsystem allocated *num_ctx* contexts.

Description: Space for the indicated number of contexts was allocated.

COMP.022

Level: CI_ERROR

Short Syntax: COMP.022 No ctx available for net *network ID* channel *channel*

Long Syntax: COMP.022 No context available for network *network ID* channel *channel*.

Description: A net tried to allocate a compression context, but none was available. This normally means the maximum number of configured contexts has been reached.

COMP.023

Level: C-TRACE

Short Syntax: COMP.023 Autofreeing context # *context* owned by nt *network ID*.

Long Syntax: COMP.023 Autofreeing context # *context* owned by network *network ID*.

Description: A net allocated a compression context, but then did not free the context when the net went down. The compression utility library detected this and freed the context on its own.

COMP.024

Level: C-TRACE

Short Syntax: COMP.024 Allocated context # *context_id* nt *network ID* channel *channel*.

Long Syntax: COMP.024 Allocated context # *context_id* for network *network ID* channel *channel*.

Description: An interface allocated a compression context.

COMP.025

Level: C-TRACE

Short Syntax: COMP.025 Freed context # *context_id* nt *network ID* channel *channel*.

Long Syntax: COMP.025 Freed context # *context_id* for network *network ID* channel *channel*.

Description: An interface freed a compression context.

COMP.026

Level: C-TRACE

Short Syntax: COMP.026 Cmp net dn nt *network ID*.

Long Syntax: COMP.026 Compression observed netdown on monitored network *network ID*.

Description: The compression system detected a net going down for a net it is actively monitoring. The compression system will check for any unfreed contexts held by the net and release them.

Panic CMP_NO_MEMORY

Short Syntax: Compression subsystem couldn’t allocate required memory.

Description: The compression subsystem could not allocate memory required for its normal operation. This is a more severe problem than indicated by message COMP_19, as it refers to allocation of internal tables whose size cannot be altered by configurable parameters, and which are sufficiently small that there should never be an allocation failure.

Panic CMP_INVALID_NET

Short Syntax: An invalid NET identifier was detected in an internal call.

Description: The NET parameter passed to a function was invalid (probably NULL).

Panic CMP_INVALID_CTX

Short Syntax: An invalid CmpContext identifier was detected in an internal call.

Description: The CmpContext parameter passed to a function was invalid.

Chapter 25. Data Encoding Subystem

This chapter describes Data Encoding Subystem messages. For information on message content and how to use the message, refer to the Introduction.

ES.001

Level: C_INFO

Short Syntax: ES.001 INFO: *infoString*

Long Syntax: ES.001 INFORMATION: *infoString*

Description: General purpose message.

ES.002

Level: C_INFO

Short Syntax: ES.002 Port *port* circuit *circuit* moved to device *device*

Long Syntax: ES.002 Port *port* circuit *circuit* moved to device *device*

Description: A circuit has been reassigned to a different compression/encryption device. This typically occurs when the hardware compression/encryption coprocessor has been disabled.

ES.003

Level: C_INFO

Short Syntax: ES.003 Getting a *objString* from the free pool

Long Syntax: ES.003 Getting a *objString* from the free pool.

Description: An ES client requested a new object from the free pool.

ES.004

Level: C_INFO

Short Syntax: ES.004 Returned a *objString* to the free pool

Long Syntax: ES.004 Returned a *objString* to the free pool.

Description: An ES client returned an object to the free pool.

ES.005

Level: CI_ERROR

Short Syntax: ES.005 Error freeing circuit, pt 0x *port* cir 0x *circuit* num cir *numCircuits* rc *errCode*

Long Syntax: ES.005 Error freeing circuit, port 0x *port* circuit 0x *circuit*, *numCircuits* circuit(s) rc *errCode*

Description: An ES client attempted to returned a circuit to the pool of free circuits, but encountered a problem. The error code is the last value specified: 2 - Invalid parameter 16 - Circuit not found

Chapter 26. Data Encryption (ENCR)

This chapter describes Data Encryption (ENCR) messages. For information on message content and how to use the message, refer to the Introduction.

ENCR.001

Level: P-TRACE

Short Syntax: ENCR.001 ENCR *alg,* , pkt len *pktlen,* -> send len *cmplen,*, net *network ID*

Long Syntax: ENCR.001 ENCR alg *alg,* encrypt: original pkt len *pktlen,*, encrypt pkt len *cmplen,*, on network *network ID*

Description: Per-packet trace encryption results.

ENCR.002

Level: UE-ERROR

Short Syntax: ENCR.002 ENCR *alg,*/encryption err *rc,* doing *doing,*, nt *network ID*

Long Syntax: ENCR.002 ENCR *alg,*/encryption error *rc,* doing *doing,* on network *network ID*

Description: Encrypter returned an error code. The "doing" parameter indicates what the encrypter was working on.

ENCR.003

Level: UE-ERROR

Short Syntax: ENCR.003 ENCR *alg,*/decrypt err *rc,* doing *doing,*, nt *network ID*

Long Syntax: ENCR.003 ENCR *alg,*/decrypt error *rc,* doing *doing,* on network *network ID*

Description: Decrypter returned an error code. The "doing" parameter indicates what the decrypter was working on.

ENCR.004

Level: UE-ERROR

Short Syntax: ENCR.004 ENCR *alg,* err nobuf net *network ID*

Long Syntax: ENCR.004 ENCR *alg,* error, can't get buffer on network *network ID*

Description: Encrypter routine couldn't obtain work buffer.

Chapter 27. Data Link Switching (DLSw)

This chapter describes Data Link Switching (DLSw) messages. For information on message content and how to use the message, refer to the Introduction.

DLS.003

Level: UI-ERROR

Short Syntax: DLS.003 DLSw, Dynamic Neighbors DISABLED, *ip_address* connection rejected

Long Syntax: DLS.003 DLSw, Dynamic Neighbors DISABLED, entry through read port from IP address *ip_address* has been rejected

Description: Our read TCP connection has been opened via an unknown Neighbor and Dynamic Neighbors are DISABLED. As a result, we reject the connection, and thereby close it.

DLS.005

Level: C-INFO

Short Syntax: DLS.005 Opening TCP connection to Neighbor *ip_address* (ports *tcb_sprt* -> *tcb_dprt*)

Long Syntax: DLS.005 Opening a new TCP connection to the Neighbor at IP address *ip_address* (Local Port *tcb_sprt* to Remote Port *tcb_dprt*)

Description: As a result of DLS requesting an OPEN to a particular destination specified by the ip address, TCPIM opens a connection to the destination using the specified ports.

DLS.008

Level: UE-ERROR

Short Syntax: DLS.008 DLSw disabled no SRB seg defined config

Long Syntax: DLS.008 DLSw forwarder disabled no SRB segment defined

Description: The Data Link Switching forwarder has been disabled because of improper configuration. This was no SRB segment number defined, though there was LLC-2 saps defined.

DLS.014

Level: UE-ERROR

Short Syntax: DLS.014 no mem to join group

Long Syntax: DLS.014 no memory to join group

Description: There was not enough free memory allocated to the data structures neccessary to join a group.

DLS.015

Level: UE-ERROR

Short Syntax: DLS.015 no iorb to send group packet

Long Syntax: DLS.015 no iorb to send group packet

Description: There was no iorb buffer available to send a group join or join response.

DLS.016

Level: P-TRACE

Short Syntax: DLS.016 Sent group pkt type *type* group *group* role *role* dest *destination*

Long Syntax: DLS.016 Sent group packet type *type* group *group* role *role* dest *destination*

Description: A DLSw group packet was sent.

DLS.018

Level: P-TRACE

Short Syntax: DLS.018 Rcvd group pkt type *type* group *group* role *role* src *source*

Long Syntax: DLS.018 Received group packet type *type* group *group* role *role* source *source*

Description: A DLSw group packet was received.

DLS.019

Level: UE-ERROR

Short Syntax: DLS.019 Rcvd bad group pkt vers *version* type *type* priority *priority* domain *domain*

Long Syntax: DLS.019 Received bad group packet version *version* type *type* priority *priority* domain *domain*

Description: A DLSw group packet was received with either a bad version #, type, priority, or domain id.

DLS.021

Level: C-INFO

Short Syntax: DLS.021 Rcvd group pkt mismatched roles group *group* role *role*

Long Syntax: DLS.021 Received group packet but mismatched roles group *group* role *role*

Description: A group packet was received but the

roles were mismatched. The only valid role matches are Client/Server and Peer/Peer.

DLS.022

Level: C-INFO

Short Syntax: DLS.022 Contacted by Neighbor *address* from group *group*

Long Syntax: DLS.022 Contacted by a Neighbor at IP Address *address* from Multicast group *group*

Description: A group match has been found and we are opening a connection.

DLS.025

Level: UE-ERROR

Short Syntax: DLS.025 No mem to queue group packet to tasker

Long Syntax: DLS.025 No memory to queue group packet to tasker

Description: There was not enough memory to get a queue header to add a task to send a group packet.

DLS.026

Level: UE-ERROR

Short Syntax: DLS.026 group packet not sent, tasker queue full

Long Syntax: DLS.026 group packet not sent, tasker queue full

Description: A group packet could not be sent becuase the tasker queue was full.

DLS.028

Level: UE-ERROR

Short Syntax: DLS.028 no mem to init SDLC link nt *network ID*

Long Syntax: DLS.028 no memory to initialize SDLC link net *network ID*

Description: There was not enough memory available to initialize an SDLC link.

DLS.029

Level: UI-ERROR

Short Syntax: DLS.029 unexp rtn code from sdlc open station = *rtn_code* nt *network ID*

Long Syntax: DLS.029 unexpected return code from sdlc open station = *rtn_code* net *network ID*

Description: The sdlc open station function returned an unexpected return code.

DLS.030

Level: UI-ERROR

Short Syntax: DLS.030 sdlc lnk ctl blk not fnd during del lnk nt *network ID*

Long Syntax: DLS.030 sdlc link control block not found during delete link net *network ID*

Description: The sdlc link control block was not found for the SDLC link being deleted.

DLS.031

Level: C-INFO

Short Syntax: DLS.031 sdlc station closed nt *network ID*

Long Syntax: DLS.031 sdlc station closed net *network ID*

Description: The sdlc station for the network interface has been successfully closed.

DLS.032

Level: UI-ERROR

Short Syntax: DLS.032 unexp rtn code from sdlc cls station = *rtn_code* nt *network ID*

Long Syntax: DLS.032 unexpected return code from sdlc close station = *rtn_code* net *network ID*

Description: The sdlc close station function returned an unexpected return code.

DLS.033

Level: UI-ERROR

Short Syntax: DLS.033 sdlc lnk ctl blk not fnd during init lnk sta nt *network ID*

Long Syntax: DLS.033 sdlc link control block not found during init link station net *network ID*

Description: The sdlc link control block was not found for the SDLC link station being initialized.

DLS.034

Level: UE-ERROR

Short Syntax: DLS.034 no mem to init SDLC link sta nt *network ID*

Long Syntax: DLS.034 no memory to initialize SDLC link station net *network ID*

Description: There was not enough memory available to initialize an SDLC link station.

DLS.035

Level: C-INFO

Short Syntax: DLS.035 sdlc link sta open addr *link_address* nt *network ID*

Long Syntax: DLS.035 sdlc link station opened address *link_address* net *network ID*

Description: The sdlc link station for the link address has been successfully opened on the network interface.

DLS.036

Level: UI-ERROR

Short Syntax: DLS.036 dupl sdlc link sta addr *link_address* nt *network ID*

Long Syntax: DLS.036 duplicate sdlc link station address *link_address* net *network ID*

Description: The specified sdlc link station could not be opened bceause it is a duplicate of one already opened.

DLS.037

Level: UI-ERROR

Short Syntax: DLS.037 unexp rtn code from sdlc open lnk sta = *rtn_code* nt *network ID*

Long Syntax: DLS.037 unexpected return code from sdlc open link station = *rtn_code* net *network ID*

Description: The sdlc open link station function returned an unexpected return code.

DLS.038

Level: C-INFO

Short Syntax: DLS.038 sdlc link station closed addr *link_address* nt *network ID*

Long Syntax: DLS.038 sdlc station closed address *link_address* net *network ID*

Description: The sdlc link station for the address and network interface specified has been successfully closed.

DLS.039

Level: C-INFO

Short Syntax: DLS.039 processing sdlc net up for addr *link_address* nt *network ID*

Long Syntax: DLS.039 processing sdlc net up for address *link_address* net *network ID*

Description: A net up indication has been received for an sdlc link station.

DLS.040

Level: C-INFO

Short Syntax: DLS.040 processing sdlc net down for addr *link_address* nt *network ID*

Long Syntax: DLS.040 processing sdlc net down for address *link_address* net *network ID*

Description: A net down indication has been received for an sdlc link station.

DLS.041

Level: C-INFO

Short Syntax: DLS.041 rcvd sdlc net up while not in down st for addr *link_address* nt *network ID*

Long Syntax: DLS.041 received sdlc net up while not in down state for address *link_address* net *network ID*

Description: A net up indication has been received for an sdlc link station on an interface that is not down.

DLS.042

Level: C-INFO

Short Syntax: DLS.042 sdlc trans to resolve pend st for addr *link_address* nt *network ID*

Long Syntax: DLS.042 sdlc transition to resolve pending state for address *link_address* net *network ID*

Description: The sdlc link station specified is transitioning to resolve pending state.

DLS.043

Level: UI-ERROR

Short Syntax: DLS.043 unexp sdlc test rsp for addr *link_address* nt *network ID*

Long Syntax: DLS.043 unexpected sdlc test response for address *link_address* net *network ID*

Description: An unexpected test response was received from the sdlc link station specified.

DLS.044

Level: UE-ERROR

Short Syntax: DLS.044 unexp sdlc non xid0 from pu 2 dev for addr *link_address* nt *network ID*

Long Syntax: DLS.044 unexpected sdlc non xid0 from pu 2 device for address *link_address* net *network ID*

Description: An unexpected XID type was received for a PU type 2 device from the sdlc link station specified.

DLS.045

Level: UE-ERROR

Short Syntax: DLS.045 invalid sdlc xid0 len from addr *link_address* nt *network ID*

Long Syntax: DLS.045 invalid sdlc xid0 length from address *link_address* net *network ID*

Description: An invalid length XID-0 was received from the sdlc link station specified.

DLS.046

Level: UE-ERROR

Short Syntax: DLS.046 conn ind rcvd from sec sdlc station from addr *link_address* nt *network ID*

Long Syntax: DLS.046 connection indication received from secondary sdlc station from address *link_address* net *network ID*

Description: An indication that a secondary SDLC link station sent a SNRM was received from the sdlc link station specified.

DLS.047

Level: C-INFO

Short Syntax: DLS.047 sdlc trans to contacted st for addr *link_address* nt *network ID*

Long Syntax: DLS.047 sdlc transition to contacted state for address *link_address* net *network ID*

Description: The sdlc link station specified is transitioning to contacted state.

DLS.048

Level: UE-ERROR

Short Syntax: DLS.048 unexp sdlc conn cfm for addr *link_address* nt *network ID*

Long Syntax: DLS.048 unexpected sdlc connect confirm for address *link_address* net *network ID*

Description: An unexpected connect confirm was received from the sdlc link station specified.

DLS.049

Level: C-INFO

Short Syntax: DLS.049 sdlc disc ind rcvd addr *link_address* nt *network ID*

Long Syntax: DLS.049 sdlc disconnect indication received address *link_address* net *network ID*

Description: A disconnect indication was received from the SDLC link station specified.

DLS.050

Level: C-INFO

Short Syntax: DLS.050 sdlc disc compl addr *link_address* nt *network ID*

Long Syntax: DLS.050 sdlc disconnect complete address *link_address* net *network ID*

Description: The disconnect sequence for the SDLC link station specified is complete.

DLS.051

Level: UE-ERROR

Short Syntax: DLS.051 unexp sdlc disc ind in st *state* for addr *link_address* nt *network ID*

Long Syntax: DLS.051 unexpected sdlc disconnect indication in state *state* for address *link_address* net *network ID*

Description: An unexpected disconnect indication was received from the sdlc link station specified.

DLS.052

Level: C-INFO

Short Syntax: DLS.052 sdlc disc conf addr *link_address* nt *network ID*

Long Syntax: DLS.052 sdlc disconnect confirm address *link_address* net *network ID*

Description: A disconnect confirm was received for the SDLC link station specified.

DLS.053

Level: UE-ERROR

Short Syntax: DLS.053 unexp sdlc disc cfm in state *state* for addr *link_address* nt *network ID*

Long Syntax: DLS.053 unexpected sdlc disconnect confirm in state *state* for address *link_address* net *network ID*

Description: An unexpected disconnect confirm was received from the sdlc link station specified.

DLS.054

Level: UE-ERROR

Short Syntax: DLS.054 unexp sdlc resolve_r in state *state* for addr *link_address* nt *network ID*

Long Syntax: DLS.054 unexpected sdlc resolve_r in state *state* for address *link_address* net *network ID*

Description: An unexpected resolve_r event was received for the sdlc link station specified.

DLS.055

Level: C-INFO

Short Syntax: DLS.055 sdlc trans to connected st for addr *link_address* nt *network ID*

Long Syntax: DLS.055 sdlc transition to connected state for address *link_address* net *network ID*

Description: The sdlc link station specified is transitioning to connected state.

DLS.056

Level: UI-ERROR

Short Syntax: DLS.056 unexp rtn code from sdlc conn req = *rtn_code* for addr *link_address* nt *network ID*

Long Syntax: DLS.056 unexpected return code from sdlc connect request = *rtn_code* for address *link_address* net *network ID*

Description: The sdlc connect request function returned an unexpected return code.

DLS.057

Level: UE-ERROR

Short Syntax: DLS.057 unexp sdlc xid from dls in state *state* for addr *link_address* nt *network ID*

Long Syntax: DLS.057 unexpected sdlc xid from dls in state *state* for address *link_address* net *network ID*

Description: An unexpected xid event was received for the sdlc link station specified.

DLS.058

Level: UE-ERROR

Short Syntax: DLS.058 unexp sdlc xid3 from dls for pu 2 dev for addr *link_address* nt *network ID*

Long Syntax: DLS.058 unexpected sdlc xid3 from dls for pu 2 device for address *link_address* net *network ID*

Description: An unexpected XID-3 was received from DLS for a PU type 2 device for the sdlc link station specified.

DLS.059

Level: UE-ERROR

Short Syntax: DLS.059 unexp sdlc dlc_contact from dls in state *state* for addr *link_address* nt *network ID*

Long Syntax: DLS.059 unexpected sdlc dlc_contact from dls in state *state* for address *link_address* net *network ID*

Description: An unexpected DLC_CONTACT event was received from DLS for the sdlc link station specified.

DLS.060

Level: UE-ERROR

Short Syntax: DLS.060 unexp sdlc dlc_info from dls in state *state* for addr *link_address* nt *network ID*

Long Syntax: DLS.060 unexpected sdlc dlc_info from dls in state *state* for address *link_address* net *network ID*

Description: An unexpected DLC_INFO event was received from DLS for the sdlc link station specified.

DLS.061

Level: UE-ERROR

Short Syntax: DLS.061 unexp sdlc dlc_dgrm from dls in state *state* for addr *link_address* nt *network ID*

Long Syntax: DLS.061 unexpected sdlc dlc_dgrm from dls in state *state* for address *link_address* net *network ID*

Description: An unexpected DLC_DGRM event was received from DLS for the sdlc link station specified.

DLS.062

Level: UE-ERROR

Short Syntax: DLS.062 unexp I-frame from sdlc in state *state* for addr *link_address* nt *network ID*

Long Syntax: DLS.062 unexpected I-frame from sdlc in state *state* for address *link_address* net *network ID*

Description: An unexpected I-frame was received from SDLC for the sdlc link station specified.

DLS.063

Level: UE-ERROR

Short Syntax: DLS.063 unexp UI-frame from sdlc in state *state* for addr *link_address* nt *network ID*

Long Syntax: DLS.063 unexpected UI-frame from sdlc in state *state* for address *link_address* net *network ID*

Description: An unexpected UI-frame was received from SDLC for the sdlc link station specified.

DLS.064

Level: C-INFO

Short Syntax: DLS.064 rcvd halt_dl from dls for sdlc addr *link_address* nt *network ID*

Long Syntax: DLS.064 received halt_dl for sdlc address *link_address* net *network ID*

Description: A HALT_DL event was sent from DLS for the specified sdlc link station

DLS.065

Level: C-INFO

Short Syntax: DLS.065 sdlc trans to disc pend st for addr *link_address* nt *network ID*

Long Syntax: DLS.065 sdlc transition to disconnect pending state for address *link_address* net *network ID*

Description: The sdlc link station specified is transitioning to disconnect pending state.

DLS.066

Level: UI-ERROR

Short Syntax: DLS.066 unexp rtn code from sdlc disc req = *rtn_code* addr *link_address* nt *network ID*

Long Syntax: DLS.066 unexpected return code from sdlc disconnect request = *rtn_code* addr *link_address* net *network ID*

Description: The sdlc disconnect request function returned an unexpected return code.

DLS.067

Level: C-INFO

Short Syntax: DLS.067 sdlc trans to disc st for addr *link_address* nt *network ID*

Long Syntax: DLS.067 sdlc transition to disconnect state for address *link_address* net *network ID*

Description: The sdlc link station specified is transitioning to disconnect state.

DLS.068

Level: UE-ERROR

Short Syntax: DLS.068 unexp sdlc dlc_halt_dl from dls in state *state* for addr *link_address* nt *network ID*

Long Syntax: DLS.068 unexpected sdlc dlc_halt_dl from dls in state *state* for address *link_address* net *network ID*

Description: An unexpected DLC_HALT_DL event was received from DLS for the sdlc link station specified.

DLS.069

Level: C-INFO

Short Syntax: DLS.069 cleanup timer expired for addr *link_address* nt *network ID*

Long Syntax: DLS.069 cleanup timer expired for address *link_address* net *network ID*

Description: The cleanup timer expired for the specified sdlc link station.

DLS.070

Level: UE-ERROR

Short Syntax: DLS.070 unexp sdlc cleanup timer exp in state *state* for addr *link_address* nt *network ID*

Long Syntax: DLS.070 unexpected sdlc cleanup timer expiration in state *state* for address *link_address* net *network ID*

Description: The sdlc cleanup timer expired, but the sdlc link station is in an unexpected state.

DLS.071

Level: C-INFO

Short Syntax: DLS.071 sdlc buf retry timer expired for addr *link_address* nt *network ID*

Long Syntax: DLS.071 sdlc buffer retry timer expired for address *link_address* net *network ID*

Description: The buffer retry timer expired for the specified sdlc link station.

DLS.072

Level: UE-ERROR

Short Syntax: DLS.072 unexp sdlc buf retry timer exp in state *state* for addr *link_address* nt *network ID*

Long Syntax: DLS.072 unexpected sdlc buffer retry timer expiration in state *state* for address *link_address* net *network ID*

Description: The sdlc buffer retry timer expired, but the sdlc link station is in an unexpected state.

DLS.073

Level: UE-ERROR

Short Syntax: DLS.073 unknown sdlc fsm input = *event* for addr *link_address* nt *network ID*

Long Syntax: DLS.073 unknown sdlc fsm input = *event* for address *link_address* net *network ID*

Description: The sdlc interface finite state machine was passed an unknown event.

DLS.077

Level: UE-ERROR

Short Syntax: DLS.077 no buf for sdlc test for addr *link_address* nt *network ID*

Long Syntax: DLS.077 no buffer for sdlc test for address *link_address* net *network ID*

Description: No buffer could be obtained for sending a test frame to an SDLC link station. The operation will be retried later.

DLS.078

Level: UI-ERROR

Short Syntax: DLS.078 unexp rtn code from sdlc test req = *rtn_code* addr *link_address* nt *network ID*

Long Syntax: DLS.078 unexpected return code from sdlc test request = *rtn_code* addr *link_address* net *network ID*

Description: The sdlc test request function returned an unexpected return code.

DLS.079

Level: UE-ERROR

Short Syntax: DLS.079 no buf for sdlc xid0 for addr *link_address* nt *network ID*

Long Syntax: DLS.079 no buffer for sdlc xid0 for address *link_address* net *network ID*

Description: No buffer could be obtained for sending an xid0 frame to an SDLC link station. The operation will be retried later.

DLS.080

Level: UE-ERROR

Short Syntax: DLS.080 no buf for sdlc null xid for addr *link_address* nt *network ID*

Long Syntax: DLS.080 no buffer for sdlc null xid for address *link_address* net *network ID*

Description: No buffer could be obtained for sending a null xid frame to an SDLC link station. The operation will be retried later.

DLS.081

Level: UI-ERROR

Short Syntax: DLS.081 unexp rtn code from sdlc I frm req = *rtn_code* addr *link_address* nt *network ID*

Long Syntax: DLS.081 unexpected return code from sdlc I frame request = *rtn_code* address *link_address* net *network ID*

Description: The sdlc I frame request function returned an unexpected return code.

DLS.082

Level: UI-ERROR

Short Syntax: DLS.082 unexp rtn code from sdlc UI frm req = *rtn_code* addr *link_address* nt *network ID*

Long Syntax: DLS.082 unexpected return code from sdlc UI frame request = *rtn_code* addr *link_address* net *network ID*

Description: The sdlc UI frame request function returned an unexpected return code.

DLS.083

Level: UI-ERROR

Short Syntax: DLS.083 unexp rtn code from sdlc force rnr req = *rtn_code* addr *link_address* nt *network ID*

Long Syntax: DLS.083 unexpected return code from sdlc force rnr request = *rtn_code* addr *link_address* net *network ID*

Description: The sdlc force rnr request function returned an unexpected return code.

DLS.086

Level: UE-ERROR

Short Syntax: DLS.086 sdlc disc rcvd rsn *reason* for addr *link_address* nt *network ID*

Long Syntax: DLS.086 sdlc disconnect received reason *reason* for address *link_address* net *network ID*

Description: The specified SDLC connection was disconnected due to an error detected by the SDLC protocol.

DLS.087

Level: C-INFO

Short Syntax: DLS.087 sdlc trans to null_xid_pend st for addr *link_address* nt *network ID*

Long Syntax: DLS.087 sdlc transition to null_xid_pend state for address *link_address* net *network ID*

Description: The sdlc link station specified is transitioning to NULL_XID_PENDING state, meaning that it is awaiting a response to a NULL XID that was sent.

DLS.088

Level: C-INFO

Short Syntax: DLS.088 sdlc trans to xid_0_pend st for addr *link_address* nt *network ID*

Long Syntax: DLS.088 sdlc transition to xid_0_pend state for address *link_address* net *network ID*

Description: The sdlc link station specified is transitioning to XID_0_PENDING state, meaning that it is awaiting a response to an XID-0 that was sent.

DLS.089

Level: UE-ERROR

Short Syntax: DLS.089 DLS, TCP conn brk to *address*, DLS sess closed *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.089 DLS forwarder experience a session loss due to TCP connection to *address* break, origin MAC *source_mac_address*->Target MAC *dest_mac_address*, origin SAP *source_sap*->Target SAP *dest_sap*

Description: TCP connection to the DLS neighbor went down. All the DLS sessions active on that TCP connection are brought down individually.

DLS.096

Level: U-INFO

Short Syntax: DLS.096 DLS, DL_STARTED event in cir-est or cir-restart state, ignore

Long Syntax: DLS.096 DLS DLC_DL_STARTED event from underlying DLCST in circuit established or circuit restart state

Description: DLS state machine received DLC_DL_STARTED event indication from the underlying LINK (LLC or SDLC) when the circuit has already been established. This could be the dribbling TEST responses coming in via bridge multi-path. Ignore them.

DLS.097

Level: UI-ERROR

Short Syntax: DLS.097 DLS, No memory available to create DLS session.

Long Syntax: DLS.097 DLS, No memory available to create DLS session.

Description: There is no memory available to allocate the resources that are required to create a DLS session.

DLS.102

Level: C-INFO

Short Syntax: DLS.102 DLS, Broadcast CANUREACH_ex sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* lfsize *largest_frame_size*

Long Syntax: DLS.102 DLS, Broadcast CANUREACH_ex sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* lfsize *largest_frame_size*

Description: While processing TEST(c) for a given destination, DLS sent out broadcast CANUREACH_ex via multicast UDP.

DLS.104

Level: C-INFO

Short Syntax: DLS.104 DLS, SAPs resolved for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.104 DLS, SAPs resolved for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS connection can get established without using the specific SAPs, for example, SAP 0 as DSAP or SSAP. However, when the specific SAPs are used, the same connection's SAPs are updated.

DLS.106

Level: UI-ERROR

Short Syntax: DLS.106 DLS, rcvd CANUREACH not proc by any DLCs for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.106 DLS, received CANUREACH could not be processed by any DLC for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: A CANUREACH SSP message we received over the TCP from a DLS Peer. However, none of the underlying data link layer, SDLC or LLC, could translate this to a TEST(c) frame.

DLS.107

Level: C-INFO

Short Syntax: DLS.107 CANUREACH-ex rcvd *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* lfsize *largest_frame_size*

Long Syntax: DLS.107 CANUREACH-ex received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* lfsize *largest_frame_size*

Description: DLS has received a CANUREACH-ex for the specified circuit.

DLS.113

Level: UI-ERROR

Short Syntax: DLS.113 LLC, Initialization FAILED for SAP *Sap*

Long Syntax: DLS.113 LLC, Initialization FAILED for SAP *Sap*

Description: Due to some problems, SAP initialization with LLC failed.

DLS.114

Level: UI-ERROR

Short Syntax: DLS.114 LLC, parameter validation FAILED for SAP *Sap*, rsn = *reason*

Long Syntax: DLS.114 LLC, parameter validation FAILED for SAP *Sap*, rsn = *reason*

Description: LLC tunable parameters are out of range.

DLS.115

Level: UI-ERROR

Short Syntax: DLS.115 *intfmod*, No Memory for SAP control block for SAP *Sap*

Long Syntax: DLS.115 *intfmod*, No Memory for SAP control block for SAP *Sap*

Description: No memory available for the SAP control block.

DLS.116

Level: UI-ERROR

Short Syntax: DLS.116 *intfmod*, Opening of SAP *Sap* FAILED, rsn = *reason*

Long Syntax: DLS.116 *intfmod*, Opening of SAP *Sap* FAILED, rsn = *reason*

Description: Opening of SAP failed due to problems in the LLC or local APPN. Reason code is indicative of the specific problem.

DLS.117

Level: C-INFO

Short Syntax: DLS.117 LLC, Closing SAP *Sap*

Long Syntax: DLS.117 LLC, Closing SAP *Sap*

Description: Closing SAP with the LLC.

DLS.118

Level: UI-ERROR

Short Syntax: DLS.118 *intfmod*, FAILED open stn, invalid sapcb, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Long Syntax: DLS.118 *intfmod*, FAILED open stn, invalid sapcb, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Description: Opening of station for LLC or APPN data link services failed because the SAP under which station to be opened is invalid.

DLS.119

Level: UI-ERROR

Short Syntax: DLS.119 *intfmod*, FAILED open stn, No memory, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Long Syntax: DLS.119 *intfmod*, FAILED open stn, No memory, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Description: Opening of station for LLC or APPN data link services failed because there is no memory available to create control block to manage the connection.

DLS.120

Level: UI-ERROR

Short Syntax: DLS.120 LLC, FAILED open stn, rsn= *Reason*, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Long Syntax: DLS.120 LLC, FAILED open stn, rsn= *Reason*, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Description: Opening of station for LLC data link services failed due to some problems within LLC. The reason code is indicative of the specific problem.

DLS.121

Level: C-INFO

Short Syntax: DLS.121 *intfmod*, opened stn, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Long Syntax: DLS.121 *intfmod*, opened stn, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Description: Opening of station for LLC data link services succeeded.

DLS.124

Level: C-INFO

Short Syntax: DLS.124 LLC, closed stn by force *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.124 LLC, closed stn by force *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Closed station by force for LLC data link services.

DLS.125

Level: C-INFO

Short Syntax: DLS.125 *intfmod*, closed stn quietly *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.125 *intfmod*, closed stn quietly *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Closed station quietly for LLC or local APPN data link services.

DLS.126

Level: UI-ERROR

Short Syntax: DLS.126 LLC, *action* Send failed, rsn= *reason*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.126 LLC, *action* Send failed rsn= *reason*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC was unsuccessful in sending out a frame. The reason indicative of the specific problem.

DLS.127

Level: U-INFO

Short Syntax: DLS.127 *intfmod*, *action* became busy after sending, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.127 *intfmod*, *action* became busy after sending, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN became busy after sending out a frame. This busyness is an honor system and the frame submitted to LLC does get accepted for sending. However, DLS takes note of this condition and refrains from sending more frames.

DLS.128

Level: U-INFO

Short Syntax: DLS.128 *intfmod* BUSY, enq frm to tx pendQ, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.128 *intfmod* BUSY, enqueue frame to pend queue, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN is busy, frames received from TCP are being enqueued to the pending queue for that LLC or APPN session. When LLC or APPN exits busy condition, all the frames from the pending queue will be flushed.

DLS.130

Level: UI-ERROR

Short Syntax: DLS.130 *intfmod*, frame refused, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.130 *intfmod*, frame not proc, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Frame was not processed by the DLS as the SAP, as well as the station was not opened for LLC or local APPN data link services.

DLS.131

Level: C-INFO

Short Syntax: DLS.131 LLC, frame refused, NOT switching for network *network*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.131 LLC, frame refused, NOT switching for network *network*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Frame was not processed by the DLS as the DSAP of the frame is not configured to be switched for the interface on which the frame was received.

DLS.134

Level: UI-ERROR

Short Syntax: DLS.134 LLC, *llcevent* event not proc, handle is bad

Long Syntax: DLS.134 LLC, *llcevent* event not proc, handle is bad

Description: An LLC event was not processed by the DLS as the handle by the LLC to DLS was bad.

DLS.135

Level: UI-ERROR

Short Syntax: DLS.135 LLC, *llcevent* unknown event, not proc for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.135 LLC, *llcevent* unknown event, not proc for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: An unrecognizable LLC event occurred. This event is not processed.

DLS.136

Level: UI-ERROR

Short Syntax: DLS.136 *intfmod*, *llcevent* req not proc, inv handle, for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.136 *intfmod*, *llcevent* req not proc, inv handle, for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: An LLC or APPN request from DLS was not processed by the LLC or APPN interface module as the handle passed from DLS is bad.

DLS.137

Level: U-INFO

Short Syntax: DLS.137 LLC, not enabled, start_dl not honored, for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.137 LLC, not enabled, start_dl not honored, for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: A Start DL request from DLS to LLC interface module was not honored as LLC portion of the DLS is not enabled. This is not an error. When a CANUREACH is received, DLS will issue START_DL request to data link service interface modules. If they are not configured to switch for the SAP or not configured at all, it is common to discard such request.

DLS.138

Level: UI-ERROR

Short Syntax: DLS.138 *intfmod*, START_DL discard as err open stn, for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.138 *intfmod*, START_DL discard as error happened during open station operation, for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: A Start DL request from DLS to LLC or APPN interface module was not honored as data link station could not be opened with LLC or APPN.

DLS.142

Level: UI-ERROR

Short Syntax: DLS.142 *intfmod*, event *eventname* received in bad state *statename*, *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.142 *intfmod* event *eventname* received in bad state *statename*, *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS interface to LLC or APPN received an event in the bad state.

DLS.144

Level: C-INFO

Short Syntax: DLS.144 LLC, Secondary TEST_R ign, for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.144 LLC Secondary TEST_R is ignored, for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: In Multipath bridge environment, it is possible to receive multiple responses to the TEST command sent earlier. The LLC interface module discards such TEST responses.

DLS.154

Level: C-INFO

Short Syntax: DLS.154 *intfmod*, *frame_type* frame drpped, *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap* prt nt up

Long Syntax: DLS.154 *intfmod*, *frame_type* frame drpped, *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap* prt nt up

Description: A frame received from DLS cloud was not successfully sent - bridge port not up or the local APPN was not available.

DLS.156

Level: C-INFO

Short Syntax: DLS.156 DLS session pool of *count* bytes created for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.156 DLS session pool of *count* bytes created for origin MAC *source_mac_address->*Target MAC *dest_mac_address*, origin SAP *source_sap->*Target SAP *dest_sap*

Description: A session pool has been created for a new DLS connection. This pool is used exclusively by this session for data transfer.

DLS.157

Level: C-INFO

Short Syntax: DLS.157 Global DLS *type* pool of *count* bytes created

Long Syntax: DLS.157 Global DLS *type* pool of *count* bytes created

Description: The global DLS memory pool has been created. This is used for SSP control messages and other non-data transfer related items.

DLS.158

Level: UE-ERROR

Short Syntax: DLS.158 Cannot create DLS session pool of *count* bytes for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.158 Cannot create DLS session pool of *count* bytes for origin MAC *source_mac_address->*Target MAC *dest_mac_address*, origin SAP *source_sap->*Target SAP *dest_sap*

Description: There is currently not enough memory available to support a new DLS session.

DLS.159

Level: UE-ERROR

Short Syntax: DLS.159 Cannot alloc global DLS *type* pool of *count* bytes

Long Syntax: DLS.159 Cannot alloc global DLS *type* pool of *count* bytes

Description: There is not enough memory available to support DLS. DLS has been disabled.

DLS.160

Level: C-INFO

Short Syntax: DLS.160 Entering *flow_ctrl_type* congestion for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.160 Entering *flow_ctrl_type* congestion for origin MAC *source_mac_address*->Target MAC *dest_mac_address*, origin SAP *source_sap*->Target SAP *dest_sap*

Description: The DLS session is congested due to either TCP backup, or the receipt of a DLS_ENTER_BUSY SSP message. If this happens too frequently, consider increasing the amount of memory allocated to each DLS session.

DLS.161

Level: C-INFO

Short Syntax: DLS.161 Entering GLOBAL congestion on global DLS *pool_type* pool state= *pool_state* mem= *memavail*

Long Syntax: DLS.161 Entering GLOBAL congestion on global DLS *pool_type* pool state= *pool_state* mem= *memavail*

Description: The total amount of memory allocated by all currently active DLS sessions exceeded the amount pre-allocated by the user for DLS. As a result, the data links are temporarily being quiesced until some of the memory is freed up. If this happens too frequently, consider increasing the amount of memory allocated to DLSw.

DLS.162

Level: C-INFO

Short Syntax: DLS.162 Exiting *flow_ctrl_type* congestion for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.162 Exiting *flow_ctrl_type* congestion for origin MAC *source_mac_address*->Target MAC *dest_mac_address*, origin SAP *source_sap*->Target SAP *dest_sap*

Description: Sufficient memory has been freed up since the last time DLS was congested to allow the data links to receive data again.

DLS.163

Level: C-INFO

Short Syntax: DLS.163 Exiting GLOBAL congestion on global DLS *pool_type* pool state= *pool_state* mem= *memavail*

Long Syntax: DLS.163 Exiting GLOBAL congestion on global DLS *pool_type* pool state= *pool_state* mem= *memavail*

Description: Sufficient memory has been freed up since the last time DLS was congested to allow the data links to receive data again.

DLS.164

Level: U-INFO

Short Syntax: DLS.164 no slow buf for copy while queueing data to *ip_address* mode *mode*

Long Syntax: DLS.164 no slow buffer for copy while queueing data to neighbor *ip_address* mode *mode*

Description: No buffer could be obtained for copying a data buffer for queueing while the transport/circuit is congested. The existing device buffer is queued. Mode: ″0″ is Normal, ″1″ is Busy, ″2″ is Urgent, and ″3″ is Pacing.

DLS.165

Level: C-INFO

Short Syntax: DLS.165 DLS session pool deleted for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.165 DLS session pool deleted for origin MAC *source_mac_address*->Target MAC *dest_mac_address*, origin SAP *source_sap*->Target SAP *dest_sap*

Description: All buffers have been returned to a DLSw session pool after the DLSw session has been closed. The pool may now be removed.

DLS.166

Level: C-INFO

Short Syntax: DLS.166 DLS, SSP msg CANUREACH received from *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.166 DLS forwarder received a SSP CANUREACH message over TCP connection to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a Switch to

Switch Protocol message of CANUREACH over TCP.

DLS.167

Level: C-INFO

Short Syntax: DLS.167 DLS, SSP msg ICANREACH received from *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.167 DLS forwarder received a SSP ICANREACH message over TCP connection to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of ICANREACH over TCP.

DLS.168

Level: C-INFO

Short Syntax: DLS.168 DLS, SSP msg REACHACK received from *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.168 DLS forwarder received a SSP REACHACK message over TCP connection to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of REACHACK over TCP.

DLS.169

Level: C-INFO

Short Syntax: DLS.169 DLS, SSP msg XIDFRAME received from *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.169 DLS forwarder received a SSP XIDFRAME message over TCP connection to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of XIDFRAME over TCP.

DLS.170

Level: C-INFO

Short Syntax: DLS.170 DLS, SSP msg DGRMFRAME received from *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.170 DLS forwarder received a SSP DGRMFRAME message over TCP connection to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of DGRMFRAME over TCP.

DLS.171

Level: C-INFO

Short Syntax: DLS.171 DLS, SSP msg CONTACT received from *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.171 DLS forwarder received a SSP CONTACT message over TCP connection to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of CONTACT over TCP.

DLS.172

Level: C-INFO

Short Syntax: DLS.172 DLS, SSP msg CONTACTED received from *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.172 DLS forwarder received a SSP CONTACTED message over TCP connection to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of CONTACTED over TCP.

DLS.173

Level: C-INFO

Short Syntax: DLS.173 DLS, SSP msg DATAFRAME received from *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.173 DLS forwarder received a SSP DATAFRAME message over TCP connection to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of DATAFRAME over TCP.

DLS.174

Level: C-INFO

Short Syntax: DLS.174 DLS, SSP msg RESTART_DL received from *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.174 DLS forwarder received a SSP RESTART_DL message over TCP connection to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of RESTART_DL over TCP.

DLS.175

Level: C-INFO

Short Syntax: DLS.175 DLS, SSP msg RESTARTED received from *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.175 DLS forwarder received a SSP RESTARTED message over TCP connection to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of RESTARTED over TCP.

DLS.176

Level: C-INFO

Short Syntax: DLS.176 DLS, SSP msg HALT_DL received from *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.176 DLS forwarder received a SSP HALT_DL message over TCP connection to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of HALT_DL over TCP.

DLS.177

Level: C-INFO

Short Syntax: DLS.177 DLS, SSP msg DL_HALTED received from *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.177 DLS forwarder received a SSP DL_HALTED message over TCP connection to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of DL_HALTED over TCP.

DLS.178

Level: C-INFO

Short Syntax: DLS.178 DLS, SSP msg INFOFRAME received from *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.178 DLS forwarder received a SSP INFOFRAME message over TCP connection to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of INFOFRAME over TCP.

DLS.179

Level: C-INFO

Short Syntax: DLS.179 DLS, SSP msg ENTER_BUSY received from *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.179 DLS forwarder received a SSP ENTER_BUSY message over TCP connection to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of ENTER_BUSY over TCP.

DLS.180

Level: C-INFO

Short Syntax: DLS.180 DLS, SSP msg EXIT_BUSY received from *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.180 DLS forwarder received a SSP EXIT_BUSY message over TCP connection to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of EXIT_BUSY over TCP.

DLS.181

Level: C-INFO

Short Syntax: DLS.181 DLS, SSP msg HALT_DL_NOACK received from *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.181 DLS forwarder received a SSP HALT_DL_NOACK message over TCP connection to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of HALT_DL_NOACK over TCP.

DLS.182

Level: C-INFO

Short Syntax: DLS.182 DLS, SSP msg IAMOKAY received from *ip_address*

Long Syntax: DLS.182 DLS forwarder received a SSP IAMOKAY message over TCP connection to *ip_address*

Description: DLS forwarder received a Switch to Switch Protocol message of IAMOKAY over TCP.

DLS.183

Level: C-INFO

Short Syntax: DLS.183 DLS, UNRECOGNIZED_SSP received from *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.183 DLS forwarder received an UNRECOGNIZED_SSP message over TCP connection to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received an unrecognized Switch to Switch Protocol message over TCP.

DLS.184

Level: C-INFO

Short Syntax: DLS.184 DLS, DLC event DLC_CONTACTED received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.184 DLS forwarder received a DLC event of type DLC_CONTACTED for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a DLC_CONTACTED event from the underlying data link which could be LLC or SDLC.

DLS.185

Level: C-INFO

Short Syntax: DLS.185 DLS, DLC event DLC_ERROR received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.185 DLS forwarder received a DLC event of type DLC_ERROR for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a DLC_ERROR event from the underlying data link which could be LLC or SDLC.

DLS.186

Level: C-INFO

Short Syntax: DLS.186 DLS, DLC event DLC_RESET received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.186 DLS forwarder received a DLC event of type DLC_RESET for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a DLC_RESET event from the underlying data link which could be LLC or SDLC.

DLS.187

Level: C-INFO

Short Syntax: DLS.187 DLS, DLC event DLC_DL_HALTED received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.187 DLS forwarder received a DLC event of type DLC_DL_HALTED for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a DLC_DL_HALTED event from the underlying data link which could be LLC or SDLC.

DLS.188

Level: C-INFO

Short Syntax: DLS.188 DLS, DLC event DLC_DL_ENTER_BUSY received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.188 DLS forwarder received a DLC event of type DLC_DL_ENTER_BUSY for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a DLC_DL_ENTER_BUSY event from the underlying data link which could be LLC or SDLC.

DLS.189

Level: C-INFO

Short Syntax: DLS.189 DLS, DLC event DLC_DL_EXIT_BUSY received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.189 DLS forwarder received a DLC event of type DLC_DL_EXIT_BUSY for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a DLC_DL_EXIT_BUSY event from the underlying data link which could be LLC or SDLC.

DLS.190

Level: C-INFO

Short Syntax: DLS.190 DLS, DLC event DLC_DL_STARTED received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.190 DLS forwarder received a DLC event of type DLC_DL_STARTED for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a

DLC_DL_STARTED event from the underlying data link which could be LLC or SDLC.

DLS.192

Level: C-INFO

Short Syntax: DLS.192 DLS, DLC event DLC_INFO received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.192 DLS forwarder received a DLC event of type DLC_INFO for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a DLC_INFO event from the underlying data link which could be LLC or SDLC.

DLS.193

Level: C-INFO

Short Syntax: DLS.193 DLS, DLC event DLC_DGRM received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.193 DLS forwarder received a DLC event of type DLC_DGRM for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a DLC_DGRM event from the underlying data link which could be LLC or SDLC.

DLS.194

Level: C-INFO

Short Syntax: DLS.194 DLS, DLC event DLC_XID received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.194 DLS forwarder received a DLC event of type DLC_XID for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a DLC_XID event from the underlying data link which could be LLC or SDLC.

DLS.195

Level: C-INFO

Short Syntax: DLS.195 DLS, DLC event DLC_DATAFRAME received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.195 DLS forwarder received a DLC event of type DLC_DATAFRAME for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a DLC_DATAFRAME event from the underlying data link which could be LLC or SDLC.

DLS.196

Level: C-INFO

Short Syntax: DLS.196 DLS, Transition to DISCONNECTED state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.196 DLS forwarder is transitioning to DISCONNECTED state for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing an event, either an SSP message received over TCP or an event presented from underlying DLC, the associated DLS session is transitioning to DISCONNECTED state.

DLS.197

Level: C-INFO

Short Syntax: DLS.197 DLS, Transition to CONNECT_PENDING state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.197 DLS forwarder is transitioning to CONNECT_PENDING state for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing an event, either an SSP message received over TCP or an event presented from underlying DLC, the associated DLS session is transitioning to CONNECT_PENDING state.

DLS.198

Level: C-INFO

Short Syntax: DLS.198 DLS, Transition to CONNECTED state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.198 DLS forwarder is transitioning to CONNECTED state for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing an event, either an SSP message received over TCP or an event presented from underlying DLC, the associated DLS session is transitioning to CONNECTED state.

DLS.199

Level: C-INFO

Short Syntax: DLS.199 DLS, Transition to DISCONNECT_PENDING state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.199 DLS forwarder is transitioning

to DISCONNECT_PENDING state for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing an event, either an SSP message received over TCP or an event presented from underlying DLC, the associated DLS session is transitioning to DISCONNECT_PENDING state.

DLS.200

Level: C-INFO

Short Syntax: DLS.200 DLS, Transition to CIRCUIT_ESTABLISHED state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.200 DLS forwarder is transitioning to CIRCUIT_ESTABLISHED state for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing an event, either an SSP message received over TCP or an event presented from underlying DLC, the associated DLS session is transitioning to CIRCUIT_ESTABLISHED state.

DLS.201

Level: C-INFO

Short Syntax: DLS.201 DLS, Transition to CIRCUIT_PENDING state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.201 DLS forwarder is transitioning to CIRCUIT_PENDING state for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing an event, either an SSP message received over TCP or an event presented from underlying DLC, the associated DLS session is transitioning to CIRCUIT_PENDING state.

DLS.202

Level: C-INFO

Short Syntax: DLS.202 DLS, Transition to CIRCUIT_RESTART state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.202 DLS forwarder is transitioning to CIRCUIT_RESTART state for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing an event, either an SSP message received over TCP or an event presented from underlying DLC, the associated DLS session is transitioning to CIRCUIT_RESTART state.

DLS.203

Level: C-INFO

Short Syntax: DLS.203 DLS, Transition to RESOLVE_PENDING state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.203 DLS forwarder is transitioning to RESOLVE_PENDING state for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing an event, either an SSP message received over TCP or an event presented from underlying DLC, the associated DLS session is transitioning to RESOLVE_PENDING state.

DLS.204

Level: C-INFO

Short Syntax: DLS.204 DLS, Transition to CONTACT_PENDING state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.204 DLS forwarder is transitioning to CONTACT_PENDING state for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing an event, either an SSP message received over TCP or an event presented from underlying DLC, the associated DLS session is transitioning to CONTACT_PENDING state.

DLS.205

Level: C-INFO

Short Syntax: DLS.205 DLS, Transition to RESTART_PENDING state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.205 DLS forwarder is transitioning to RESTART_PENDING state for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing an event, either an SSP message received over TCP or an event presented from underlying DLC, the associated DLS session is transitioning to RESTART_PENDING state.

DLS.206

Level: C-INFO

Short Syntax: DLS.206 DLS, Transition to HALT_PENDING state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.206 DLS forwarder is transitioning to HALT_PENDING state for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing an event, either an SSP message received over TCP or an event presented from underlying DLC, the associated DLS session is transitioning to HALT_PENDING state.

DLS.207

Level: UE-ERROR

Short Syntax: DLS.207 DLS, DLC_CONTACTED rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.207 DLS forwarder received a DLC_CONTACTED event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a DLC_CONTACTED event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.208

Level: UE-ERROR

Short Syntax: DLS.208 DLS, DLC_DGRM rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.208 DLS forwarder received a DLC_DGRM event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a DLC_DGRM event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.209

Level: UE-ERROR

Short Syntax: DLS.209 DLS, DLC_ERROR rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.209 DLS forwarder received a DLC_ERROR event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a DLC_ERROR event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.210

Level: UE-ERROR

Short Syntax: DLS.210 DLS, DLC_INFO rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.210 DLS forwarder received a DLC_INFO event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a DLC_INFO event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.211

Level: UE-ERROR

Short Syntax: DLS.211 DLS, DLC_DL_HALTED rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.211 DLS forwarder received a DLC_DL_HALTED event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a DLC_DL_HALTED event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.212

Level: UE-ERROR

Short Syntax: DLS.212 DLS, DLC_DL_STARTED rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.212 DLS forwarder received a DLC_DL_STARTED event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a DLC_DL_STARTED event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.213

Level: UE-ERROR

Short Syntax: DLS.213 DLS, DLC_RESET rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.213 DLS forwarder received a DLC_RESET event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a DLC_RESET event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.214

Level: UE-ERROR

Short Syntax: DLS.214 DLS, DLC_RESOLVE_C rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.214 DLS forwarder received a DLC_RESOLVE_C event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a DLC_RESOLVE_C event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.215

Level: UE-ERROR

Short Syntax: DLS.215 DLS, DLC_XID rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.215 DLS forwarder received a DLC_XID event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a DLC_XID event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.216

Level: UE-ERROR

Short Syntax: DLS.216 DLS, CANUREACH rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.216 DLS forwarder received a CANUREACH event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a CANUREACH event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.217

Level: UE-ERROR

Short Syntax: DLS.217 DLS, DGRMFRAME rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.217 DLS forwarder received a DGRMFRAME event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a DGRMFRAME event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.218

Level: UE-ERROR

Short Syntax: DLS.218 DLS, XIDFRAME rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.218 DLS forwarder received a XIDFRAME event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a XIDFRAME event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.220

Level: UE-ERROR

Short Syntax: DLS.220 DLS, CONTACT rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.220 DLS forwarder received a CONTACT event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a CONTACT event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.221

Level: UE-ERROR

Short Syntax: DLS.221 DLS, CONTACTED rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.221 DLS forwarder received a CONTACTED event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a CONTACTED event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.222

Level: UE-ERROR

Short Syntax: DLS.222 DLS, RESTART_DL rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.222 DLS forwarder received a RESTART_DL event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a RESTART_DL event,

the DLS state machine discovered the event occurring in an unexpected state.

DLS.223

Level: UE-ERROR

Short Syntax: DLS.223 DLS, DL_RESTARTED rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.223 DLS forwarder received a DL_RESTARTED event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a DL_RESTARTED event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.224

Level: UE-ERROR

Short Syntax: DLS.224 DLS, INFOFRAME rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.224 DLS forwarder received a INFOFRAME event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing an INFOFRAME event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.225

Level: UE-ERROR

Short Syntax: DLS.225 DLS, HALT_DL rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.225 DLS forwarder received a HALT_DL event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a HALT_DL event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.226

Level: UE-ERROR

Short Syntax: DLS.226 DLS, HALT_DL_NOACK rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.226 DLS forwarder received a HALT_DL_NOACK event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a HALT_DL_NOACK event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.227

Level: UE-ERROR

Short Syntax: DLS.227 DLS, DL_HALTED rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.227 DLS forwarder received a DL_HALTED event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing a DL_HALTED event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.228

Level: UE-ERROR

Short Syntax: DLS.228 DLS, ENTER_BUSY rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.228 DLS forwarder received a ENTER_BUSY event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing an ENTER_BUSY event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.229

Level: UE-ERROR

Short Syntax: DLS.229 DLS, EXIT_BUSY rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.229 DLS forwarder received a EXIT_BUSY event in bad state *state* for the DLS session with data link id *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: While processing an EXIT_BUSY event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.230

Level: UE-ERROR

Short Syntax: DLS.230 DLS, REACHACK rcvd in bad state *state* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.230 DLS forwarder received a REACHACK event in bad state *state* for the DLS session with data link id *source_mac_address*->

dest_mac_address, sap *source_sap*-> *dest_sap*

Description: While processing a REACHACK event, the DLS state machine discovered the event occurring in an unexpected state.

DLS.231

Level: C-INFO

Short Syntax: DLS.231 DLS, CANUREACH to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.231 DLS, CANUREACH to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS successfully sent out a CANUREACH SSP message over TCP to its DLS peer.

DLS.232

Level: C-INFO

Short Syntax: DLS.232 DLS, ICANREACH to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.232 DLS, ICANREACH to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS successfully sent out an ICANREACH SSP message over TCP to its DLS peer.

DLS.233

Level: C-INFO

Short Syntax: DLS.233 DLS, REACH_ACK to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.233 DLS, REACH_ACK to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS successfully sent out a REACH_ACK SSP message over TCP to its DLS peer.

DLS.234

Level: C-INFO

Short Syntax: DLS.234 DLS, CONTACT to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.234 DLS, CONTACT to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS successfully sent out a CONTACT SSP message over TCP to its DLS peer.

DLS.235

Level: C-INFO

Short Syntax: DLS.235 DLS, CONTACTED to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.235 DLS, CONTACTED to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS successfully sent out a CONTACTED SSP message over TCP to its DLS peer.

DLS.236

Level: C-INFO

Short Syntax: DLS.236 DLS, RESTART_DL to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.236 DLS, RESTART_DL to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS successfully sent out a RESTART_DL SSP message over TCP to its DLS peer.

DLS.237

Level: C-INFO

Short Syntax: DLS.237 DLS, DL_RESTARTED to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.237 DLS, DL_RESTARTED to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS successfully sent out a DL_RESTARTED SSP message over TCP to its DLS peer.

DLS.238

Level: C-INFO

Short Syntax: DLS.238 DLS, ENTER_BUSY to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.238 DLS, ENTER_BUSY to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS successfully sent out an ENTER_BUSY SSP message over TCP to its DLS peer.

DLS.239

Level: C-INFO

Short Syntax: DLS.239 DLS, EXIT_BUSY to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.239 DLS, EXIT_BUSY to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS successfully sent out an EXIT_BUSY SSP message over TCP to its DLS peer.

DLS.240

Level: C-INFO

Short Syntax: DLS.240 DLS, HALT_DL to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.240 DLS, HALT_DL to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS successfully sent out a HALT_DL SSP message over TCP to its DLS peer.

DLS.241

Level: C-INFO

Short Syntax: DLS.241 DLS, DL_HALTED to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.241 DLS, DL_HALTED to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS successfully sent out a DL_HALTED SSP message over TCP to its DLS peer.

DLS.242

Level: C-INFO

Short Syntax: DLS.242 DLS, HALT_DL_NOACK to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.242 DLS, HALT_DL_NOACK to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS successfully sent out a HALT_DL_NOACK SSP message over TCP to its DLS peer.

DLS.243

Level: C-INFO

Short Syntax: DLS.243 DLS, TEST_CIRCUIT_RSP to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.243 DLS, TEST_CIRCUIT_RSP to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS successfully sent out a TEST_CIRCUIT_RSP SSP message over TCP to its DLS peer.

DLS.244

Level: UI-ERROR

Short Syntax: DLS.244 DLS, FAILED to send CANUREACH to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.244 DLS, FAILED to send CANUREACH to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: A CANUREACH SSP control message was not sent because either there are no buffers or the DLSw partner does not support the source sap in its DLSw capabilities exchange SAP list.

DLS.245

Level: UI-ERROR

Short Syntax: DLS.245 DLS, FAILED to send ICANREACH to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.245 DLS, FAILED to send ICANREACH to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: An ICANREACH SSP control message was not sent because either there are no buffers or the DLSw partner does not support the source sap in its DLSw capabilities exchange SAP list.

DLS.246

Level: UI-ERROR

Short Syntax: DLS.246 DLS, FAILED to send REACH_ACK to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.246 DLS, FAILED to send REACH_ACK to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: Due to lack of buffers, a REACH_ACK SSP control message could not be sent out.

DLS.247

Level: UI-ERROR

Short Syntax: DLS.247 DLS, FAILED to send CONTACT to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.247 DLS, FAILED to send CONTACT to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Due to lack of buffers, a CONTACT SSP control message could not be sent out.

DLS.248

Level: UI-ERROR

Short Syntax: DLS.248 DLS, FAILED to send CONTACTED to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.248 DLS, FAILED to send CONTACTED to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Due to lack of buffers, a CONTACTED SSP control message could not be sent out.

DLS.249

Level: UI-ERROR

Short Syntax: DLS.249 DLS, FAILED to send RESTART_DL to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.249 DLS, FAILED to send RESTART_DL to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Due to lack of buffers, a RESTART_DL SSP control message could not be sent out.

DLS.250

Level: UI-ERROR

Short Syntax: DLS.250 DLS, FAILED to send DL_RESTARTED to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.250 DLS, FAILED to send DL_RESTARTED to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Due to lack of buffers, a DL_RESTARTED SSP control message could not be sent out.

DLS.251

Level: UI-ERROR

Short Syntax: DLS.251 DLS, FAILED to send ENTER_BUSY to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.251 DLS, FAILED to send ENTER_BUSY to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Due to lack of buffers, an ENTER_BUSY SSP control message could not be sent out.

DLS.252

Level: UI-ERROR

Short Syntax: DLS.252 DLS, FAILED to send EXIT_BUSY to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.252 DLS, FAILED to send EXIT_BUSY to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Due to lack of buffers, an EXIT_BUSY SSP control message could not be sent out.

DLS.253

Level: UI-ERROR

Short Syntax: DLS.253 DLS, FAILED to send HALT_DL to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.253 DLS, FAILED to send HALT_DL to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Due to lack of buffers, a HALT_DL SSP control message could not be sent out.

DLS.254

Level: UI-ERROR

Short Syntax: DLS.254 DLS, FAILED to send DL_HALTED to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.254 DLS, FAILED to send DL_HALTED to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Due to lack of buffers, a DL_HALTED SSP control message could not be sent out.

DLS.255

Level: UI-ERROR

Short Syntax: DLS.255 DLS, FAILED to send HALT_DL_NOACK to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.255 DLS, FAILED to send HALT_DL_NOACK to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Due to lack of buffers, a

HALT_DL_NOACK SSP control message could not be sent out.

DLS.256

Level: UI-ERROR

Short Syntax: DLS.256 DLS, FAILED to send TEST_CIRCUIT_RSP to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.256 DLS, FAILED to send TEST_CIRCUIT_RSP to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: Due to lack of buffers, a TEST_CIRCUIT_RSP SSP control message could not be sent out.

DLS.258

Level: C-INFO

Short Syntax: DLS.258 DLS, XIDFRAME to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.258 DLS, XIDFRAME to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS successfully sent out a XIDFRAME SSP message over TCP to its DLS peer.

DLS.259

Level: C-INFO

Short Syntax: DLS.259 DLS, DGRMFRAME to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.259 DLS, DGRMFRAME to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS successfully sent out a DGRMFRAME SSP message over TCP to its DLS peer.

DLS.260

Level: C-INFO

Short Syntax: DLS.260 DLS, DATAFRAME to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.260 DLS, DATAFRAME to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS successfully sent out a DATAFRAME SSP message over TCP or UDP to its DLSw peer.

DLS.261

Level: C-INFO

Short Syntax: DLS.261 DLS, INFOFRAME to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.261 DLS, INFOFRAME to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS successfully sent out an INFOFRAME SSP message over TCP to its DLS peer.

DLS.262

Level: C-INFO

Short Syntax: DLS.262 DLS, SSP msg TEST_CIRCUIT_REQ received from *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.262 DLS forwarder received a SSP TEST_CIRCUIT_REQ message over TCP connection to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of TEST_CIRCUIT_REQ over TCP.

DLS.263

Level: C-INFO

Short Syntax: DLS.263 DLS, SSP msg TEST_CIRCUIT_RSP received from *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.263 DLS forwarder received a SSP TEST_CIRCUIT_RSP message over TCP connection to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of TEST_CIRCUIT_RSP over TCP.

DLS.264

Level: C-INFO

Short Syntax: DLS.264 *intfmod*, Flushed Info frame sent, *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.264 *intfmod*, Flushed info frame sent, *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: An information frame received from DLS cloud was successfully flushed to a LLC end station or to the local APPN.

DLS.265

Level: C-INFO

Short Syntax: DLS.265 *intfmod*, TEST_C frame sent, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.265 *intfmod*, TEST_C frame sent, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: A TEST Command frame was successfully sent to a LLC end station or to the local APPN as result of receiving a CANUREACH from a DLSw peer router.

DLS.266

Level: C-INFO

Short Syntax: DLS.266 *intfmod*,, TEST_R frame sent, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.266 *intfmod*,, TEST_R frame sent, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: A TEST Response frame was successfully sent to a LLC end station or to the local APPN as result of receiving an ICANREACH from a DLSw peer router.

DLS.268

Level: C-INFO

Short Syntax: DLS.268 LLC, UI frame sent, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.268 LLC, UI frame sent, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: An UI frame was successfully sent to a LLC end station.

DLS.269

Level: U-INFO

Short Syntax: DLS.269 *intfmod*, TEST_C frame refused by st mch, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.269 *intfmod*, TEST_C frame refused by st mch, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: A TEST command frame was not processed by the LLC or APPN interface module's state machine as it could not be successfully relayed via DLS. This instance is normal for TEST command frame as DLS copies the frame and still refuses the frame so that it can also be sent via the bridge path.

DLS.270

Level: U-INFO

Short Syntax: DLS.270 *intfmod*, TEST_R frame refused by st mch, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.270 *intfmod*, TEST_R frame refused by st mch, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: A TEST response frame was not processed by the LLC or APPN interface module's state machine as it could not be successfully relayed via DLS.

DLS.271

Level: U-INFO

Short Syntax: DLS.271 *intfmod*, XID_C frame refused by st mch, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.271 *intfmod*, XID_C frame refused by st mch, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: A XID command frame was not processed by the LLC or APPN interface module's state machine as it could not be successfully relayed via DLS.

DLS.272

Level: U-INFO

Short Syntax: DLS.272 *intfmod*, XID_R frame refused by st mch, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.272 *intfmod*, XID_R frame refused by st mch, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: A XID response frame was not processed by the LLC or APPN interface module's state machine as it could not be successfully relayed via DLS.

DLS.273

Level: U-INFO

Short Syntax: DLS.273 LLC, UI frame refused by st mch, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.273 LLC, UI frame refused by st mch, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: An UI frame was not processed by the DLS-LLC interface module's state machine as it could not be successfully relayed via DLS.

DLS.274

Level: U-INFO

Short Syntax: DLS.274 LLC, INFO frame refused by st mch, *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.274 LLC, INFO frame refused by st mch, *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: An INFO frame was not processed by the DLS-LLC interface module's state machine as it could not be successfully relayed via DLS.

DLS.276

Level: C-INFO

Short Syntax: DLS.276 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.276 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: LLC or APPN interface module for the DLS received a NETUP event.

DLS.277

Level: C-INFO

Short Syntax: DLS.277 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.277 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: LLC or APPN interface module for the DLS received a NETDOWN event.

DLS.278

Level: C-INFO

Short Syntax: DLS.278 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.278 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: LLC or APPN interface module for the DLS received a CONNECT_IND event.

DLS.279

Level: C-INFO

Short Syntax: DLS.279 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.279 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: LLC or APPN interface module for the DLS received a CONNECT_CONF event.

DLS.280

Level: C-INFO

Short Syntax: DLS.280 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.280 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: LLC or APPN interface module for the DLS received a ENTER_BUSY event.

DLS.281

Level: C-INFO

Short Syntax: DLS.281 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.281 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: LLC or APPN interface module for the DLS received a EXIT_BUSY event.

DLS.282

Level: C-INFO

Short Syntax: DLS.282 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.282 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: LLC or APPN interface module for the DLS received a DISC_IND event.

DLS.283

Level: C-INFO

Short Syntax: DLS.283 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.283 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS received a DISC_CONF event.

DLS.284

Level: C-INFO

Short Syntax: DLS.284 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.284 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS received a RESET_IND event.

DLS.285

Level: C-INFO

Short Syntax: DLS.285 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.285 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS received a TEST_C event.

DLS.286

Level: C-INFO

Short Syntax: DLS.286 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.286 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS received a TEST_R event.

DLS.287

Level: C-INFO

Short Syntax: DLS.287 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.287 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS received a XID_C event.

DLS.288

Level: C-INFO

Short Syntax: DLS.288 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.288 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS received a XID_R event.

DLS.289

Level: C-INFO

Short Syntax: DLS.289 LLC, event LLCIM_LLC_UI received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.289 LLC, event LLCIM_LLC_UI received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC interface module for the DLS received a LLCIM_LLC_UI event from LLC.

DLS.290

Level: C-INFO

Short Syntax: DLS.290 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.290 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS received a DLC_START_DL event from DLS.

DLS.291

Level: C-INFO

Short Syntax: DLS.291 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.291 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS received a DLC_RESOLVE_R event from DLS.

DLS.292

Level: C-INFO

Short Syntax: DLS.292 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.292 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: LLC or APPN interface module for the DLS received a DLC_CONTACT event from DLS.

DLS.293

Level: C-INFO

Short Syntax: DLS.293 LLC, event LLCIM_DLC_DGRM received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.293 LLC, event LLCIM_DLC_DGRM received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: LLC interface module for the DLS received a LLCIM_DLC_DGRM event from DLS.

DLS.294

Level: C-INFO

Short Syntax: DLS.294 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.294 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: LLC or APPN interface module for the DLS received a DLC_XID event from DLS.

DLS.295

Level: C-INFO

Short Syntax: DLS.295 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.295 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: LLC or APPN interface module for the DLS received a DLC_HALT_DL event from DLS.

DLS.296

Level: C-INFO

Short Syntax: DLS.296 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.296 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: LLC or APPN interface module for the DLS received a DLC_ENTER_BUSY event from DLS.

DLS.297

Level: C-INFO

Short Syntax: DLS.297 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.297 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: LLC or APPN interface module for the DLS received a DLC_EXIT_BUSY event from DLS.

DLS.298

Level: C-INFO

Short Syntax: DLS.298 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.298 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: LLC or APPN interface module for the DLS received a INFOFRAME event.

DLS.299

Level: C-INFO

Short Syntax: DLS.299 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.299 *intfmod*, event *eventname* received for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: LLC or APPN interface module for the DLS received a DLC_INFO event from DLS.

DLS.300

Level: C-INFO

Short Syntax: DLS.300 *intfmod*, Transition to *statename* state for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.300 *intfmod*, Transition to *statename* state for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: LLC or APPN interface module for the DLS is transitioning to the CONTACTED state.

DLS.301

Level: C-INFO

Short Syntax: DLS.301 *intfmod*, Transition to *statename* state for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.301 *intfmod* Transition to *statename* state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS is transitioning to the RESOLVE_PEND state.

DLS.302

Level: C-INFO

Short Syntax: DLS.302 *intfmod*, Transition to *statename* state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.302 *intfmod*, Transition to *statename* state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS is transitioning to the CONNECTED state.

DLS.303

Level: C-INFO

Short Syntax: DLS.303 *intfmod*, Transition to *statename* state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.303 *intfmod* Transition to *statename* state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS is transitioning to the CONNECT_PEND state.

DLS.304

Level: C-INFO

Short Syntax: DLS.304 *intfmod*, Transition to *statename* state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.304 *intfmod* Transition to *statename* state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS is transitioning to the CONTACT_PEND state.

DLS.305

Level: C-INFO

Short Syntax: DLS.305 *intfmod*, Transition to *statename* state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.305 *intfmod* Transition to *statename* state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS is transitioning to the DISCONNECTED state.

DLS.306

Level: C-INFO

Short Syntax: DLS.306 *intfmod*, Transition to *statename* state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.306 *intfmod* Transition to *statename* state for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS is transitioning to the DISC_PEND state.

DLS.307

Level: UE-ERROR

Short Syntax: DLS.307 DLSw disabled no mem for buffers

Long Syntax: DLS.307 DLSw forwarder disabled no memory for buffers

Description: The Data Link Switching forwarder has been disabled because there was not enough memory to create the DLS private buffer pool.

DLS.308

Level: UE-ERROR

Short Syntax: DLS.308 DLSw disabled no mem for llcim struct

Long Syntax: DLS.308 DLSw forwarder disabled no memory for llcim structures

Description: The Data Link Switching forwarder has been disabled because there was not enough memory to create necessary llcim data structures.

DLS.309

Level: UE-ERROR

Short Syntax: DLS.309 DLSw disabled no mem for tcpim struct

Long Syntax: DLS.309 DLSw forwarder disabled no memory for tcpim structures

Description: The Data Link Switching forwarder has been disabled because there was not enough memory to create necessary tcpim data structures.

DLS.310

Level: UE-ERROR

Short Syntax: DLS.310 DLSw disabled no mem for sdlcim struct

Long Syntax: DLS.310 DLSw forwarder disabled no memory for sdlcim structures

Description: The Data Link Switching forwarder has

been disabled because there was not enough memory to create necessary sdlcim data structures.

DLS.311

Level: UE-ERROR

Short Syntax: DLS.311 DLSw disabled no mem for group struct

Long Syntax: DLS.311 DLSw forwarder disabled no memory for group protocol structures

Description: The Data Link Switching forwarder has been disabled because there was not enough memory to create necessary group protocol data structures.

DLS.312

Level: UE-ERROR

Short Syntax: DLS.312 DLSw disabled no mem for dl corr array

Long Syntax: DLS.312 DLSw forwarder disabled no memory for dl correlator array

Description: The Data Link Switching forwarder has been disabled because there was not enough memory to create necessary dl correlator array.

Cause: Cannot allocate necessary memory for the dl correlator array.

Action: Reduce the maximum number of DLSw sessions.

DLS.313

Level: C-INFO

Short Syntax: DLS.313 *intfmod*, INFO frame sent, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.313 *intfmod*, INFO frame sent, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: An INFO frame received from DLS cloud was successfully sent to a LLC end station or to the local APPN.

DLS.314

Level: C-INFO

Short Syntax: DLS.314 TCP, cfg xmit buf too large for group *group*, clipped to *transmit_buffer_size*

Long Syntax: DLS.314 TCP, configured transmit buffer size too large for group *group*, clipped to *transmit_buffer_size*

Description: The user configured a TCP transmit buffer size in the group configuration that cannot be handled by the router. It has automatically been set to a lower value than can be allocated by the router.

DLS.315

Level: C-INFO

Short Syntax: DLS.315 TCP, cfg xmit buf too large for *neighbour_address*, clipped to *transmit_buffer_size*

Long Syntax: DLS.315 TCP, configured transmit buffer size too large for *neighbour_address*, clipped to *transmit_buffer_size*

Description: The user configured a TCP transmit buffer size that cannot be handled by the router. It has automatically been set to a lower value that can be allocated by the router.

DLS.316

Level: C-INFO

Short Syntax: DLS.316 DLS, CANUREACH frame coll, frame ign, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.316 DLS, CANUREACH frame collision, frame ignored, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: A CANUREACH frame was received from the DLS cloud, but could not be processed because a CANUREACH is already outstanding from this router for the MAC addresses and SAPs specified in the CANUREACH, and the origin MAC address for the existing circuit is greater than the origin MAC address specified in the CANUREACH.

DLS.317

Level: C-INFO

Short Syntax: DLS.317 *intfmod*, XID_C frame sent, len= *xid_data_len*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.317 *intfmod*, XID_C frame sent, len= *xid_data_len*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: An XID_C frame was successfully sent to a LLC end station or to local APPN.

DLS.318

Level: C-INFO

Short Syntax: DLS.318 *intfmod*, XID_R frame sent, len= *xid_data_len*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.318 *intfmod*, XID_R frame sent, len= *xid_data_len*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: An XID_R frame was successfully sent to

a LLC end station or to local APPN.

DLS.319

Level: C-INFO

Short Syntax: DLS.319 *intfmod*, XID_C dropped, len= *xid_data_len*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.319 *intfmod*, XID_C dropped, len= *xid_data_len*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: A received XID_C frame was discarded because a previously received XID_C is being processed by DLSw. This XID_C is considered a duplicate. This will occur frequently since the end station retries XID_Cs.

DLS.320

Level: C-INFO

Short Syntax: DLS.320 LLC, XID_R dropped, len= *xid_data_len*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.320 LLC, XID_R dropped, len= *xid_data_len*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: A received XID_R frame was discarded because there is no XID command outstanding. This occurs normally since DLSw retries XID_Cs and it is possible for many XID_Rs to come back.

DLS.321

Level: UI-ERROR

Short Syntax: DLS.321 LLC, XIDFRAME dropped-bad XID state, len= *xid_data_len*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.321 LLC, XIDFRAME dropped-bad XID state, len= *xid_data_len*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: An unexpected received XIDFRAME frame was discarded. The LLC does not normally expect to receive an XID in this state.

Action: None, unless you are having a problem establishing connections between the end stations described in the message.

DLS.322

Level: C-INFO

Short Syntax: DLS.322 *intfmod*, unexpected null XID, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.322 *intfmod*, unexpected null XID, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: A null XID has been received after the SABME/UA exchange. This is not normal in this state and indicates that the end station is trying to start a new session. The current DLSw session will be terminated.

DLS.323

Level: C-INFO

Short Syntax: DLS.323 DLS, activ XIDFRAME dropped-bad state, len= *xid_data_len*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.323 DLS, activ XIDFRAME dropped-bad state, len= *xid_data_len*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: A received activation XIDFRAME is not allowed in the DLSw DLS_CONNECTED or DLS_CONNECT_PENDING state. The frame is discarded.

DLS.324

Level: C-INFO

Short Syntax: DLS.324 DLS, activ XID dropped-bad state, len= *xid_data_len*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.324 DLS, activ XID dropped-bad state, len= *xid_data_len*, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: A received activation XID is not allowed in the DLSw DLS_CONNECTED or DLS_CONNECT_PENDING state. The frame is discarded.

DLS.325

Level: UI-ERROR

Short Syntax: DLS.325 DLS, Session not created - Maximum Number of DLS Sessions exceeded.

Long Syntax: DLS.325 DLS, Session not created - Maximum Number of DLS Sessions exceeded.

Description: A DLSw Session is not created since the configured Maximum Number of DLSw sessions is exceeded.

DLS.326

Level: UI-ERROR

Short Syntax: DLS.326 close transport cnn to *ip_address*, unrecoverable SSP sync error

Long Syntax: DLS.326 closing transport connection to *ip_address*, unrecoverable SSP syncronization error

Description: Due to an error in a received DLSw SSP message, the TCP session must be closed in an attempt to recover. The cause is either due to an invalid message length in the previous SSP message or from an unsupported DLSw SSP version in the current message.

DLS.327

Level: UE-ERROR

Short Syntax: DLS.327 DL_HALTED timer expired, closing session *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.327 DL_HALTED timer expired, closing session *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: An expected DL_HALTED SSP message has not been received in response to a previously sent HALT_DL message. As a result, the connection is now considered to be in the disconnected state.

DLS.328

Level: C-INFO

Short Syntax: DLS.328 DLSw, Reconnect TCP connection to Neighbor *ip_address*

Long Syntax: DLS.328 TCP, Automatically reconnecting to neighbor at *ip_address*

Description: A previously down TCP connection is being re-established because the user has it defined as 'Active'. To prevent automatic reconnection, define this connection as 'Passive'.

DLS.330

Level: C-INFO

Short Syntax: DLS.330 sdlc link sta reopen addr *link_address* nt *network ID*

Long Syntax: DLS.330 SDLC link station reopened address *link_address* net *network ID*

Description: The SDLC link station for the link address has been successfully re-opened on the network interface because the SDLC link station was added again on the SDLC console.

DLS.331

Level: C-INFO

Short Syntax: DLS.331 TCP, no mem for cnn to nbr at *ip_address*

Long Syntax: DLS.331 TCP, cannot create a new connection to neighbor at *ip_address* due to a memory shortage

Description: There is insufficient memory in the router to create a new TCP connection.

DLS.332

Level: UI-ERROR

Short Syntax: DLS.332 Ptr to SCB is NULL. Event: *event*; DLC: *dlc_name*; CCB State: *ccb_state*; MAC: *source_mac_address*-> *dest_mac_address*, SAP: *source_sap*-> *dest_sap*

Long Syntax: DLS.332 Ptr to SCB is NULL. Event: *event*; DLC: *dlc_name*; CCB State: *ccb_state*; MAC: *source_mac_address*-> *dest_mac_address*, SAP: *source_sap*-> *dest_sap*

Description: While processing a DLC event that expects and requires a valid DLS SCB, the DLS state machine discovered that the passed pointer to the SCB was NULL.

DLS.333

Level: UI-ERROR

Short Syntax: DLS.333 Ptr to SCB is NULL and ptr to CCB is also NULL. Event: *event*

Long Syntax: DLS.333 Ptr to SCB is NULL and ptr to CCB is also NULL. Event: *event*

Description: While processing a DLC event that expects and requires a valid DLS SCB and a DLC CCB, the DLS state machine discovered that the passed pointers to both were NULL.

DLS.334

Level: UI-ERROR

Short Syntax: DLS.334 Ptr to SCB is NULL and CCB identifier is invalid. Event: *event*

Long Syntax: DLS.334 Ptr to SCB is NULL and CCB identifier is invalid. Event: *event*

Description: While processing a DLC event that expects and requires a valid DLS SCB and a DLC CCB, the DLS state machine discovered that the passed pointer to the SCB was NULL and the DLC CCB identifier was invalid.

DLS.335

Level: UI-ERROR

Short Syntax: DLS.335 SSP msg received. Ptr to SCB is NULL. Xport state: *transport_state*; IP: *remote_ip_addr*

Long Syntax: DLS.335 SSP msg received. Ptr to SCB is NULL. Xport state: *transport_state*; IP: *remote_ip_addr*

Description: While processing an SSP event that expects and requires a valid DLS SCB, the DLS state

machine discovered that the passed pointer to the SCB was NULL.

DLS.338

Level: UE-ERROR

Short Syntax: DLS.338 Could not send ctrl msg, closing session *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.338 Could not send control message, closing session *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: The router could not send a response control message due to a lack of a buffer to send it in. Terminate the DLSw session by sending a HALT_DL_NOACK.

DLS.339

Level: C-INFO

Short Syntax: DLS.339 DLS, Cleanup HALT_DL_NOACK to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.339 DLS, Cleanup HALT_DL_NOACK to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS successfully sent out a HALT_DL_NOACK SSP message over TCP to its DLS peer as a result of an out of buffer condition when attempting to send a control message.

DLS.340

Level: C-INFO

Short Syntax: DLS.340 Sent Capex Request to *ip_address*.

Long Syntax: DLS.340 A Capabilities Exchange Request has been sent to the DLSw neighbor at *ip_address*.

Description: The DLSw TCP Interface module determined that the TCP Transport is capable of conducting a Capabilities Exchange. The DLSw TCP interface module has sent the Capabilities Exchange Request to the DLSw neighbor.

DLS.341

Level: C-INFO

Short Syntax: DLS.341 Received Capex Request from *ip_address*.

Long Syntax: DLS.341 A Capabilities Exchange Request has been received from the DLSw neighbor at *ip_address*.

Description: The DLSw TCP Interface module received a Capabilities Exchange message from a DLSw neighbor. The parsing module determined that the message type is a request.

DLS.342

Level: C-INFO

Short Syntax: DLS.342 Sent Capex Pos. Response to *ip_address*.

Long Syntax: DLS.342 A Capabilities Exchange Positive Response has been sent to the DLSw neighbor at *ip_address*.

Description: The DLSw Capabilities Exchange parsing module successfully processed a Capabilities Exchange Request from a DLSw neighbor.

DLS.343

Level: C-INFO

Short Syntax: DLS.343 Received Capex Pos. Response from *ip_address*.

Long Syntax: DLS.343 A Capabilities Exchange Positive Response has been received from the DLSw neighbor at *ip_address*.

Description: The DLSw TCP Interface module received a Capabilities Exchange message from a DLSw neighbor. The parsing module determined that the message type is a Positive Response.

DLS.344

Level: UI-ERROR

Short Syntax: DLS.344 Sent Capex Neg. Response to *ip_address* reason *reason_code* offset *offset_value*.

Long Syntax: DLS.344 A Capabilities Exchange Negative Response has been sent to the DLSw neighbor at *ip_address* Reason *reason_code* Offset *offset_value*.

Description: The DLSw Capabilities Exchange parsing module processed a Capabilities Exchange Request from a DLSw neighbor. The Request was determined to contain an error or invalid Control Vector.

DLS.345

Level: UI-ERROR

Short Syntax: DLS.345 Received Capex Neg. Response from *ip_address* reason *reason_code* offset *offset_value*.

Long Syntax: DLS.345 A Capabilities Exchange Negative Response has been received from the DLSw neighbor at *ip_address*. The Negative Response contained a Reason of *reason_code* at Offset *offset_value*.

Description: The DLSw TCP Interface module received a Capabilities Exchange message from a DLSw

neighbor. The parsing module determined that the message type is a Negative Response.

DLS.346

Level: UI-ERROR

Short Syntax: DLS.346 Capex Aborted!, *ip_address* is assumed to be DLSw RFC 1434+ compliant.

Long Syntax: DLS.346 Capabilities Exchange has been aborted with the neighbor at *ip_address*. DLSw processing will continue by assuming that his neighbor is DLSw RFC 1434+ compliant.

Description: The DLSw Capabilities Exchange manager determined that the neighbor is not capable of supporting DLSw AIW_V1. Assume that the neighbor is capable of supporting DLSw RFC 1434+.

DLS.347

Level: UI-ERROR

Short Syntax: DLS.347 Received unknown Capex message from *ip_address*.

Long Syntax: DLS.347 Received Unknown Capabilities Exchange Message from the neighbor at *ip_address*.

Description: The DLSw neighbor sent an Unknown Capabilities Exchange message. Capabilities Exchange may fail if this was meant to be a Request or Response.

DLS.348

Level: UI-ERROR

Short Syntax: DLS.348 No memory available to create DLS Capex message for *ip_address*.

Long Syntax: DLS.348 An attempt to allocate the memory required to build a Capabilities Exchange message has failed. The message can not be sent to the neighbor at *ip_address*.

Description: There is no memory available to allocate the resources that the router needs to build a Capabilities Exchange message. Capabilities Exchange will fail with this neighbor.

DLS.349

Level: UI-ERROR

Short Syntax: DLS.349 Capex Failed! *ip_address* is not DLSw AIW_V1 compliant.

Long Syntax: DLS.349 Capabilities Exchange has failed with the neighbor at *ip_address*. DLSw processing can not continue. This neighbor is non compliant to DLSw AIW_V1.

Description: The DLSw Capabilities Exchange manager determined that the neighbor is not capable of supporting DLSw AIW_V1. The TCP Transport connection will be terminated with the neighbor.

DLS.350

Level: C-INFO

Short Syntax: DLS.350 Capex Successful! *ip_address* is DLSw AIW compliant.

Long Syntax: DLS.350 Capabilities Exchange has completed successfully with the neighbor at *ip_address*. DLSw processing can now continue in a DLSw AIW compliant mode.

Description: The DLSw Capabilities Exchange manager determined that the neighbor is capable of supporting DLSw AIW.

DLS.351

Level: C-INFO

Short Syntax: DLS.351 DLS, SSP msg IFCM *indmsg ackmsg* received from *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.351 DLS, DLS forwarder received a SSP IFCM *indmsg ackmsg* message over TCP connection to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol (SSP) message of an Isolated Flow Control Message (IFCM).

DLS.352

Level: UI-ERROR

Short Syntax: DLS.352 DLS, SSP msg received carrying flow control data. Ptr to SCB is NULL. Xport state: *transport_state*; IP: *remote_ip_addr*

Long Syntax: DLS.352 DLS, SSP msg received carrying flow control data. Ptr to SCB is NULL. Xport state: *transport_state*; IP: *remote_ip_addr*

Description: The DLS state machine discovered that the passed pointer to the SCB was NULL while processing an SSP event containing flow control data that expects and requires a valid DLS SCB.

DLS.353

Level: UI-ERROR

Short Syntax: DLS.353 DLS, Sender granted units for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* is less than zero

Long Syntax: DLS.353 DLS, Sender granted units for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* is less than zero

Description: The sender granted Service Access Point (SAP) units for this circuit and the SAP was

decremented to a value less than zero. This is a protocol violation and the router took the circuit down.

DLS.354

Level: UI-ERROR

Short Syntax: DLS.354 DLS, Received increment window for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* when window equal maximum size

Long Syntax: DLS.354 DLS, Received increment window for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* when window equal maximum size

Description: DLSw received an increment window operator while the window size is equal to the maximum size.

DLS.355

Level: UI-ERROR

Short Syntax: DLS.355 DLS, Received decrement window for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* when window size equal 1

Long Syntax: DLS.355 DLS, Received decrement window for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* when window size equal 1

Description: DLSw received a decrement window operator while the window size is equal to 1.

DLS.356

Level: UI-ERROR

Short Syntax: DLS.356 DLS, Received unrecognized flow control operator for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.356 DLS, Received unrecognized flow control operator for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Received an unrecognized flow control operator.

DLS.357

Level: C-INFO

Short Syntax: DLS.357 DLS, FCIND *operator* recv'd for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* tx_grant= *txgrant* tx_window= *txwindow*

Long Syntax: DLS.357 DLS, FCIND *operator* recv'd for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* tx_grant= *txgrant* tx_window= *txwindow*

Description: The router received an Isolated Flow Control Message (IFCM) or piggybacked flow control indication Switch to Switch Protocol (SSP) message.

DLS.358

Level: UI-ERROR

Short Syntax: DLS.358 DLS, Unexpected flow control acknowledgement recv'd for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.358 DLS, Unexpected flow control acknowledgement recv'd for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: The router received an IFCM or piggybacked flow control acknowledgement in an invalid state.

DLS.359

Level: C-INFO

Short Syntax: DLS.359 DLS, SSP msg IFCM *operator* sent over TCP connection to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.359 DLS, DLS forwarder sent an IFCM SSP message *operator* over TCP connection to *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: The DLS forwarder sent a Switch to Switch Protocol (SSP) message of IFCM.

DLS.360

Level: UI-ERROR

Short Syntax: DLS.360 DLS, Receiver detected granted units exceeded for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.360 DLS, Receiver detected granted units exceeded for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: The router received an SSP_INFOFRAME or SSP_DGRMFRAME that caused the granted units to be exceeded.

DLS.361

Level: UI-ERROR

Short Syntax: DLS.361 DLS, FCACK expected before end of current window for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.361 DLS, Receiver expected flow control ack before end of current window for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: The receiver did not get flow control ack before the end of the current window.

DLS.362

Level: UI-ERROR

Short Syntax: DLS.362 DLS, Receiver attempted to increment window greater than maximum window size for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.362 DLS, Receiver attempted to increment window greater than maximum window size for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: The receiver attempted to increment the window beyond the maximum window size.

DLS.363

Level: C-INFO

Short Syntax: DLS.363 DLS, Receiver attempted to decrement window less than minimum window size for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.363 DLS, Receiver attempted to decrement window less than minimum window size for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: The receiver attempted to decrement the window beyond the minimum window size.

DLS.364

Level: C-INFO

Short Syntax: DLS.364 DLS, MAC cache hit, selecting *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.364 DLS, MAC cache hit, selecting *ip_address* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLSw found an entry in the MAC cache for the target MAC address of this circuit. The router selected the partner with the indicated IP address for this circuit.

DLS.365

Level: C-INFO

Short Syntax: DLS.365 DLS, MAC expl already outstd, queue exp req for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.365 DLS, MAC explorer already outstanding, queueing the explorer request for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLSw found an entry in the MAC cache for the target MAC address of this circuit with an explorer already outstanding. The router queued this MAC explorer request and will process it when the outstanding explorer completes.

DLS.366

Level: UI-ERROR

Short Syntax: DLS.366 DLS, No mem to create exp CB for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.366 DLS, No memory to create explorer control block for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Due to a memory shortage condition, DLSw could not create an explorer control block for the specified circuit. The exploration fails and the router does not start a circuit.

DLS.367

Level: C-INFO

Short Syntax: DLS.367 DLS, Creating new MAC cache entry for *target_mac_address*

Long Syntax: DLS.367 DLS, Creating new MAC cache entry for *target_mac_address*

Description: The router creates a new cache entry for the specified target MAC address.

DLS.368

Level: C-INFO

Short Syntax: DLS.368 DLS, Explorer priority wait timer expired for *target_mac_address*

Long Syntax: DLS.368 DLS, Explorer priority wait timer expired for *target_mac_address*

Description: The priority wait timer expired for the specified target MAC address. The router is now attempting to satisfy the explorer request with the known DLSw partners that can reach this MAC address.

DLS.369

Level: C-INFO

Short Syntax: DLS.369 DLS, MAC explorer satisfied for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.369 DLS, MAC explorer satisfied for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS found a partner DLSw router that satisfies the explorer for this circuit.

DLS.370

Level: C-INFO

Short Syntax: DLS.370 DLS, ICANREACH-ex timer expired for *target_mac_address*

Long Syntax: DLS.370 DLS, ICANREACH-ex timer expired for *target_mac_address*

Description: The ICANREACH-ex timer expired for the specified target MAC address. The router is now attempting to satisfy the explorer request with the known DLSw partners that can reach this MAC address.

DLS.371

Level: C-INFO

Short Syntax: DLS.371 DLS, MAC explorer failed for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.371 DLS, MAC explorer failed for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS failed to find a partner DLSw router that satisfies the MAC explorer for this circuit.

DLS.372

Level: C-INFO

Short Syntax: DLS.372 DLS, Reslvd tmr exp, tgt MAC expl failed for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.372 DLS, Resolved timer expired, target MAC explorer failed for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: The target MAC explorer resolved timer expired without receiving a DLC_RESOLVED from any of the DLCs for the specified circuit. The target MAC address explorer failed.

DLS.373

Level: C-INFO

Short Syntax: DLS.373 DLS, CANUREACH-ex rcvd while exploring for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.373 DLS, CANUREACH-ex received while exploring for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: The router received a CANUREACH-ex while it was already processing a previous CANUREACH-ex. The router ignored this new CANUREACH-ex request.

DLS.374

Level: UI-ERROR

Short Syntax: DLS.374 DLS, No mem to create tgt exp CB for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.374 DLS, No memory to create target explorer control block for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS could not create a target explorer control block for the specified circuit due to a memory shortage condition. The exploration fails and the router does not start a circuit.

DLS.375

Level: C-INFO

Short Syntax: DLS.375 DLS, DLC_RESOLVED not processed, CB not found *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.375 DLS, DLC_RESOLVED not processed, CB not found *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: A DLC sent DLC_RESOLVED. The router could not process the DLC_RESOLVED because the router could not find the corresponding control block. This may occur if the resolved timer already expired, or a DLC_RESOLVED from a different DLC already satisfied the target MAC address explorer.

DLS.376

Level: C-INFO

Short Syntax: DLS.376 DLS, ICANREACH-ex to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap* lfsize *largest_frame_size*

Long Syntax: DLS.376 DLS, ICANREACH-ex to *ip_address* sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap* lfsize *largest_frame_size*

Description: DLSw successfully sent out an ICANREACH_ex SSP message over TCP or UDP to its DLSw peer.

DLS.377

Level: C-INFO

Short Syntax: DLS.377 DLS, Deleting MAC cache entry for *target_mac_address*

Long Syntax: DLS.377 DLS, Deleting MAC cache entry for *target_mac_address*

Description: The router deleted the MAC cache entry for the specified address.

DLS.378

Level: C-INFO

Short Syntax: DLS.378 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.378 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS received a DLC_RESOLVE event from DLS.

DLS.379

Level: UI-ERROR

Short Syntax: DLS.379 LLC, FAILED pas opn stn, invld sapcb, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Long Syntax: DLS.379 LLC, FAILED passive open station, invalid sapcb, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Description: Passive open of station for LLC data link services failed because the SAP to open the station is invalid.

DLS.380

Level: UI-ERROR

Short Syntax: DLS.380 LLC, FAILED pass open stn, No memory, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Long Syntax: DLS.380 LLC, FAILED passive open stn, No memory, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Description: Passive open of station for LLC data link services failed because there is no memory available to create a control block to manage the connection.

DLS.381

Level: UI-ERROR

Short Syntax: DLS.381 LLC, FAILED pass open stn, rsn= *Reason*, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Long Syntax: DLS.381 LLC, FAILED passive open stn, rsn= *Reason*, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Description: Passive open of station for LLC data link services failed due to some problems within LLC. The reason code is indicative of the specific problem.

DLS.382

Level: C-INFO

Short Syntax: DLS.382 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.382 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS received a DLC_CS_CONFIRM event from DLS.

DLS.383

Level: C-INFO

Short Syntax: DLS.383 *intfmod*, event *eventname* rcvd drng exp for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.383 *intfmod*, event *eventname* received during exploration for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS received a DLC_HALT_NOACK event from DLS during exploration.

DLS.384

Level: C-INFO

Short Syntax: DLS.384 *intfmod*, event *eventname* rcvd drg tgt exp for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.384 *intfmod*, event *eventname* received during target exploration for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS received a DLC_HALT_NOACK event from DLS during target side exploration.

DLS.385

Level: C-INFO

Short Syntax: DLS.385 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.385 *intfmod*, event *eventname* received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC or APPN interface module for the DLS received a DLC_HALT_NOACK event from DLS.

DLS.386

Level: UI-ERROR

Short Syntax: DLS.386 *intfmod*, No mem to create LLC/APPN exp CB for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.386 *intfmod*, No memory to create LLC/APPN explorer control block for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Due to a memory shortage condition, LLCIM or APPNIM could not create an explorer control block for the specified circuit. The exploration fails and the router does not start a circuit.

DLS.387

Level: C-INFO

Short Syntax: DLS.387 LLC, Receivd passive open SABME for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.387 LLC, Receivd passive open SABME for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: The LLC Interface module for the DLS received a SABME for a station that the router had not opened. This causes a passive open for the LLC station.

DLS.388

Level: UI-ERROR

Short Syntax: DLS.388 *intfmod*, No mem to allocate LLC/APPN exp buffer for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.388 *intfmod*, No memory to allocate LLC/APPN explorer buffer for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Due to a memory shortage condition, the LLC or APPN interface module could not allocate an explorer buffer for the specified circuit. The exploration fails and the router does not start a circuit.

DLS.389

Level: UI-ERROR

Short Syntax: DLS.389 *intfmod*, No mem to create LLC/APPN tgt exp CB for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.389 *intfmod*, No memory to create LLC/APPN target explorer control block for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: Due to a memory shortage condition, LLCIM or APPNIM could not create a target explorer control block for the specified circuit. The exploration fails and the router does not start a circuit.

DLS.390

Level: C-INFO

Short Syntax: DLS.390 LLC, opened stn passive, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Long Syntax: DLS.390 LLC, opened stn passive, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Description: Passive open of station for LLC data link services succeeded.

DLS.391

Level: C-INFO

Short Syntax: DLS.391 LLC, chgd tx wdw frm *old_tx_window* to *new_tx_window* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.391 LLC, Changed transmit window from *old_tx_window* to *new_tx_window* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLCIM has changed the transmit window to match the receive window requested in an XID-3 received from a PU 2.1 LLC end station.

DLS.392

Level: UI-ERROR

Short Syntax: DLS.392 LLC, cld nt chg tx wdw frm *old_tx_window* to *new_tx_window* ret= *return_code* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.392 LLC, Could not change transmit window from *old_tx_window* to *new_tx_window*, return = *return_code* for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLCIM could not change the transmit window to match the window requested in an XID-3 received from a PU 2.1 LLC end station. The dl_open_station function call failed with the specified return_code.

DLS.393

Level: C-INFO

Short Syntax: DLS.393 Sent runtime Capex Request to *ip_address*.

Long Syntax: DLS.393 A Runtime Capabilities Exchange Request has been sent to the DLSw neighbor at *ip_address*.

Description: The router sent a runtime Capabilities Exchange Request message to the DLSw neighbor. This is the result of a change to the information that was initially exchanged.

DLS.394

Level: UI-ERROR

Short Syntax: DLS.394 Invalid interface number *interface* in Intf/SAP record with SAP *sap*

Long Syntax: DLS.394 Invalid interface number *interface* in Intf/SAP record with SAP *sap*

Description: DLSw detected an invalid configuration parameter at start-up time. The interface on which DLSw is to open an LLC SAP does not exist on the router. You should change the configuration and restart the router.

DLS.395

Level: UI-ERROR

Short Syntax: DLS.395 Invalid intf number *interface* for SDLC station record with link addr *link_address*

Long Syntax: DLS.395 Invalid intf number *interface* for SDLC station record with link addr *link_address*

Description: DLSw detected an invalid configuration parameter at start-up time. The interface specified for a DLSw SDLC link station either does not exist or is not of type SDLC. You should change the configuration and restart the router.

DLS.396

Level: UI-ERROR

Short Syntax: DLS.396 Invalid SAP number *sap* in Intf/SAP record for interface *interface*

Long Syntax: DLS.396 Invalid SAP number *sap* in Intf/SAP record for interface *interface*

Description: DLSw detected an invalid configuration parameter at start-up time. The SAP value that DLSw is to open on the specified interface is odd or outside the allowable range. Change the configuration and restart the router.

DLS.397

Level: UI-ERROR

Short Syntax: DLS.397 Invalid source SAP *sap* in SDLC record for intfc *interface*, addr *link_address*

Long Syntax: DLS.397 Invalid source SAP *sap* in SDLC record for intfc *interface*, addr *link_address*

Description: DLSw detected an invalid configuration parameter at start-up time. The source SAP for an SDLC link station is outside the allowable range. DLSw has not added this SDLC link station. You should change the configuration and restart the router.

DLS.398

Level: UI-ERROR

Short Syntax: DLS.398 Invalid TCP receive buffer size *buf_size* for a neighbor or group

Long Syntax: DLS.398 Invalid TCP receive buffer size *buf_size* for a neighbor or group

Description: DLSw detected an invalid configuration parameter at start-up time. The neighbor or group receive buffer size was outside the allowable range, but DLSw adjusted it to the nearest range limit and processed the configuration record anyway. Fix the configuration at some point by comparing with console information to isolate the problem and restart the router.

DLS.399

Level: UI-ERROR

Short Syntax: DLS.399 Invalid neighbor priority *priority_value* for neighbor or group record

Long Syntax: DLS.399 Invalid neighbor priority *priority_value* for neighbor or group record

Description: DLSw detected an invalid configuration parameter at start-up time. The neighbor priority value was outside the allowable range, but DLSw adjusted it to the nearest range limit and processed the configuration record anyway. Fix the configuration by comparing with console information to isolate the problem and restart the router.

DLS.400

Level: UI-ERROR

Short Syntax: DLS.400 Invalid priority wait timer *timer_value* (in tenth seconds)

Long Syntax: DLS.400 Invalid priority wait timer *timer_value* (in tenth seconds)

Description: DLSw detected an invalid configuration parameter at start-up time. The priority wait timer value was outside the allowable range, but DLSw adjusted it to the nearest range limit and will use the adjusted value. Fix the configuration and restart the router.

DLS.401

Level: UI-ERROR

Short Syntax: DLS.401 Invalid DLSw session priority *priority_value* for *protocol* DLSw sessions

Long Syntax: DLS.401 Invalid DLSw session priority *priority_value* for *protocol* DLSw sessions

Description: DLSw detected an invalid configuration parameter at start-up time. The session priority for either SNA or NetBIOS is out of range, but DLSw adjusted it to the nearest range limit and will use the adjusted value. Fix the configuration and restart the router.

DLS.402

Level: UI-ERROR

Short Syntax: DLS.402 Invalid session priority frame allocation value *frame_alloc_value*

Long Syntax: DLS.402 Invalid session priority frame allocation value *frame_alloc_value*

Description: DLSw detected an invalid configuration parameter at start-up time. The value for the number of frames to be sent at one of the four session priorities is out of range, but DLSw adjusted it to the nearest range limit and will use the adjusted value. Fix the configuration by comparing with console information to isolate the problem and restart the router.

DLS.403

Level: UI-ERROR

Short Syntax: DLS.403 Invalid NetBIOS MTU size *mtu_size*

Long Syntax: DLS.403 Invalid NetBIOS maximum transmission unit size *mtu_size*

Description: DLSw detected an invalid configuration parameter at start-up time. The NetBIOS MTU size is out of range, but DLSw adjusted it to the nearest range limit and will use the adjusted value. Fix the configuration and restart the router.

DLS.404

Level: UI-ERROR

Short Syntax: DLS.404 SNA SAP configured on interface *interface*, but not SAP 00

Long Syntax: DLS.404 SNA SAP configured on interface *interface*, but not SAP 00

Description: DLSw detected an invalid configuration condition at start-up time. One of the standard SNA SAPs (04, 08, or 0C) is open on an interface, but SAP 00 is not open on the same interface. Without SAP 00 open, SNA data link switching will not work. The router cannot establish circuits. If you did not intend to temporarily disable an interface for DLSw, fix the configuration by opening SAP 00, where needed, and restart the router.

DLS.405

Level: UI-ERROR

Short Syntax: DLS.405 Duplicate SDLC source MAC address *interface* detected

Long Syntax: DLS.405 Duplicate SDLC source MAC address *interface* detected

Description: DLSw detected an invalid configuration condition at start-up time. A DLSw SDLC source MAC address has been duplicated within the router, either on the same SDLC interface or on a different one. The router permits only one occurrence of a given source MAC address, and DLSw uses only the first one encountered. Fix the configuration and restart the router.

DLS.408

Level: C-INFO

Short Syntax: DLS.408 DLS, FCACK recv'd for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* rx_grant= *rxgrant* rx_window= *rxwindow*

Long Syntax: DLS.408 DLS, FCACK recv'd for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* rx_grant= *rxgrant* rx_window= *rxwindow*

Description: The router received an IFCM ACK or piggybacked flow control acknowledgement SSP message.

DLS.409

Level: C-INFO

Short Syntax: DLS.409 DLS, (PacingQ) DGRMFRAME to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.409 DLS, (PacingQ) DGRMFRAME to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS successfully sent out a DGRMFRAME SSP message from the PacingQ over TCP to its DLS peer.

DLS.410

Level: C-INFO

Short Syntax: DLS.410 DLS, (PacingQ) INFOFRAME to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.410 DLS, (PacingQ) INFOFRAME to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS successfully sent out an INFOFRAME SSP message from the PacingQ over TCP to its DLS peer.

DLS.411

Level: C-INFO

Short Syntax: DLS.411 DLS, Pool status for *pool* pool is *pstatus*

Long Syntax: DLS.411 DLS, Pool status for *pool* pool is *pstatus*

Description: The router reported the status of a DLSw buffer pool.

DLS.412

Level: UI-ERROR

Short Syntax: DLS.412 DLS, Received invalid window operator *operator* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap* after receiving RESET operator

Long Syntax: DLS.412 DLS, Received invalid window operator *operator* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap* after receiving RESET operator

Description: The router received an invalid window operator after receiving a RESET operator.

DLS.413

Level: C-INFO

Short Syntax: DLS.413 DLS, Pacing task called to process *operator* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.413 DLS, Pacing task called to process *operator* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: The router called the pacing operator scheduler to process a scheduled pacing action.

DLS.414

Level: C-INFO

Short Syntax: DLS.414 DLS, Receiver *source_mac_address* FCIND for *dest_mac_address-> source_sap*, sap *dest_sap->*

Long Syntax: DLS.414 DLS, Receiver *source_mac_address* FCIND for *dest_mac_address-> source_sap*, sap *dest_sap->*

Description: The receiver side of the pacing circuit processed a request to withhold or permit a flow control indication.

DLS.415

Level: C-INFO

Short Syntax: DLS.415 DLS, SSP msg NETBIOS_NQ received from *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.415 DLS forwarder received a SSP NETBIOS_NQ message over TCP connection to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of NETBIOS_NQ over TCP.

DLS.416

Level: C-INFO

Short Syntax: DLS.416 DLS, SSP msg NETBIOS_NR received from *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.416 DLS forwarder received a SSP NETBIOS_NR message over TCP connection to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of NETBIOS_NR over TCP.

DLS.417

Level: C-INFO

Short Syntax: DLS.417 DLS, SSP msg NETBIOS_ANQ received from *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.417 DLS forwarder received a SSP NETBIOS_ANQ message over TCP connection to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of NETBIOS_ANQ over TCP.

DLS.418

Level: C-INFO

Short Syntax: DLS.418 DLS, SSP msg NETBIOS_ANR received from *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.418 DLS forwarder received a SSP NETBIOS_ANR message over TCP connection to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: DLS forwarder received a Switch to Switch Protocol message of NETBIOS_ANR over TCP.

DLS.419

Level: C-INFO

Short Syntax: DLS.419 DLS, NETBIOS_NQ to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* lfsize *largest_frame_size*

Long Syntax: DLS.419 DLS, NETBIOS_NQ to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* lfsize *largest_frame_size*

Description: DLS successfully sent out a NETBIOS_NQ SSP message over TCP to its DLS peer.

DLS.420

Level: C-INFO

Short Syntax: DLS.420 DLS, NETBIOS_NR to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* lfsize *largest_frame_size*

Long Syntax: DLS.420 DLS, NETBIOS_NR to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* lfsize *largest_frame_size*

Description: DLS successfully sent out a NETBIOS_NR SSP message over TCP to its DLS peer.

DLS.421

Level: C-INFO

Short Syntax: DLS.421 DLS, NETBIOS_ANQ to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.421 DLS, NETBIOS_ANQ to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS successfully sent out a NETBIOS_ANQ SSP message over TCP to its DLS peer.

DLS.422

Level: C-INFO

Short Syntax: DLS.422 DLS, NETBIOS_ANR to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.422 DLS, NETBIOS_ANR to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: DLS successfully sent out a NETBIOS_ANR SSP message over TCP to its DLS peer.

DLS.423

Level: UE-ERROR

Short Syntax: DLS.423 no mem to init NetBIOS DLSw function

Long Syntax: DLS.423 no memory to initialize NetBIOS DLSw function

Description: There was not enough memory available to initialize the NetBIOS DLSw function. The router needs memory for a session control block and a UI traffic buffer pool.

DLS.425

Level: UI-ERROR

Short Syntax: DLS.425 NetBIOS SSP message received without reqd DLC header

Long Syntax: DLS.425 NetBIOS SSP message received without the required DLC header

Description: All NetBIOS SSP messages must have a DLC header. This SSP message did not have one, but the router will continue to process the frame. This indicates an RFC1795 compatibility problem.

DLS.426

Level: C-INFO

Short Syntax: DLS.426 DLS, Learning new NBName-IP assoc from *IPaddr* for *NBName*

Long Syntax: DLS.426 DLS, Learning new NBName-IP association from *IPaddr* for *NBName*

Description: DLS is learning a new NetBIOS name and IP association from an SSP message received from the peer DLS. This typically occurs during NETBIOS_NR message receipt time.

DLS.427

Level: UI-ERROR

Short Syntax: DLS.427 LLC, FAILED open NB stn, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Long Syntax: DLS.427 LLC, FAILED open NetBIOS stn, dst= *Destination*,src= *Source*,dsap= *Dsap*,ssap= *Ssap*

Description: The opening of a NetBIOS station for LLC data link services failed due to some problems within LLC.

DLS.428

Level: U-INFO

Short Syntax: DLS.428 LLC, NetBIOS UI frame disc (pool cong) for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.428 LLC, NetBIOS UI frame discarded (pool congested) for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: The LCC interface module is discarding the NetBIOS UI frames it received from the LLC. The NetBIOS UI frame buffer pool is congested.

DLS.429

Level: C-INFO

Short Syntax: DLS.429 DLS, NetBIOS function is enabled

Long Syntax: DLS.429 DLS, NetBIOS function is enabled

Description: DLSw NetBIOS function is now enabled. This occurs whenever the NetBIOS SAP (0xf0) is enabled on at least one port.

DLS.430

Level: C-INFO

Short Syntax: DLS.430 DLS, NetBIOS function is disabled

Long Syntax: DLS.430 DLS, NetBIOS function is disabled

Description: DLSw NetBIOS function is now disabled. This occurs whenever the NetBIOS SAP (0xf0) is disabled on all ports.

DLS.431

Level: C-INFO

Short Syntax: DLS.431 DLS, Broadcast CANUREACH-ex unsuccessful for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.431 DLS, Broadcast CANUREACH-ex unsuccessful for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: The router could not send CANUREACH-ex to any DLSw partners. Either there are no DLSw partners with transport connections in the proper state, no DLSw partners that support circuits for the requested SAPs, or the router could not allocate buffers for sending the CANUREACH-ex.

DLS.432

Level: C-INFO

Short Syntax: DLS.432 *intfmod*, Src SAP not open, DLC_START_DL not proc for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.432 *intfmod*, Source SAP not open, DLS_START_DL not processed for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: The router could not process a DLC_START_DL request because the source SAP specified has not been opened on any network interfaces.

DLS.433

Level: C-INFO

Short Syntax: DLS.433 DLS, CANUREACH_ex to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* lfsize *largest_frame_size*

Long Syntax: DLS.433 DLS, CANUREACH_ex to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* lfsize *largest_frame_size*

Description: DLSw successfully sent out a CANUREACH_ex SSP message over TCP or UDP to its DLSw peer.

DLS.434

Level: C-INFO

Short Syntax: DLS.434 TCP, lstn cplt from *ip_address*, closing existing cnn

Long Syntax: DLS.434 TCP, new connect request from neighbor *ip_address*, closing existing connection

Description: A remote DLSw neighbor is attempting to connect. A TCP connection already exists with this neighbor. The old TCP connection will be torn down. This happens when a remote DLSw either restarts or a remote communications problem caused the remote DLSw to close the connection, but the local DLSw TCP connection has not yet detected it.

DLS.435

Level: UE-ERROR

Short Syntax: DLS.435 TCP, no mem for TCP listen

Long Syntax: DLS.435 TCP, no memory to post a new TCP listen - will retry later

Description: There is insufficient memory to post a new TCP listen. The router will attempt to post another TCP listen in 10 seconds. During this time, the router will not accept any new TCP connections until sufficient memory becomes available.

DLS.436

Level: C-INFO

Short Syntax: DLS.436 DLS, CANUREACH_cs to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* lfsize *largest_frame_size*

Long Syntax: DLS.436 DLS, CANUREACH_cs to *ip_address* sent for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* lfsize *largest_frame_size*

Description: DLSw successfully sent out a CANUREACH_cs SSP message over TCP to its DLSw peer.

DLS.437

Level: UE-ERROR

Short Syntax: DLS.437 DLS, ICANREACH-cs rcvd with lfsize *frame_size*, less than req for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.437 DLS, ICANREACH-cs received with lfsize *frame_size*, less than requested for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: An ICANREACH-cs SSP message received from a DLSw neighbor contained an lfsize value less than what the router sent in the CANUREACH-cs. This is a DLSw protocol violation by the neighbor DLSw because the lfsize control flag was set in the CANUREACH-cs indicating that the neighbor should fail the circuit setup if it cannot establish a circuit with the lfsize the router requested in the CANUREACH-cs.

DLS.438

Level: C-INFO

Short Syntax: DLS.438 sdlc trans to sec/nego idle st for addr *link_address* nt *network ID*

Long Syntax: DLS.438 sdlc transition to secondary or negotiable idle state for address *link_address* net *network ID*

Description: The sdlc link station specified is transitioning to secondary or negotiable idle state.

DLS.439

Level: UE-ERROR

Short Syntax: DLS.439 unexp sdlc test cmd for addr *link_address* nt *network ID*

Long Syntax: DLS.439 unexpected sdlc test cmd for address *link_address* net *network ID*

Description: An unexpected test cmd frame was received from the sdlc link station specified.

DLS.440

Level: C-INFO

Short Syntax: DLS.440 nego sdlc pu 2 sta; lcl role set prim for addr *link_address* nt *network ID*

Long Syntax: DLS.440 negotiable pu 2 link being set primary for address *link_address* net *network ID*

Description: A PU 2 configured negotiable link is being overridden to a primary role.

DLS.441

Level: UE-ERROR

Short Syntax: DLS.441 unexp sdlc non-nxid recv for pu 2 sec dev addr *link_address* nt *network ID*

Long Syntax: DLS.441 unexpected sdlc non-null xid recv from primary for addr *link_address* net *network ID*

Description: An unexpected XID type was received for a secondary PU type 2 device from the primary sdlc link station specified.

DLS.442

Level: C-INFO

Short Syntax: DLS.442 conn ind rcvd from prim sdlc station addr *link_address* nt *network ID*

Long Syntax: DLS.442 connection indication received from primary sdlc station address *link_address* net *network ID*

Description: An indication that a primary SDLC link station sent a SNRM was received from the sdlc link station specified.

DLS.443

Level: UE-ERROR

Short Syntax: DLS.443 conn ind rcvd in invld state from sdlc sta to addr *link_address* nt *network ID*

Long Syntax: DLS.443 connection indication received in invalid state from sdlc station to address *link_address* net *network ID*

Description: An indication that an SDLC link station sent a SNRM in an invalid state was received from the sdlc link station specified.

DLS.444

Level: UE-ERROR

Short Syntax: DLS.444 unexp sdlc cs_confirm in state *state* for addr *link_address* nt *network ID*

Long Syntax: DLS.444 unexpected sdlc cs_confirm in state *state* for address *link_address* net *network ID*

Description: An unexpected cs_confirm event was received for the sdlc link station specified.

DLS.445

Level: UE-ERROR

Short Syntax: DLS.445 unexp xid recv from DLS for sdlc pu 2 sec addr *link_address* nt *network ID*

Long Syntax: DLS.445 unexpected sdlc xid recv from DLS for sdlc addr *link_address* net *network ID*

Description: An unexpected XID type was received

for a secondary PU type 2 device from the DLS layer.

DLS.446

Level: C-INFO

Short Syntax: DLS.446 sdlc trans to sec null_xid_pend st for addr *link_address* nt *network ID*

Long Syntax: DLS.446 sdlc transition to secondary null_xid_pend state for address *link_address* net *network ID*

Description: The secondary sdlc link station specified is transitioning to SEC_NULL_XID_PENDING state, meaning that it is awaiting a response to a NULL XID that was sent.

DLS.447

Level: C-INFO

Short Syntax: DLS.447 sdlc trans to sec contact pnd st for addr *link_address* nt *network ID*

Long Syntax: DLS.447 sdlc transition to secondary contact pending state for address *link_address* net *network ID*

Description: The secondary sdlc link station specified is transitioning to secondary contact pending state.

DLS.448

Level: UI-ERROR

Short Syntax: DLS.448 unexp rtn code from sdlc dl conn rsp = *rtn_code* for addr *link_address* nt *network ID*

Long Syntax: DLS.448 unexpected return code from sdlc dl connect response = *rtn_code* for address *link_address* net *network ID*

Description: An attempt to send an sdlc connect response returned an unexpected return code from the DL.

DLS.449

Level: UI-ERROR

Short Syntax: DLS.449 unexp sdlc contacted_rcv in state *state* for addr *link_address* nt *network ID*

Long Syntax: DLS.449 unexpected sdlc contacted_rcv in state *state* for address *link_address* net *network ID*

Description: An unexpected contacted_rcv event was received for the sdlc link station specified.

DLS.450

Level: U-INFO

Short Syntax: DLS.450 SDLC, link role secondary, start_dl not honored, for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.450 SDLC, link role secondary, start_dl not honored, for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: A Start DL request from DLS to the SDLC interface module was not honored because the link role for the SDLC station was Secondary. This is not an error. This is due to the fact that SDLC is an unbalanced protocol that requires a secondary station to be polled by a primary before it can respond or initiate some action.

DLS.452

Level: C-INFO

Short Syntax: DLS.452 nego sdlc pu 5 sta; lcl role set prim for addr *link_address* nt *network ID*

Long Syntax: DLS.452 negotiable pu 5 link being set secondary for address *link_address* net *network ID*

Description: A PU 5 configured negotiable link is being overridden to a secondary role.

DLS.454

Level: UE-ERROR

Short Syntax: DLS.454 DLSw disabled no mem for appnim struct

Long Syntax: DLS.454 DLSw forwarder disabled no memory for appnim structures

Description: The Data Link Switching forwarder has been disabled because there was not enough memory to create necessary appnim data structures.

DLS.455

Level: C-INFO

Short Syntax: DLS.455 DLSw, Created a new transport record for neighbor *ip_address*

Long Syntax: DLS.455 DLSw, A new Transport record was created successfully for a previously unknown Neighbor at IP address *ip_address*

Description: DLSw allocated a new control block for a neighbor that was previously unknown. This neighbor was learned either from group exploration or from console definition.

DLS.457

Level: C-INFO

Short Syntax: DLS.457 DLSw, Deleted transport record for Neighbor *ip_address*

Long Syntax: DLS.457 DLSw, A Transport record was deleted successfully for a Neighbor at IP address *ip_address*

Description: DLSw deleted a tran_man entry for a

Neighbor which was previously known. It will have deleted due to a lost TCP connection for a Dynamic Neighbor or specific deletion at the console.

DLS.458

Level: C-INFO

Short Syntax: DLS.458 TCP connection to Neighbor *ip_address* has closed!

Long Syntax: DLS.458 The TCP connection to the Neighbor at IP address *ip_address* has closed

Description: DLSw has had a TCP connection to a Neighbor close. This will either be due to the foreign host dropping the connection or a lost connection. It may also be due to the expiration of the Inactivity Neighbor Termination Timer when a TCP connection has become Idle.

DLS.459

Level: C-INFO

Short Syntax: DLS.459 DLSw, Adding a Dynamic transport record for Neighbor *ip_address*

Long Syntax: DLS.459 DLSw, A new Dynamic Transport record was added successfully for a previously unknown Neighbor at IP address *ip_address*

Description: DLSw allocated a new Dynamic Transport entry for a Neighbor which was previously unknown. It will have been learned from a TCP connection initiation when Dynamic Neighbors is Enabled. It will be configured with the Dynamic Neighbor TCP parameters.

DLS.460

Level: C-INFO

Short Syntax: DLS.460 Sent Unicast Capex Request to *ip_address*

Long Syntax: DLS.460 A Unicast Capabilities Exchange Request has been sent to the DLSw neighbor at *ip_address*

Description: The DLSw UDP Interface module has sent a Unicast Capabilities Exchange Request message to the DLSw neighbor. This is the result of a desire to exchange information with the neighbor without establishing a TCP connection. Passive Neighbors and Passive Group members will send this message.

DLS.461

Level: C-INFO

Short Syntax: DLS.461 Received Unicast Capex Request from *ip_address*

Long Syntax: DLS.461 A Unicast Capabilities Exchange Request has been received from the DLSw neighbor at *ip_address*

Description: The DLSw UDP Interface module received a Unicast Capabilities Exchange message from a DLSw neighbor. The parsing module determined that the message type is a Request.

DLS.462

Level: C-INFO

Short Syntax: DLS.462 Sent Unicast Capex Response to *ip_address*

Long Syntax: DLS.462 A Unicast Capabilities Exchange Response has been sent to the DLSw neighbor at *ip_address*

Description: The DLSw UDP Interface module has sent a Unicast Capabilities Exchange Response message to a DLSw neighbor. This is in response to a Unicast Capabilities Exchange Request.

DLS.463

Level: C-INFO

Short Syntax: DLS.463 Received Unicast Capex Response from *ip_address*

Long Syntax: DLS.463 A Unicast Capabilities Exchange Response has been received from the DLSw neighbor at *ip_address*

Description: The DLSw UDP Interface module received a Unicast Capabilities Exchange message from a DLSw neighbor. The parsing module determined that the message type is a Response.

DLS.464

Level: C-INFO

Short Syntax: DLS.464 ICANREACH-ex rcvd *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* lfsize *largest_frame_size*

Long Syntax: DLS.464 ICANREACH-ex received for *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap* lfsize *largest_frame_size*

Description: DLS has received a ICANREACH-ex for the specified circuit.

DLS.465

Level: UE-ERROR

Short Syntax: DLS.465 DLSw disabled no mem for group struct

Long Syntax: DLS.465 DLSw forwarder disabled no memory for group protocol structures

Description: The Data Link Switching forwarder has been disabled because there was not enough memory

to create necessary group protocol data structures.

DLS.466

Level: C-INFO

Short Syntax: DLS.466 DLS, udpim did not create transport control block for *ip_address*

Long Syntax: DLS.466 DLS, udpim did not create transport control block for *ip_address*

Description: DLS udpim module did not create a transport control block for the incoming ip address. This could be due to a lack of memory or dynamic neighbors being disabled.

DLS.467

Level: C-INFO

Short Syntax: DLS.467 DLS, udpim received an invalid unicast/multicast packet from *ip_address*

Long Syntax: DLS.467 DLS, udpim received an invalid unicast/multicast packet from *ip_address*

Description: DLS udpim module receive an invalid unicast or multicast packet from the incoming ip address. The packet was discarded.

DLS.468

Level: C-INFO

Short Syntax: DLS.468 DLS, udpim attempted to send an invalid unicast/multicast packet

Long Syntax: DLS.468 DLS, udpim attempted to send an invalid unicast/multicast packet

Description: DLS udpim module attempted to send an invalid unicast or multicast packet. The packet was discarded.

DLS.469

Level: UI-ERROR

Short Syntax: DLS.469 QLLC config error: *error_msg*

Long Syntax: DLS.469 QLLC configuration error: *error_msg*

Description: There is an error in DLSw QLLC configuration information, as indicated. DLSw has skipped any invalid information and continued. A user may correct the configuration and restart the router to clear this problem.

DLS.470

Level: UI-ERROR

Short Syntax: DLS.470 QLLC init error: *error_msg*

Long Syntax: DLS.470 QLLC initialization error: *error_msg*

Description: There is an error initializing DLSw QLLC support, as indicated. These are serious errors that usually require software service to correct.

DLS.471

Level: C-INFO

Short Syntax: DLS.471 QLLC, *event_id* for intf *interface result_msg*

Long Syntax: DLS.471 QLLC, *event_id* for interface *interface result_msg*

Description: An interface-level event occurred for the specified interface. In general, these are normal events that link and unlink DLSw to the underlying QLLC and X.25 protocol layers.

DLS.472

Level: C-INFO

Short Syntax: DLS.472 QLLC, *event_id* for *station_id* in state *state more_info*

Long Syntax: DLS.472 QLLC, *event_id* for *station_id* in state *state more_info*

Description: DLS passed the specified request to its QLLC interface module. It usually does this in response to a received SSP message from a DLSw partner, or in response to a DLS timer event.

DLS.473

Level: C-INFO

Short Syntax: DLS.473 QLLC, *event_id* for *station_id* in state *state more_info*

Long Syntax: DLS.473 QLLC, *event_id* for *station_id* in state *state more_info*

Description: QLLC passed the specified Indicate or Confirm to DLSw. Indicates are notifications of asynchronous events (usually the arrival of a packet to QLLC), and Confirms report the delayed success or failure of Requests that DLSw previously issued to QLLC.

DLS.474

Level: C-INFO

Short Syntax: DLS.474 QLLC, *event_id* for *station_id* in state *state more_info*

Long Syntax: DLS.474 QLLC, *event_id* for *station_id* in state *state more_info*

Description: DLSw passed the specified Request or Response to QLLC. Requests are commands asking for a service from QLLC, and Responses are DLSw's

answer to an Indicate that QLLC previously gave to DLSw. Note that Requests are normally logged following the return of control to DLSw after issuing the command, so that the return code from QLLC can be included in the ELS message.

DLS.475

Level: UE-ERROR

Short Syntax: DLS.475 QLLC, *event_id* for *station_id* in unexp state *state more_info*

Long Syntax: DLS.475 QLLC, *event_id* for *station_id* in unexpected state *state more_info*

Description: The DLSw QLLC interface module has received an event notification from DLS or from QLLC in an unexpected state. This does not always interfere with successful operation, but if it does, contact service.

DLS.476

Level: C-INFO

Short Syntax: DLS.476 QLLC, Call Ind from net *interface* dte addr *dte_address*

Long Syntax: DLS.476 QLLC, Call Indicate from net *interface* dte address *dte_address*

Description: QLLC has received a Call Request packet from the network and is giving DLSw an opportunity to accept the call. This message should be followed by others indicating how DLSw responded.

DLS.477

Level: C-INFO

Short Syntax: DLS.477 QLLC, Call from net *interface* dte *dte_address* refused: *reason*

Long Syntax: DLS.477 QLLC, Call from net *interface* dte *dte_address* refused: *reason*

Description: QLLC is refusing an incoming call for the reason indicated. If DLSw is the intended recipient of the call, this may indicate a user configuration error. If some other QLLC user (e.g., APPN) is the intended recipient of the call, it is normal for DLSw to refuse the call.

DLS.478

Level: C-INFO

Short Syntax: DLS.478 QLLC, Call accept pend for net *interface* dte *dte_address*, *call_type*

Long Syntax: DLS.478 QLLC, Call accept pending for net *interface* dte *dte_address*, *call_type*

Description: QLLC is taking ownership of an incoming call, and is beginning to contact remote DLSw partners to search for the associated destination resource. If this search is successful, DLSw will later accept the call completely. The call_type parameter indicates whether this call is from a QLLC device configured to DLSw, or is dynamic.

DLS.479

Level: UE-ERROR

Short Syntax: DLS.479 QLLC, role conflict for *station_id*: *reason*

Long Syntax: DLS.479 QLLC, role conflict for *station_id*: *reason*

Description: An event has occurred indicating a QLLC link station role (primary or secondary) that is inconsistent with configured or previous discovered information. The exact conflict is described in the ″reason″ part of this message.

DLS.480

Level: C-INFO

Short Syntax: DLS.480 QLLC, *event_id* for *station_id* in state *state more_info*

Long Syntax: DLS.480 QLLC, *event_id* for *station_id* in state *state more_info*

Description: An internal event has occurred that is not covered by the description of other station-level messages. This is a normal event, and is described by the ″event_id″ part of this message.

DLS.481

Level: UI-ERROR

Short Syntax: DLS.481 QLLC, *event_id* for *station_id* in state *state more_info*

Long Syntax: DLS.481 QLLC, *event_id* for *station_id* in state *state more_info*

Description: An error event has occurred that is not covered by the description of other station-level messages. These are unusual events that may result in circuit establishment failure and need to be reported to service.

DLS.482

Level: UE-ERROR

Short Syntax: DLS.482 QLLC, no dest MAC/SAP defined for *station_id*, search aborted

Long Syntax: DLS.482 QLLC, no destination MAC/SAP defined for *station_id*, search aborted

Description: An event has occurred that normally would have caused DLSw to explore for and initiate a connection to the destination MAC/SAP defined for the QLLC station. Because the user has not configured a

destination MAC/SAP pair, it is not possible to explore for that destination. The event is ignored.

DLS.483

Level: C-INFO

Short Syntax: DLS.483 QLLC, *event_id* rcvd for *source_mac_addr*-> *dest_mac_addr*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.483 QLLC, *event_id* received for *source_mac_addr*-> *dest_mac_addr*, sap *source_sap*-> *dest_sap*

Description: DLS passed the specified event to its QLLC interface module.

DLS.484

Level: C-INFO

Short Syntax: DLS.484 DLC, *station_id* mtu reduced *old_mtu* to *new_mtu* per rcvd XID_ *xid_format*

Long Syntax: DLS.484 DLC, station *station_id* mtu reduced from *old_mtu* to *new_mtu* per received XID_ *xid_format*

Description: DLSw has received a SNA XID from the specified station indicating that it cannot handle receiving frames of the configured MTU size. DLSw has therefore reduced the effective MTU size for this station. This message is common to the DLCs that DLSw supports; the station id indicates the DLC type involved.

DLS.485

Level: UI-ERROR

Short Syntax: DLS.485 QLLC, *station_id* automatically disabled by Register Req failure

Long Syntax: DLS.485 QLLC, *station_id* automatically disabled by Register Req failure

Description: DLSw has attempted to register a configured PVC with the X.25 stack, and X.25 has rejected this request. DLSw automatically disables its PVC definition so that this failure will not repeat forever. The probable causes of this problem are that the PVC is not configured in X.25, or it is configured but for a protocol other than DLS. The user should fix X.25 configuration and restart the router. The disabling of the PVC in DLSw will not survive the restart.

DLS.486

Level: C-INFO

Short Syntax: DLS.486 QLLC, XID FSM for *station_id*: *event_id* in xid state *old_state*, role *role*; action= *action*, new state= *new_state*

Long Syntax: DLS.486 QLLC, XID FSM for *station_id*: *event_id* in xid state *old_state*, role *role*; action= *action*, new state= *new_state*

Description: The DLSw QLLC interface maintains an XID state machine to control XIDs flowing to and from DLS and the QLLC device. This message indicates that the FSM was called, and shows its inputs and outputs.

DLS.487

Level: C-INFO

Short Syntax: DLS.487 LLC, *frame_type* frame sent, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.487 LLC, *frame_type* frame sent, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: A TEST or XID frame was successfully sent to the Channel.

DLS.488

Level: UI-ERROR

Short Syntax: DLS.488 No mem to create LLC address map for Channel mac address *dest_mac_address*

Long Syntax: DLS.488 No mem to create LLC address map for Channel mac address *dest_mac_address*

Description: Due to a memory shortage condition, LLCIM could not create an address map entry for the specifed mac address. DLSw cannot forward traffic to the Channel assigned this mac address.

DLS.489

Level: C-INFO

Short Syntax: DLS.489 LLC, *frame_type* frame send failed, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Long Syntax: DLS.489 LLC, *frame_type* frame send failed, *source_mac_address*-> *dest_mac_address*, sap *source_sap*-> *dest_sap*

Description: LLC was unsuccessful sending a TEST or XID frame to the Channel. This condition could be a result of incorrect DLSw and/or Channel configuration.

DLS.490

Level: UI-ERROR

Short Syntax: DLS.490 DLS Dropping an unsupported SSP version packet received from *ip_addr*!

Long Syntax: DLS.490 DLS Dropping an unsupported SSP version packet received from *ip_addr*!

Description: An unsupported SSP version packet was received from a DLSw neighbor and dropped.

DLS.491

Level: UE-ERROR

Short Syntax: DLS.491 DLSw disabled no mem for MAC list struct

Long Syntax: DLS.491 DLSw forwarder disabled no memory for MAC list structures

Description: The Data Link Switching forwarder has been disabled because there was not enough memory to create necessary MAC list data structures.

DLS.492

Level: UI-ERROR

Short Syntax: DLS.492 DLS, FAILED to send DATAFRAME to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.492 DLS, FAILED to send DATAFRAME to *ip_address* for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: A DATAFRAME SSP control message was not sent because either there are no buffers or the DLSw partner does not support the source sap in its DLSw capabilities exchange SAP list.

DLS.493

Level: C-INFO

Short Syntax: DLS.493 DLS, Broadcast DATAFRAME sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.493 DLS, Broadcast DATAFRAME sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: While processing UI-frame for a given destination, DLS sent out broadcast DATAFRAME via multicast UDP.

DLS.494

Level: C-INFO

Short Syntax: DLS.494 DLS, Broadcast DATAFRAME unsuccessful for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.494 DLS, Broadcast DATAFRAME unsuccessful for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: The router could not send DATAFRAME to any DLSw partners. Either there are no DLSw partners with transport connections in the proper state, no DLSw partners that support circuits for the requested SAPs, or the router could not allocate buffers for sending the DATAFRAME.

DLS.495

Level: C-INFO

Short Syntax: DLS.495 DLS, Broadcast NETBIOS_NQ_ex sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap* lfsize *largest_frame_size*

Long Syntax: DLS.495 DLS, Broadcast NETBIOS_NQ_ex sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap* lfsize *largest_frame_size*

Description: While processing UI-frame for a given destination, DLS sent out broadcast NETBIOS_NQ_ex via multicast UDP.

DLS.496

Level: C-INFO

Short Syntax: DLS.496 DLS, Broadcast NETBIOS_ANQ sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Long Syntax: DLS.496 DLS, Broadcast NETBIOS_ANQ sent for *source_mac_address-> dest_mac_address*, sap *source_sap-> dest_sap*

Description: While processing UI-frame for a given destination, DLS sent out broadcast NETBIOS_ANQ via multicast UDP.

DLS.497

Level: UE-ERROR

Short Syntax: DLS.497 DLSw disabled no mem for circuit priority overrides

Long Syntax: DLS.497 DLSw forwarder disabled no memory for circuit priority overrides

Description: The Data Link Switching forwarder has been disabled because there was not enough memory to create necessary circuit priority override structures.

DLS.498

Level: UI-ERROR

Short Syntax: DLS.498 DLS, SSP msg rcvd from *ip_address*, msg_length too large, frame dropped

Long Syntax: DLS.498 DLS, SSP message received from *ip_address* has a msg_length greater than the largest packet size which can be processed. It has been dropped.

Description: DLS forwarder received a Switch to Switch Protocol message over TCP that had a message length greater than the largest packet size which can be processed. The frame has been dropped. The system-wide setting of PACKET-SIZE on the specified Neighbor should be reviewed.

DLS.499

Level: UI-ERROR

Short Syntax: DLS.499 DLSw SDLC link non-switched, SDLC link nt *network ID* switched

Long Syntax: DLS.499 DLSw SDLC link is configured at non-switched, but SDLC link net *network ID* is configured as switched

Description: DLSw detected an invalid configuration condition. A DLSw SDLC link station has been configured as non-switched. For the same interface, SDLC was configued as switched. Fix the configuration and restart the router.

DLS.500

Level: UI-ERROR

Short Syntax: DLS.500 DLSw SDLC link switched, SDLC link nt *network ID* non-switched

Long Syntax: DLS.500 DLSw SDLC link is configured at switched, but SDLC link net *network ID* is configured as non-switched

Description: DLSw detected an invalid configuration condition. A DLSw SDLC link station has been configured as switched. For the same interface, SDLC was configued as non-switched. Fix the configuration and restart the router.

DLS.501

Level: UI-ERROR

Short Syntax: DLS.501 Invalid interface number *interface* in Interface/SAP List record

Long Syntax: DLS.501 Invalid interface number *interface* in Interface/SAP List record

Description: DLSw detected an invalid configuration parameter at start-up time. The interface on which DLSw is to open a list of LLC SAPs does not exist on the router. You should change the configuration and restart the router.

Chapter 28. Default Gateways (DGW)

This chapter describes Default Gateways (DGW) messages. For information on message content and how to use the message, refer to the Introduction.

DGW.001

Level: C-INFO

Short Syntax: DGW.001 cfg ent fnd on nt *net_no*

Long Syntax: DGW.001 found a configuration entry for a gateway on net *net_no*

Description: This message is generated when an interface comes up and a gateway is configured on the interface

DGW.002

Level: C-INFO

Short Syntax: DGW.002 dgw *gw_ip_address* crtd on nt *net_no*

Long Syntax: DGW.002 created gateway *gw_ip_address* on net *net_no*

Description: This message is generated when an interface comes up an successfully creates a default gateway

DGW.003

Level: C-INFO

Short Syntax: DGW.003 sent arp rply on nt *net_no* for *gw_ip_address*

Long Syntax: DGW.003 sent an automatic arp reply for a gateway on net *net_no* for *gw_ip_address*

Description: This message is generated when an ARP reply is automatically sent by the gateway code. This reply is sent so bridge ports can learn the source of the gateway.

DGW.004

Level: C-INFO

Short Syntax: DGW.004 activated gw *gw_ip_address* on nt *net_no*

Long Syntax: DGW.004 activated gateway *gw_ip_address* on net *net_no*

Description: This message is generated when a gateway is activated on an interface. The interface is now able to receive packets destined for the gateway's MAC and IP addresses.

DGW.005

Level: C-INFO

Short Syntax: DGW.005 de-activated gw *net_no* on nt

Long Syntax: DGW.005 de-activated gateway *net_no* on net

Description: This message is generated when a gateway is de-activated on an interface. The interface is now unable to receive packets destined for the gateway's MAC and IP addresses.

DGW.006

Level: C-INFO

Short Syntax: DGW.006 de-activated all gw on nt *net_no*

Long Syntax: DGW.006 de-activated all gateway on net *net_no*

Description: This message is generated when all gateways are de-activated on an interface. The interface is now unable to receive packets destined for any gateway's MAC and IP addresses on the interface.

DGW.007

Level: C-INFO

Short Syntax: DGW.007 prm gw MAC query on nt *net_no*

Long Syntax: DGW.007 a mac address was found to be a primary gateway on net *net_no*

Description: This message is generated when another subsystem queries the gateway to determine is a MAC address is that of a primary gateway.

DGW.008

Level: C-INFO

Short Syntax: DGW.008 bk gw MAC query on nt *net_no*

Long Syntax: DGW.008 a mac address was found to be a backup gateway on net *net_no*

Description: This message is generated when another subsystem queries the gateway to determine is a MAC address is that of a backup gateway.

DGW.009

Level: C-INFO

Short Syntax: DGW.009 gw IP *gw_ip_address* query on nt *net_no*

Long Syntax: DGW.009 an IP address *gw_ip_address* was found to be a gateway on net *net_no*

Description: This message is generated when another subsystem queries the gateway to determine is an IP address is that of a gateway.

DGW.010

Level: C-INFO

Short Syntax: DGW.010 max gws on nt *net_no* reached

Long Syntax: DGW.010 greater than maximum number of gateways configured on net *net_no*

Description: More than the Maximum number of Redundant Default Gateways has been configured on the specified LEC interface

Chapter 29. DHCP Services

This chapter describes DHCP Services messages. For information on message content and how to use the message, refer to the Introduction.

DHCP.001

Level: P-TRACE

Short Syntax: DHCP.001 Sent DHCP packet to server *destination* giaddr *giaddr* haddr *haddr*

Long Syntax: DHCP.001 Sent DHCP packet to server *destination* giaddr *giaddr* haddr *haddr*

Description: Proxy DHCP sent a packet

DHCP.002

Level: CI-ERROR

Short Syntax: DHCP.002 Error Sending Dhcp Packet: Bad Dest Address *destination*

Long Syntax: DHCP.002 Error Sending Dhcp Packet: Bad Dest Address *destination*

Description: An error occurred sending the DHCP packet. This will occur if there currently is no route to one of the DHCP servers configured.

DHCP.003

Level: C-TRACE

Short Syntax: DHCP.003 Option: tag= *tag* len= *len*

Long Syntax: DHCP.003 Option: tag= *tag* len= *len*

Description: Processed a DHCP option, tag and length are reported here.

DHCP.004

Level: UI-ERROR

Short Syntax: DHCP.004 removed

Long Syntax: DHCP.004 removed

Description: none

DHCP.005

Level: C-TRACE

Short Syntax: DHCP.005 Proxy DHCP Closing on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.005 Proxy DHCP Closing on network *network ID* clientid *clientid* state *state*

Description: Should occur when IPCP closes normally or is other wise halted by the line being disconnected or terminated for any reason.

DHCP.006

Level: C-TRACE

Short Syntax: DHCP.006 Proxy DHCP state transition from *oldstate* to *newstate* on net *network ID* cid *clientid*

Long Syntax: DHCP.006 Proxy DHCP state transition from *oldstate* to *newstate* on network *network ID* clientid *clientid*

Description: A Proxy DHCP state transition occured (states are defined in RFC 2131)

DHCP.007

Level: C-TRACE

Short Syntax: DHCP.007 Proxy DHCP Reset on net *network ID* cid *clientid*

Long Syntax: DHCP.007 Proxy DHCP Reset on network *network ID* clientid *clientid*

Description: Proxy DHCP Reset of state machine occurred

DHCP.008

Level: C-TRACE

Short Syntax: DHCP.008 Proxy DHCP IP Address Retry on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.008 Proxy DHCP IP Address Retry on network *network ID* clientid *clientid* state *state*

Description: Retry for Proxy DHCP Get IP address. This occurs if no response is received from a DHCP server for a specific amount of time

DHCP.009

Level: C-TRACE

Short Syntax: DHCP.009 Initiate Proxy DHCP Get IP Address on network *network ID* state *state*

Long Syntax: DHCP.009 Initiate Proxy DHCP Get IP Address on network *network ID* state *state*

Description: Initial Proxy DHCP Get IP address

DHCP.010

Level: UE-ERROR

Short Syntax: DHCP.010 Could not find DHCP Option *option*

Long Syntax: DHCP.010 Could not find DHCP Option *option*

Description: Searched for DHCP Option and could not find it in a received packet. This may occur if the server sends us an option we do not understand, or if the packet is corrupted. See also event DHCP_25

DHCP.011

Level: C-TRACE

Short Syntax: DHCP.011 Processing DHCP NAK on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.011 Processing DHCP NAK on network *network ID* clientid *clientid* state *state*

Description: Processing DHCP NAK

DHCP.012

Level: C-TRACE

Short Syntax: DHCP.012 Processing DHCP ACK on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.012 Processing DHCP ACK on network *network ID* clientid *clientid* state *state*

Description: Processing DHCP ACK - this is the final message we will receive from the DHCP server before we move to the BOUND state.

DHCP.013

Level: UI-ERROR

Short Syntax: DHCP.013 ERROR: *desc* on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.013 ERROR: *desc* on network *network ID* clientid *clientid* state *state*

Description: General Error

DHCP.014

Level: UI-ERROR

Short Syntax: DHCP.014 WARNING: *desc* on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.014 WARNING: *desc* on network *network ID* clientid *clientid* state *state*

Description: General Warning

DHCP.015

Level: C-TRACE

Short Syntax: DHCP.015 Option DHCP_MESSAGE_TYPE = *message_type* (*message_text*) on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.015 Option DHCP_MESSAGE_TYPE = *message_type* (*message_text*) on network *network ID* clientid *clientid* state *state*

Description: Processed DHCP option of this type

DHCP.016

Level: C-TRACE

Short Syntax: DHCP.016 Option DHCP_REQUESTED_IP = *ipaddr* on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.016 Option DHCP_REQUESTED_IP = *ipaddr* on network *network ID* clientid *clientid* state *state*

Description: Processed DHCP option of this type

DHCP.017

Level: C-TRACE

Short Syntax: DHCP.017 Option DHCP_LEASE_TIME = *time* on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.017 Option DHCP_LEASE_TIME = *time* on network *network ID* clientid *clientid* state *state*

Description: Processed DHCP option of this type

DHCP.018

Level: C-TRACE

Short Syntax: DHCP.018 Option DHCP_HOSTNAME on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.018 Option DHCP_HOSTNAME on network *network ID* clientid *clientid* state *state*

Description: Currently recognized but not supported as there is no way to transmit this information to the client. Note that this is NOT Dynamic DNS, where the Proxy DHCP client actually sends the HOSTNAME to the DHCP server.

DHCP.019

Level: C-TRACE

Short Syntax: DHCP.019 Option DHCP_DOMAINNAME on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.019 Option DHCP_DOMAINNAME on network *network ID* clientid *clientid* state *state*

Description: Currently recognized but not supported, again this information cannot be sent to the client over IPCP.

DHCP.020

Level: C-TRACE

Short Syntax: DHCP.020 Option DHCP_SERVER_ID = *server* on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.020 Option DHCP_SERVER_ID = *server* on network *network ID* clientid *clientid* state *state*

Description: Option of this type received

DHCP.021

Level: C-TRACE

Short Syntax: DHCP.021 MESSAGE FROM DHCP SERVER: (len = *length*) *message* on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.021 MESSAGE FROM DHCP SERVER: (len = *length*) *message* on network *network ID* clientid *clientid* state *state*

Description: DHCP server sent us a message. This should be a human-readable ASCII text string.

DHCP.022

Level: C-TRACE

Short Syntax: DHCP.022 Option DHCP_RENEWAL_TIME = *time* on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.022 Option DHCP_RENEWAL_TIME = *time* on network *network ID* clientid *clientid* state *state*

Description: Processed DHCP option of this type

DHCP.023

Level: C-TRACE

Short Syntax: DHCP.023 Option DHCP_REBIND_TIME = *time* on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.023 Option DHCP_REBIND_TIME = *time* on network *network ID* clientid *clientid* state *state*

Description: Processed DHCP option of this type

DHCP.024

Level: C-TRACE

Short Syntax: DHCP.024 Option DHCP_CLIENT_ID = *clientid* on network *network ID* state *state*

Long Syntax: DHCP.024 Option DHCP_CLIENT_ID = *clientid* on network *network ID* state *state*

Description: Processed DHCP option of this type

DHCP.025

Level: CE-ERROR

Short Syntax: DHCP.025 Unknown option type *option* on net *network ID* state *state*

Long Syntax: DHCP.025 Unknown option type *option* on network *network ID* state *state*

Description: Unknown option received. This occurs when the DHCP server sends us an option we do not recognize. We will simply ignore the option (we most likely cannot cannot utilize it for Proxy DHCP). If the client needs additional options, it should issue a DHCPINFORM after the IP link is established to obtain them.

DHCP.026

Level: P-TRACE

Short Syntax: DHCP.026 Processing DHCP OFFER on net *network ID* clientid *clientid* state *state*

Long Syntax: DHCP.026 Processing DHCP OFFER on network *network ID* clientid *clientid* state *state*

Description: Processing DHCP OFFER received from server.

DHCP.027

Level: P-TRACE

Short Syntax: DHCP.027 Received DHCP PACKET on net *network ID* state *state*

Long Syntax: DHCP.027 Received DHCP PACKET on network *network ID* state *state*

Description: Received a DHCP PACKET from server destined for Proxy DHCP

DHCP.028

Level: P-TRACE

Short Syntax: DHCP.028 DHCP Release Sent on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.028 DHCP Release Sent on network *network ID* clientid *clientid* state *state*

Description: Sent DHCP Release to free lease. This should occur when an IP connection is terminated that had utilized an IP address from a DHCP server.

DHCP.029

Level: P-TRACE

Short Syntax: DHCP.029 DHCP Decline Sent on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.029 DHCP Decline Sent on network *network ID* clientid *clientid* state *state*

Description: Sent DHCP Decline. This should happen if for some reason we do not like the parameters offered to us by the DHCP server.

DHCP.030

Level: P-TRACE

Short Syntax: DHCP.030 DHCP Discover Sent on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.030 DHCP Discover Sent on network *network ID* clientid *clientid* state *state*

Description: Sent DHCP Discover. This is the first message sent. We should send one for each dhcp server configured.

DHCP.031

Level: P-TRACE

Short Syntax: DHCP.031 DHCP Request Sent on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.031 DHCP Request on network *network ID* clientid *clientid* state *state*

Description: Sent DHCP Request. We send this in reponse to a DHCP offer from the DHCP server.

DHCP.032

Level: P-TRACE

Short Syntax: DHCP.032 DHCP Request Retry on net *network ID* cid *clientid* state *state*

Long Syntax: DHCP.032 DHCP Request Retry on network *network ID* clientid *clientid* state *state*

Description: Sent DHCP Request retry. This occurs after a specific amount of time if we have not received a response from our DHCP server.

DHCP.033

Level: P-TRACE

Short Syntax: DHCP.033 Received DHCP Packet: claddr= *&bpkt->btp_claddr.i_lwrd* yraddr= *&bpkt->btp_yraddr.i_lwrd* svaddr= *&bpkt->btp_svaddr.i_lwrd* gwaddr= *&bpkt->btp_gwaddr.i_lwrd*

Long Syntax: DHCP.033 claddr= *&bpkt->btp_claddr.i_lwrd* yraddr= *&bpkt->btp_yraddr.i_lwrd* svaddr= *&bpkt->btp_svaddr.i_lwrd* gwaddr= *&bpkt->btp_gwaddr.i_lwrd*

Description: Received a DHCP packet.

DHCP.034

Level: C-TRACE

Short Syntax: DHCP.034 Option DHCP_CLIENT_FQDN = *clientid* on network *network ID* state *state*

Long Syntax: DHCP.034 Option DHCP_CLIENT_FQDN = *clientid* on network *network ID* state *state*

Description: Processed DHCP option of this type

DHCP.035

Level: P-TRACE

Short Syntax: DHCP.035 DHCP Renewal Request *network ID* int *minutes/ seconds* cid *clientid* state *state*

Long Syntax: DHCP.035 DHCP Renewal Request *network ID* interface *minutes/ seconds* clientid *clientid* state *state*

Description: Sent DHCP Renewal Request

DHCP.036

Level: P-TRACE

Short Syntax: DHCP.036 DHCP Rebind Request *network ID* int *minutes/ seconds* cid *clientid* state *state*

Long Syntax: DHCP.036 DHCP Rebind Request *network ID* interface *minutes/ seconds* clientid *clientid* state *state*

Description: Sent DHCP Rebind Request

DHCP.037

Level: UI-ERROR

Short Syntax: DHCP.037 Received DHCP Packet on network *network ID* while DHCP Not Enabled!!

Long Syntax: DHCP.037 Received DHCP Packet on network *network ID* while DHCP Not Enabled!!

Description: Received DHCP Packet while DHCP is not enabled.

DHCP.038

Level: P-TRACE

Short Syntax: DHCP.038 DHCP Request Denial Notification Sent from network *network ID* clientid *clientid* state *state*

Long Syntax: DHCP.038 DHCP Request Denial

Notification Sent from network *network ID* clientid *clientid* state *state*

Description: Sent DHCP Request Denial Notification. This is sent to servers who offered an address, after we already chose a different server.

DHCP.039

Level: CE-ERROR

Short Syntax: DHCP.039 Unable to contact DHCP server with successive retries, giving up on network *network ID*

Long Syntax: DHCP.039 Unable to contact DHCP server with successive retries, giving up on network *network ID*

Description: Giving up Proxy DHCP. IPCP probably timed out before we got here anyway.

DHCP.040

Level: CE-ERROR

Short Syntax: DHCP.040 DHCP server offered address not equal to current address, closing IPCP on *network ID*

Long Syntax: DHCP.040 DHCP server offered address not equal to current address, closing IPCP on *network ID*

Description: Received a different address after Rebinding. We cannot handle this, so we close IPCP.

DHCP.041

Level: UI-ERROR

Short Syntax: DHCP.041 ERROR: *desc*

Long Syntax: DHCP.041 ERROR: *desc*

Description: General Error - no interface information available

---

DHCP.042

Level: UI-ERROR

Short Syntax: DHCP.042 WARNING: *desc*

Long Syntax: DHCP.042 WARNING: *desc*

Description: General Warning - no interface information available

---

DHCP.043

Level: C-INFO

Short Syntax: DHCP.043 DHCP server has been reset by user

Long Syntax: DHCP.043 DHCP server has been reset by user

Description: User invoked command DHCP reset to initialize DHCP server

---

DHCP.044

Level: UI-ERROR

Short Syntax: DHCP.044 Failed to initialize server, DHCP is not active

Long Syntax: DHCP.044 Failed to initialize server, DHCP is not active

Description: Initialization process of DHCP failed in profile init or address mapper, therefore DHCP is not active

---

DHCP.045

Level: C-INFO

Short Syntax: DHCP.045 Cannot initialize profile repository

Long Syntax: DHCP.045 Cannot initialize profile repository

Description: Initialization process of DHCP fails in initializing address configuration fails

---

DHCP.046

Level: C-INFO

Short Syntax: DHCP.046 Produced zero-length opt of type *type*

Long Syntax: DHCP.046 Produced zero-length option of type *type*

Description: Conversion routine for option has Produced zero-length option of type %S

DHCP.047

Level: C-INFO

Short Syntax: DHCP.047 Decoupling a client which has no coupled address

Long Syntax: DHCP.047 Decoupling a client which has no coupled address

Description: Attempt to Decouple a client which has no coupled address

---

DHCP.048

Level: C-INFO

Short Syntax: DHCP.048 Decoupling an address which has no coupled client

Long Syntax: DHCP.048 Decoupling an address which has no coupled client

Description: Attempt to decouple an address which has no association to a client

---

DHCP.049

Level: UI-ERROR

Short Syntax: DHCP.049 index *index* to *type_of_record* is too large

Long Syntax: DHCP.049 index *index* to *type_of_record* is too large

Description: Index to address or client record is out of bound

---

DHCP.050

Level: UE-ERROR

Short Syntax: DHCP.050 Request packet is too short. Size *size_of_packet*

Long Syntax: DHCP.050 Request packet is too short. Size *size_of_packet*

Description: The size of the request packet is shorter than the minimum size

---

DHCP.051

Level: UE-ERROR

Short Syntax: DHCP.051 Option field is too short. Size *size_of_option_field*

Long Syntax: DHCP.051 Option field is too short. Size *size_of_option_field*

Description: Size of option field is shorter than the mininum bootp vendor field size

---

DHCP.052

Level: C-INFO

Short Syntax: DHCP.052 Validity of network location undetermined--reply none

Long Syntax: DHCP.052 Validity of network location undetermined--reply none

Description: Query for network address comes back with no information, since we do not know the validity of this address, we do not send out the reply

---

DHCP.053

Level: UE-ERROR

Short Syntax: DHCP.053 Clnt is configured for hostname *configured_hostname* on server but requested hostname *requested_hostname*

Long Syntax: DHCP.053 Client is configured for hostname *configured_hostname* on server but requested hostname *requested_hostname*

Description: Client hostname check fails. Configured hostname on the server is different form Requested hostname from client (option 12)

DHCP.054

Level: UE-ERROR

Short Syntax: DHCP.054 Unknown opcode *opcode* in header of dhcp packet

Long Syntax: DHCP.054 Unknown opcode *opcode* in the header of dhcp packet

Description: Received a dhcp packet with unrecognized opcode in the header

---

DHCP.055

Level: C-INFO

Short Syntax: DHCP.055 Decoupling address mapping of a NAKed client

Long Syntax: DHCP.055 Decoupling address mapping of a NAKed client

Description: We just nacked the client, now decouple address mapping of it.

DHCP.056

Level: UE-ERROR

Short Syntax: DHCP.056 Bad format for option in packet - dropped

Long Syntax: DHCP.056 Bad format for option in packet - dropped

Description: Attempt to validate packet fails due to option verification

---

DHCP.057

Level: UI-ERROR

Short Syntax: DHCP.057 Cannot hook the UDP entry point for DHCP to receive packets

Long Syntax: DHCP.057 Cannot hook the UDP entry point for DHCP Server and/or DHCP Proxy to receive packets.

Description: Registration with UDP to hook into the BOOTP Server port has failed. The DHCP Server and/or the DHCP Proxy will not be able to receive any packets.

DHCP.058

Level: C-INFO

Short Syntax: DHCP.058 Replyblock indicates no message is to be genarated

Long Syntax: DHCP.058 Replyblock indicates no message is to be generated

Description: Replytype in the reply block indicates that no message is to be generated out to the client

DHCP.059

Level: UE-ERROR

Short Syntax: DHCP.059 Overload Option of received packet contains bad data(*bad_data*)

Long Syntax: DHCP.059 Overload Option of received packet contains bad data(*bad_data*)

Description: Option field of the received packet contains bad data

DHCP.060

Level: UE-ERROR

Short Syntax: DHCP.060 Aborting ACK due to discrepency in hostname in request and offer

Long Syntax: DHCP.060 Aborting ACK due to discrepency in hostname in request and offer

Description: Server has offered a hostname to a client, but when the client sends in the request packet, the hostname is different, so aborting the dhcpack

DHCP.061

Level: UE-ERROR

Short Syntax: DHCP.061 Request packet is not legible, ignored

Long Syntax: DHCP.061 Request packet is not legible, ignored

Description: The DHCP boot packet fails format verification. We will not process this packet any further since it is not understood by the server

DHCP.062

Level: UE-ERROR

Short Syntax: DHCP.062 Packet opcode *opcode* is not BOOTREQUEST

Long Syntax: DHCP.062 Packet opcode *opcode* is not BOOTREQUEST

Description: The server can only process dhcp packet of type bootrequest. If it is not, we need to drop the packet

DHCP.063

Level: C-INFO

Short Syntax: DHCP.063 Relay agent *relay_net_address* is *number_of_hops* hop away from client

Long Syntax: DHCP.063 Relay agent *relay_net_address* is *number_of_hops* hop away from the client

Description: Number of hops the relay agent is from the client

DHCP.064

Level: C-INFO

Short Syntax: DHCP.064 All client records are *action*, look for a *action*

Long Syntax: DHCP.064 All client records are *action*, look for a *action*

Description: Searching for a free client record, starting from unmapped ones to released ones

DHCP.065

Level: UE-ERROR

Short Syntax: DHCP.065 Unknown DHCP message type *message_type*

Long Syntax: DHCP.065 Unknown DHCP message type *message_type*

Description: Unknown DHCP Message type (option 53) in the received packet

DHCP.066

Level: C-INFO

Short Syntax: DHCP.066 ReplyBlock contains no message to build BOOTREPLY

Long Syntax: DHCP.066 ReplyBlock contains no message to build BOOTREPLY

Description: Replyblock is empty therefore cannot generate a replymessage

DHCP.067

Level: UI-ERROR

Short Syntax: DHCP.067 Buffer for reply message is not spplied in the replyblock

Long Syntax: DHCP.067 Buffer for reply message is not supplied in the replyblock

Description: Cannot generate reply message, buffer for reply message is not supplied in the reply block

DHCP.068

Level: C-INFO

Short Syntax: DHCP.068 Ping completes, reprocessing packet

Long Syntax: DHCP.068 Ping completes, reprocessing packet

Description: Ping response comes back, the code will reenter to process the DHCP packet

DHCP.069

Level: C-INFO

Short Syntax: DHCP.069 No DHCP message type in req pkt - pkt comes fr a BOOTP clnt

Long Syntax: DHCP.069 No DHCP message type in request - packet comes from a BOOTP client

Description: There is no DHCP message type in the request packet, must be coming from the BOOTP client

DHCP.070

Level: UE-ERROR

Short Syntax: DHCP.070 *packet_type* without magic cookie field

Long Syntax: DHCP.070 *packet_type* without magic cookie field

Description: BOOTP or DHCP packet without magic cookie field. For bootp, this is normal, for dhcp this is an error

DHCP.071

Level: UE-ERROR

Short Syntax: DHCP.071 *packet_type* without End option

Long Syntax: DHCP.071 *packet_type* without End option

Description: BOOTP or DHCP packet without and End option. For bootp, this is normal, for dhcp, this is an error

DHCP.072

Level: UE-ERROR

Short Syntax: DHCP.072 *packet_type* with unrecognized option *option*

Long Syntax: DHCP.072 *packet_type* with unrecognized option *option*

Description: Received BOOTP or DHCP packet with unrecognized option

DHCP.073

Level: C-INFO

Short Syntax: DHCP.073 Option cookie in string format is *vendor_string*

Long Syntax: DHCP.073 Option cookie in string format is *vendor_string*

Description: Option cookie, read in ascii string

DHCP.074

Level: C-INFO

Short Syntax: DHCP.074 DHCP option Client-id is specified in request block

Long Syntax: DHCP.074 DHCP option Client-identifier is specified in request block

Description: There is an option Client-identifier specified in request block

DHCP.075

Level: UE-ERROR

Short Syntax: DHCP.075 Length *length* of Client-id option is more than max of 64 bytes

Long Syntax: DHCP.075 Length *length* of Client-id option is more than max of 64 bytes

Description: Length of Client-id option which is specified in the received packet is longer than maximum allowed of 64 bytes

DHCP.076

Level: C-INFO

Short Syntax: DHCP.076 No clnt-id opt, use htype, hlen and chaddr to identify client

Long Syntax: DHCP.076 No clnt-id option, use htype, hlen and chaddr to identify client instead

Description: If client does not identify themselves in option 61, the server will use it's hardware type, hardware length and hardware address field to identify the client

DHCP.077

Level: UE-ERROR

Short Syntax: DHCP.077 Hardware address field is too large *length*

Long Syntax: DHCP.077 Hardware address field is too large *length*

Description: Client's hardware address field in the boot packet is too large

DHCP.078

Level: C-INFO

Short Syntax: DHCP.078 A requesting client did not identify itself, idtype= *id_type*, idlen= *id_length*

Long Syntax: DHCP.078 A requesting client did not identify itself, idtype= *id_type*, idlen= *id_length* \

Description: Shows the ip type and id len of a client who did not identify itself with the hardware address

DHCP.079

Level: UI-ERROR

Short Syntax: DHCP.079 dhcp server cant xmit pkt via net *network ID*

Long Syntax: DHCP.079 dhcp server cant transmit packet via via net *network ID*

Description: dhcp server is attempting to transmit packet via interface which is down

DHCP.080

Level: C-INFO

Short Syntax: DHCP.080 Request from *lsa_client_type* type Client is received

Long Syntax: DHCP.080 Request from *lsa_client_type* type Client is received

Description: dhcp packet is coming from LSA type of client

DHCP.081

Level: C-INFO

Short Syntax: DHCP.081 If LSA2 option also indicated LSA1 rules will override

Long Syntax: DHCP.081 If LSA2 option also indicated LSA1 rules will override

Description: If LSA2 option also indicated LSA1 rules will override

DHCP.082

Level: C-INFO

Short Syntax: DHCP.082 Client access denied

Long Syntax: DHCP.082 Client access denied

Description: Client access denied after checking usersite class (option 77)

DHCP.083

Level: C-INFO

Short Syntax: DHCP.083 DHCP packet type *Message_type* received

Long Syntax: DHCP.083 DHCP packet type *Message_type* received

Description: Request block indicates packet of type %S received

DHCP.084

Level: C-INFO

Short Syntax: DHCP.084 No reply is generated

Long Syntax: DHCP.084 No reply is generated

Description: This message is posted after process each of the sequence of packet and determine if the reply should not be generated

DHCP.085

Level: UI-ERROR

Short Syntax: DHCP.085 Not used

Long Syntax: DHCP.085 Not used

Description: Not used

DHCP.086

Level: C-INFO

Short Syntax: DHCP.086 *position_of_binder* Configured client inside binder showing client record

Long Syntax: DHCP.086 *position_of_binder* Configured client inside binder showing client record

Description: Searching for client specified in the binder and found the configuration in the server profile

DHCP.087

Level: C-INFO

Short Syntax: DHCP.087 IndexAddressRecord returned Null address record

Long Syntax: DHCP.087 IndexAddressRecord returned Null address record

Description: Attempt to index into the address record to get the information but the index is out of bound, so returning null record

DHCP.088

Level: C-INFO

Short Syntax: DHCP.088 DHCP Server is initialized

Long Syntax: DHCP.088 DHCP Server is initialized

Description: DHCP finished all initialization process. Now it is active

---

DHCP.089

Level: C-INFO

Short Syntax: DHCP.089 Server is not configured to support BOOTP client, *action_string*

Long Syntax: DHCP.089 Server is not configured to support BOOTP client, *action_string*

Description: Since the server is not configured to support the BOOTP client, we will send no reply, or release the binding, or cancel the reservation.

DHCP.090

Level: C-INFO

Short Syntax: DHCP.090 Generates no reply waiting for ping to complete

Long Syntax: DHCP.090 Generates no reply waiting for ping to complete

Description: Routine to generate reply is exiting and not build reply block because we are waiting for the result of ping to come back

---

DHCP.091

Level: C-INFO

Short Syntax: DHCP.091 Address manager receives command *command_type*

Long Syntax: DHCP.091 Address manager receives command *command_type*

Description: Address manager receives command to access or update an address

---

DHCP.092

Level: C-INFO

Short Syntax: DHCP.092 Function *functionname* entered

Long Syntax: DHCP.092 Function *functionname* entered

Description: This message is displayed when routine %S is entered

---

DHCP.093

Level: C-INFO

Short Syntax: DHCP.093 There is no outstanding exchange for this client, return

Long Syntax: DHCP.093 There is no outstanding exchange for this client, return

Description: There is no outstanding exchange for this client, return

---

DHCP.094

Level: C-INFO

Short Syntax: DHCP.094 Match on *grouptype* list

Long Syntax: DHCP.094 Match on *grouptype* list

Description: Found a match on balance or inorder list

DHCP.095

Level: UI-ERROR

Short Syntax: DHCP.095 An option is too big to fit into the option block

Long Syntax: DHCP.095 An option is too big to fit into the option block

Description: Server is trying to place an option in the option block (for an outgoing packet), but it is too big

---

DHCP.096

Level: UI-ERROR

Short Syntax: DHCP.096 Option block is overflown

Long Syntax: DHCP.096 Option block is overflown

Description: Option block is overflown for outgoing packet

DHCP.097

Level: C-INFO

Short Syntax: DHCP.097 Checking if the client is configured in the server, found *true_or_false*

Long Syntax: DHCP.097 Checking if the client is configured in the server, found *true_or_false*

Description: The routine which checks if the client is to be serviced an address is checking with the configuration to see if the client is specified in there

---

DHCP.098

Level: UI-ERROR

Short Syntax: DHCP.098 End option in between options of outgoing message

Long Syntax: DHCP.098 End option in between options of outgoing message

Description: End option appears in between other options when trying to compose reply message

---

DHCP.099

Level: UI-ERROR

Short Syntax: DHCP.099 Cannot allocate memory for *data_structure*

Long Syntax: DHCP.099 Cannot allocate Memory for *data_structure*

Description: Cannot allocate memory a runtime structure

Cause: The router memory is not large enough for this configuration

Action: Decrease configuration size or order more memory

Cause: The router has run out of memory due to a memory leak

Action: Contact Customer Service

DHCP.100

Level: UI-ERROR

Short Syntax: DHCP.100 Sub *subnet_address* with subgrp name *subnet_group_name* not added to list. Dup sub diff subgrp.

Long Syntax: DHCP.100 Subnet *subnet_address* with subnet group name *subnet_group_name* was not added to the list. It is a duplicate subnet with a different subnet group name

Description: A subnet already exists that has the same subnet address but a different subnet group name

---

DHCP.101

Level: UE-ERROR

Short Syntax: DHCP.101 options field is too short *option_field_size* for a magic cookie

Long Syntax: DHCP.101 options field is too short *option_field_size* for a magic cookie

Description: Option field in the received packet is too short

---

DHCP.102

Level: UE-ERROR

Short Syntax: DHCP.102 no magic cookie in options field of size *option_field_size*

Long Syntax: DHCP.102 no magic cookie in options field of size *option_field_size*

Description: There is no option field in the received packet.

DHCP.103

Level: UE-ERROR

Short Syntax: DHCP.103 Dropping a packet due to bad format in magic cookie field

Long Syntax: DHCP.103 Dropping a packet due to bad format in magic cookie field

Description: The received packet has been dropped due to bad format for magic cookie field

---

DHCP.104

Level: UE-ERROR

Short Syntax: DHCP.104 Options field does not contain ANY option

Long Syntax: DHCP.104 Options field does not contain ANY option

Description: The option field of the received packet does not contain any options

---

DHCP.105

Level: C-INFO

Short Syntax: DHCP.105 Renewing an *actionstring* lease, *action*

Long Syntax: DHCP.105 Renewing an *actionstring* lease, *action*

Description: This message indicates if we are renewing an unused lease or expired lease in the first case we will send no reply and second case will reply

---

DHCP.106

Level: C-INFO

Short Syntax: DHCP.106 Option: *optioncode*, length: *optionlen* value: *optiondata_decimal* (0x *optiondata_hex*)

Long Syntax: DHCP.106 Option: *optioncode*, length: *optionlen* value: *optiondata_decimal* (0x *optiondata_hex*)

Description: Describe option code, option length and the value of option inside a DHCP packet

---

DHCP.107

Level: C-INFO

Short Syntax: DHCP.107 Offer not accepted by client - no reply

Long Syntax: DHCP.107 Offer not accepted by client - no reply

Description: The request packet indicates that the client has used another server, therefore send no reply

---

DHCP.108

Level: C-INFO

Short Syntax: DHCP.108 Option: *option*, length: *length_of_option*

Long Syntax: DHCP.108 Option: *option*, length: *length_of_option*

Description: Options that client requests

DHCP.109

Level: C-INFO

Short Syntax: DHCP.109 Option: Parameter Request List, length *length*

Long Syntax: DHCP.109 Option: Parameter Request List, length *length*

Description: Option parameter request list, length

DHCP.110

Level: C-INFO

Short Syntax: DHCP.110 Option *option* requested under parameter request list (opt 55)

Long Syntax: DHCP.110 Option *option* requested under parameter request list (option 55)

Description: Option inside parameter request list (option 55) which are requested

DHCP.111

Level: UE-ERROR

Short Syntax: DHCP.111 Undefined option *option* requested, ignored

Long Syntax: DHCP.111 Undefined option *option* requested, ignored

Description: Undefined option is specified in the parameter request list, ignored

DHCP.112

Level: UE-ERROR

Short Syntax: DHCP.112 Option: *option*,length *length* value: <no conversion defined for value>

Long Syntax: DHCP.112 Option: *option*,length *length* value: <no conversion defined for value>

Description: This option is valid (opcode is less than max option), but there is no conversion routine defined in our code is defined

DHCP.113

Level: UE-ERROR

Short Syntax: DHCP.113 Option: *option* not valid

Long Syntax: DHCP.113 Option: *option* not valid

Description: Option number is not recognized, and it is beyond the maximum 255

DHCP.114

Level: UE-ERROR

Short Syntax: DHCP.114 size of opt *option_code* exceeds max size of opt in received pkt

Long Syntax: DHCP.114 size of option *option_code* exceeds field boundary in the received packet

Description: The size of the option fields in the received packet exceeds maximum of what option should be

DHCP.115

Level: UE-ERROR

Short Syntax: DHCP.115 There is no End Option in field with size *option_size*

Long Syntax: DHCP.115 There is no End Option in field with size *option_size*

Description: There is no End Option in the received packet

DHCP.116

Level: C-INFO

Description: Address manager experience error while commiting, binding or cancelling

DHCP.117

Level: C-INFO

Short Syntax: DHCP.117 No more free or reusable client record

Long Syntax: DHCP.117 No more free or reusable client record

Description: Looking for a free client record to map to an address but find none that can be used

DHCP.118

Level: C-INFO

Short Syntax: DHCP.118 No addr available with class info, ignore the class and try again

Long Syntax: DHCP.118 No address available with class info, ignore the class and try again

Description: No address available with class info, ignore the class and try again

---

DHCP.119

Level: C-INFO

Short Syntax: DHCP.119 First check for class match in global classes

Long Syntax: DHCP.119 First check for class match in global classes

Description: Searching for a requested class in the global level

---

DHCP.120

Level: C-INFO

Short Syntax: DHCP.120 Searching all the classes at this subnet level

Long Syntax: DHCP.120 Searching all the classes at this subnet level

Description: After searching for class in the global level and find none matching the requested class, not start looking in the subnet level

---

DHCP.121

Level: C-INFO

Short Syntax: DHCP.121 Not doing MSSW because of client statement

Long Syntax: DHCP.121 Not doing MSSW because of client statement

Description: Not checking multiple subnet on the same wire because client has a specific configuration on this server

---

DHCP.122

Level: C-INFO

Short Syntax: DHCP.122 Requested address not in address pool - no reply

Long Syntax: DHCP.122 Requested address not in address pool - no reply

Description: Requested address is not in the pool, therefore server does not own it. Send no reply

---

DHCP.123

Level: UI-ERROR

Short Syntax: DHCP.123 Neither client nor address was specified in query mapper

Long Syntax: DHCP.123 Neither client nor address was specified in query mapper

Description: Neither client nor address was specified in query mapper\r\n

---

DHCP.124

Level: UI-ERROR

Short Syntax: DHCP.124 Client record has a bad address index

Long Syntax: DHCP.124 Client record has a bad address index

Description: Client record has an index to a mapped address but it is bad index

---

DHCP.125

Level: UI-ERROR

Short Syntax: DHCP.125 Query both address and client at the same time is not supported

Long Syntax: DHCP.125 Query both address and client at the same time is not supported

Description: Query both address and client at the same time is not supported

---

DHCP.126

Level: C-INFO

Short Syntax: DHCP.126 Bad mapping for address *address*

Long Syntax: DHCP.126 Bad mapping for address *address*

Description: Index of address does not match the addressIndex in the client record

---

DHCP.127

Level: C-INFO

Short Syntax: DHCP.127 MSSW list has wrapped, no more to check

Long Syntax: DHCP.127 MSSW list has wrapped, no more to check

Description: Finish searching through all the subnet on the same wire

---

DHCP.128

Level: C-INFO

Short Syntax: DHCP.128 Turning twiddle on due to request type

Long Syntax: DHCP.128 Turning twiddle on due to request type

Description: Turning twiddle on due to request type

DHCP.129

Level: C-INFO

Short Syntax: DHCP.129 We will remain silent server doesn't own the address

Long Syntax: DHCP.129 We will remain silent server doesn't own the address

Description: Server does not own the address, therefore will remain silent

DHCP.130

Level: C-INFO

Short Syntax: DHCP.130 Received request to index into address record 0

Long Syntax: DHCP.130 Received request to index into address record 0

Description: Received request to index into address record 0

DHCP.131

Level: C-INFO

Short Syntax: DHCP.131 Out of bounds for this subnet, trying next subnet

Long Syntax: DHCP.131 Out of bounds for this subnet, trying next subnet

Description: Out of bounds for this subnet, trying next subnet

DHCP.132

Level: C-INFO

Short Syntax: DHCP.132 client *client_id* authentication failed - no reply

Long Syntax: DHCP.132 client *client_id* authentication failed - no reply

Description: Client specified failed authentication, server will not send reply

DHCP.133

Level: C-INFO

Short Syntax: DHCP.133 Client requested addr *address_string*, but addr already in use

Long Syntax: DHCP.133 Client requested addr *address_string*, but address is already in use

Description: Client has requested address that is being in used by another station already

DHCP.134

Level: C-INFO

Short Syntax: DHCP.134 Removing old addr record for addr *address_string*

Long Syntax: DHCP.134 Removing old address record for address *address_string*

Description: Found old address record for client who will be assigned another address

DHCP.135

Level: UI-ERROR

Short Syntax: DHCP.135 Client statement found but add in use - REPLY NONE

Long Syntax: DHCP.135 Client statement found but address in use - REPLY NONE

Description: Client is configured for a particular address on the server, but that address is already in use by other station. This indicates configuration problem

DHCP.136

Level: C-INFO

Short Syntax: DHCP.136 Address *ipaddress* has been *action*

Long Syntax: DHCP.136 Address *ipaddress* has been *action*

Description: Address has been reserved for client

DHCP.137

Level: C-INFO

Short Syntax: DHCP.137 Offering *type_of_request* address to the client - REPLY OFFER

Long Syntax: DHCP.137 Offering *type_of_request* address to the client - REPLY OFFER

Description: Offering desired or reserved type of address to the client in a reply OFFER packet

DHCP.138

Level: C-INFO

Short Syntax: DHCP.138 Binder status *status_string*

Long Syntax: DHCP.138 Binder status *status_string*

Description: Status of binder as returned by the address manager

DHCP.139

Level: C-INFO

Short Syntax: DHCP.139 Failed to *action* addr for clnt *client_string* - no reply

Long Syntax: DHCP.139 Failed to *action* an address for client *client_string* - no reply

Description: Failed to have an address reserved or bound for client - will send no reply

DHCP.140

Level: C-INFO

Short Syntax: DHCP.140 client access validated

Long Syntax: DHCP.140 client access validated

Description: Client passess access validation

DHCP.141

Level: C-INFO

Short Syntax: DHCP.141 Lease was reserved but no outstanding exchange

Long Syntax: DHCP.141 Lease was reserved but no outstanding exchange

Description: Lease is found in reserved state but there is no outstanding exchange information about this client

DHCP.142

Level: CE-ERROR

Short Syntax: DHCP.142 DISCOVER from client *client_id* already bound with *ipaddress*

Long Syntax: DHCP.142 DISCOVER from client *client_id* already bound with *ipaddress*

Description: DISCOVER message from client is already bound to an address

DHCP.143

Level: C-INFO

Short Syntax: DHCP.143 Previously bound/no address available for client - NO REPLY

Long Syntax: DHCP.143 Previously bound/no address available for client - NO REPLY

Description: Address Previously bound, not available for client

DHCP.144

Level: CE-ERROR

Short Syntax: DHCP.144 Client *client_string* sent a repeated DISCOVER

Long Syntax: DHCP.144 Client *client_string* sent a repeated DISCOVER

Description: Client sent a repeated DISCOVER since there is already an exchanged information for this client

DHCP.145

Level: C-INFO

Short Syntax: DHCP.145 Expired/released address not available for the client - NO REPLY

Long Syntax: DHCP.145 Previously bound/no addr available for clnt - NO REPLY

Description: Expired/released address not available for the client - NO REPLY

DHCP.146

Level: UI-ERROR

Short Syntax: DHCP.146 Missing global *record_type* record *record_index*

Long Syntax: DHCP.146 Missing global *record_type* record *record_index*

Description: The record corresponding to the record index is missing

Cause: Configuration has become corrupted

Action: Delete all the global records of this type and re-enter

DHCP.147

Level: C-INFO

Short Syntax: DHCP.147 Client id matches an active exchange

Long Syntax: DHCP.147 Client id matches an active exchange

Description: Found client id in the exchanged queue

DHCP.148

Level: UI-ERROR

Short Syntax: DHCP.148 Number of outstanding exchanges reached high threshold *xThresholdHigh*

Long Syntax: DHCP.148 Number of outstanding exchanges reached high threshold *xThresholdHigh*

Description: Number of outstanding exchanges

reached high threshold of 64

DHCP.149

Level: UI-ERROR

Short Syntax: DHCP.149 Number of outstanding exchanges decreased to the low threshold *xThresholdLow*

Long Syntax: DHCP.149 Number of outstanding exchanges decreased to the low threshold *xThresholdLow*

Description: Number of outstanding exchanges decreased to the low threshold of 8

DHCP.150

Level: UI-ERROR

Short Syntax: DHCP.150 Failed to address client *client_string*

Long Syntax: DHCP.150 Failed to address client *client_string*

Description: Failed allocate a client record for mapping with an address

---

DHCP.151

Level: C-INFO

Short Syntax: DHCP.151 Cant locate client *client_string*, no IP address supplied

Long Syntax: DHCP.151 Cannot locate client *client_string*, no IP address supplied

Description: Attempt to locate a configured client for an ip address, but no ip address is supplied due to no existing information exists about this client

---

DHCP.152

Level: C-INFO

Short Syntax: DHCP.152 Found match for client id *client_hardware_address* in *type_pf_scope*

Long Syntax: DHCP.152 Found match for client id *client_hardware_address* in *type_pf_scope*

Description: locateConfiguredClient found client id matched in global or subnet scope

---

DHCP.153

Level: C-INFO

Short Syntax: DHCP.153 Check sub move, clue *IpAddressClue* vs. req IP *ReqIpAddress*

Long Syntax: DHCP.153 Check client subnet movement is comparing addr clue of *IpAddressClue*, with requested IP address *ReqIpAddress*

Description: Checking if the client has moved from one subnet to another by comparing the interface IP address with the requested IP address from client

---

DHCP.154

Level: C-INFO

Short Syntax: DHCP.154 Cant find addr *ipaddress* in addr records, instance *position*

Long Syntax: DHCP.154 Cant find addr *ipaddress* in addr records, instance *position*

Description: Display the position of this message regarding the module who requested to find an ip address

---

DHCP.155

Level: C-INFO

Short Syntax: DHCP.155 No client info found in profile

Long Syntax: DHCP.155 No client information found in profile

Description: Could not find information about this client in the profile

---

DHCP.156

Level: C-INFO

Short Syntax: DHCP.156 Comparing requested ip address *requested_ip_address* with subnet *subnet_ip_address*

Long Syntax: DHCP.156 Comparing requested ip address *requested_ip_address* with subnet *subnet_ip_address*

Description: Checking client subnet movement is comparing requested ip address with subnet IP address

---

DHCP.157

Level: C-INFO

Short Syntax: DHCP.157 Checking multiple subnet on same wire *temp_subnet_label* against *inorder_label*

Long Syntax: DHCP.157 Checking multiple subnet on same wire *temp_subnet_label* against *inorder_label*

Description: Checking for multiple subnet on the same wire

DHCP.158

Level: C-INFO

Short Syntax: DHCP.158 *MsgString*

Long Syntax: DHCP.158 *MsgString*

Description: Any message string

DHCP.159

Level: C-INFO

Short Syntax: DHCP.159 Checking *TypeofSubnet* addr *SubnetLabel1* against *SubnetLabel2*

Long Syntax: DHCP.159 Checking *TypeofSubnet* addr *SubnetLabel1* against *SubnetLabel2*

Description: Checking type of subnet with addr Subnet1 against subnet2

DHCP.160

Level: C-INFO

Short Syntax: DHCP.160 Client requested classname matched in *scope* scope name = *classname*

Long Syntax: DHCP.160 Client requested class name matched in *scope* scope name = *classname*

Description: Search for the client requested class name matched with classname in the global/regexec/regexec-partial scope name = %S

DHCP.161

Level: C-INFO

Short Syntax: DHCP.161 Address manager got unrecognized command *command*

Long Syntax: DHCP.161 Address manager got unrecognized command *command*

Description: Address Manager is called with unrecognized command for an address

DHCP.162

Level: C-INFO

Short Syntax: DHCP.162 Query portfolio for *ipaddress* failed

Long Syntax: DHCP.162 Query portfolio for *ipaddress* failed

Description: Query for address in portfolio returns failure

DHCP.163

Level: C-INFO

Short Syntax: DHCP.163 Address *ipaddress* has no profile in this server, instance: *position*

Long Syntax: DHCP.163 Address *ipaddress* has no profile in this server, instance: *position*

Description: Address specified has no profile in this server

DHCP.164

Level: C-INFO

Short Syntax: DHCP.164 Finding clnt to map w/ addr *binder_ip_address* of subn *binder_subnet*

Long Syntax: DHCP.164 Trying to locate client from pool or creating new one to map with address *binder_ip_address* of subnet *binder_subnet*

Description: Searching client pool for a specified client, if not found will create new client record in order to prepare for mapping with an address and subnet specified

DHCP.165

Level: C-INFO

Short Syntax: DHCP.165 Found client id *client_string* in client records

Long Syntax: DHCP.165 Found client id *client_string* requested in client records

Description: Found a match of client id in the client records

DHCP.166

Level: C-INFO

Short Syntax: DHCP.166 Query prof for addr *binder_address* for subn *binder_subnet* mssw *mssw*

Long Syntax: DHCP.166 Query profile about address *binder_address* for subnet *binder_subnet* mssw *mssw*

Description: am_addressClient queries profile repository about an address of a subnet

DHCP.167

Level: C-INFO

Short Syntax: DHCP.167 Server doesnt own requested IP address *ipaddress* will ignore the suggestion

Long Syntax: DHCP.167 Server does not own the requested IpAddr *ipaddress* will ignore the suggestion

Description: server does not own requested ip address, will ignore the suggestion

DHCP.168

Level: UE-ERROR

Short Syntax: DHCP.168 client *client_string* had address *ipaddress* mapped previously

Long Syntax: DHCP.168 client *client_string* had address *ipaddress* mapped previously

Description: Client specified had another address mapped previously

DHCP.169

Level: C-INFO

Short Syntax: DHCP.169 Req might come fr (sub)net which this srvr isnt cfg to managed

Long Syntax: DHCP.169 Request might have come from (sub)net for which this server is not configured to managed

Description: Interface address does not match with the requested address, therefore it may have come from (sub)net which this server is not meant to manage

---

DHCP.170

Level: C-INFO

Short Syntax: DHCP.170 Clnt *client_string* suggested addr *ipaddress* is in range

Long Syntax: DHCP.170 Client *client_string* suggested address *ipaddress* is in range

Description: Clients suggested address is in the range

DHCP.171

Level: C-INFO

Short Syntax: DHCP.171 Clnt *client_string* req addr *ipaddress* is out of range

Long Syntax: DHCP.171 Client *client_string* requested address *ipaddress* is out of range

Description: Client suggested address is out of range

DHCP.172

Level: UI-ERROR

Short Syntax: DHCP.172 *String*

Long Syntax: DHCP.172 *String*

Description: Message of type error

DHCP.173

Level: C-INFO

Short Syntax: DHCP.173 Client *client_string* moved from addr1 *strAddr1* to addr2 *strAddr2*

Long Syntax: DHCP.173 Client *client_string* moved from addr1 *strAddr1* to addr2 *strAddr2*

Description: Client specified moved from one address to another address

---

DHCP.174

Level: C-INFO

Short Syntax: DHCP.174 Client *client_string* had no previous mapping, getting one

Long Syntax: DHCP.174 Client *client_string* had no previous mapping, getting one

Description: Client specified had no previous mapping, getting one

---

DHCP.175

Level: C-INFO

Short Syntax: DHCP.175 *position* Net *ipaddress* has no more available addresses

Long Syntax: DHCP.175 *position* Net *ipaddress* has no more available addresses

Description: Network specified has no more addresses to serve

---

DHCP.176

Level: C-INFO

Short Syntax: DHCP.176 Curr sub *subnet* is out of addrs, trying next sub 0x

Long Syntax: DHCP.176 Current subnet *subnet* is out of addresses, trying next subnet 0x

Description: Trying to get an address in the current subnet, but the address is exhausted therefore trying another subnet on the same wire

---

DHCP.177

Level: UE-ERROR

Short Syntax: DHCP.177 All new client records *index* are filled by previous clients

Long Syntax: DHCP.177 All new client records *index* are filled by previous clients

Description: Index of highest client records that are filled by previous clients

---

DHCP.178

Level: C-INFO

Short Syntax: DHCP.178 Subnet *subnet* with mask *subnetmask* is removed due to overlapping

Long Syntax: DHCP.178 Subnet *subnet* with mask *subnetmask* is removed due to overlapping

Description: Subnet with specified mask is removed due to overlapping

---

DHCP.179

Level: UE-ERROR

Description: Looking for an available address record in this order, available, stocked, release, expired

---

DHCP.180

Level: UE-ERROR

Short Syntax: DHCP.180 Bad subnet mask

Long Syntax: DHCP.180 Bad subnet mask

Description: Bad subnet mask. There are more than 24 digits of 0 in the mask

---

DHCP.181

Level: UE-ERROR

Short Syntax: DHCP.181 No expired address is available, nothing left to try on

Long Syntax: DHCP.181 No expired address is available, nothing left to try on

Description: No expired address is available, nothing left to try on

---

DHCP.182

Level: UI-ERROR

Short Syntax: DHCP.182 Cannot find client *client_string* to *action* its reservation

Long Syntax: DHCP.182 Cannot find client *client_string* to *action* its reservation

Description: Cannot find client to commit or cancel its reservation

---

DHCP.183

Level: C-INFO

Short Syntax: DHCP.183 To *action* an address *binder_address* other than the reserved *record_address*

Long Syntax: DHCP.183 To *action* an address *binder_address* other than the reserved *record_address*

Description: Client tries to commit or cancel an address which is different from the reserved address

---

DHCP.184

Level: C-INFO

Short Syntax: DHCP.184 Received packet of type *opcode* sz *request_size* option sz *option_size*

Long Syntax: DHCP.184 Received a packet of type *opcode* size *request_size* option size *option_size*

Description: Received a new DHCP packet with opcode, size, and option size specified

---

DHCP.185

Level: C-INFO

Short Syntax: DHCP.185 To release an address *ipaddress* other than the bound *boundaddress*

Long Syntax: DHCP.185 To release an address *ipaddress* other than the bound *boundaddress*

Description: Client tries to release an address other than the bound address

---

DHCP.186

Level: C-INFO

Short Syntax: DHCP.186 Clnt *client_string* not known to addr mapr, ask clientele

Long Syntax: DHCP.186 Client *client_string* is not known to address mapper, ask clientele

Description: Client specified does not yet have an address mapped to it, check to see if we should serve address to it

---

DHCP.187

Level: C-INFO

Short Syntax: DHCP.187 Clnt *client_string* known to addr mapr, stat = *status*

Long Syntax: DHCP.187 Client *client_string* is known to address mapper, status = *status*

Description: A client is found in the address mapper with a valid status

---

DHCP.188

Level: C-INFO

Short Syntax: DHCP.188 *modulename* failed when queried about client *client_string*

Long Syntax: DHCP.188 *modulename* failed when queried about client *client_string*

Description: Display module name which query about a client

DHCP.189

Level: C-INFO

Short Syntax: DHCP.189 Address *ipaddress* is not known to address mapper, status *status*

Long Syntax: DHCP.189 Address *ipaddress* is not known to address mapper, status *status*

Description: Display status and address which is not known to address mapper

---

DHCP.190

Level: C-INFO

Short Syntax: DHCP.190 Address mapper failed when queried about address *ipaddress*

Long Syntax: DHCP.190 Address mapper failed when queried about address *ipaddress*

Description: Address mapper failed when queried about an address

DHCP.191

Level: C-INFO

Short Syntax: DHCP.191 Responding with client info - REPLY ACK, client ipaddress *ipaddress*

Long Syntax: DHCP.191 Responding with client information - REPLY ACK, client ipaddress *ipaddress*

Description: Response to inform packet from client specified

DHCP.192

Level: UI-ERROR

Short Syntax: DHCP.192 Error adding client subnet to list

Long Syntax: DHCP.192 Error adding client subnet to list

Description: Error occured while adding a subnet created for a client to the subnet list

DHCP.193

Level: C-INFO

Short Syntax: DHCP.193 Client *client_string* sent REQUEST without previous binding with this srvr

Long Syntax: DHCP.193 Client *client_string* sent REQUEST without previous binding with this server

Description: Client sent REQUEST without previous binding with this server

DHCP.194

Level: C-INFO

Short Syntax: DHCP.194 Client *client_string* specified another server *server_ipaddress* in renew REQUEST

Long Syntax: DHCP.194 Client *client_string* specified another server *server_ipaddress* in renew REQUEST

Description: Client specified another server in renew REQUEST

DHCP.195

Level: C-INFO

Short Syntax: DHCP.195 Client *client_string* didn't specify a server in REQUEST

Long Syntax: DHCP.195 Client *client_string* didn't specify a server in REQUEST

Description: Client didn't specify a server in REQUEST packet

DHCP.196

Level: C-INFO

Short Syntax: DHCP.196 Not used

Long Syntax: DHCP.196 Not used

Description: Not used

DHCP.197

Level: C-INFO

Short Syntax: DHCP.197 Client *client_string* requests an unavailable address - NO REPLY

Long Syntax: DHCP.197 Client *client_string* requests an unavailable address - NO REPLY

Description: Address Manager fails to bind address for the client, send no reply

DHCP.198

Level: UI-ERROR

Short Syntax: DHCP.198 State of exchange is out of sync with address manager

Long Syntax: DHCP.198 State of exchange is out of sync with address manager

Description: State of an address recorded by the address manager is different from what is in the exchange block

DHCP.199

Level: C-INFO

Short Syntax: DHCP.199 Requesting an existing lease - REPLY ACK

Long Syntax: DHCP.199 Requesting an existing lease - REPLY ACK

Description: Requesting an existing lease - REPLY ACK

DHCP.200

Level: C-INFO

Short Syntax: DHCP.200 Failed to renew for client *client_string* - reply nak

Long Syntax: DHCP.200 Failed to renew for client *client_string* - reply nak

Description: Address manager fails to renew lease for client, will send reply nak

DHCP.201

Level: UE-ERROR

Short Syntax: DHCP.201 Client *client_string* tries to renew *type_of_lease* lease

Long Syntax: DHCP.201 Client *client_string* tries to renew *type_of_lease* lease

Description: Client tries to renew released or expired lease

DHCP.202

Level: CI-ERROR

Short Syntax: DHCP.202 Failed to bind for client *client_string* while *action*

Long Syntax: DHCP.202 Failed to bind for client *client_string* while *action*

Description: Failed to bind for client while release or expired

DHCP.203

Level: C-INFO

Short Syntax: DHCP.203 Offer was selected by client *client_string*

Long Syntax: DHCP.203 Offer was selected by client *client_string*

Description: Server sends an offer which is selected by client speicified

DHCP.204

Level: UE-ERROR

Short Syntax: DHCP.204 Client *client_string* REQUESTed lease (*lease*) differ from reserved (*reserved_lease*)

Long Syntax: DHCP.204 Client *client_string* REQUESTed lease (*lease*) differ from reserved (*reserved_lease*)

Description: Client REQUESTed leasetime which is different from reserved time

DHCP.205

Level: C-INFO

Short Syntax: DHCP.205 Addr *ipaddress* has been bound to client *client_string*

Long Syntax: DHCP.205 Address *ipaddress* has been bound to client *client_string*

Description: Display an address which has been bound to a client

DHCP.206

Level: UI-ERROR

Short Syntax: DHCP.206 Reserved *address1* but committed *address2* for client *client_string*

Long Syntax: DHCP.206 Reserved *address1* but committed *address2* for client *client_string*

Description: Client has one address reserved, but server is committing another address for it

DHCP.207

Level: C-INFO

Short Syntax: DHCP.207 Requesting an unavailable reserved address - REPLY NAK

Long Syntax: DHCP.207 Requesting an unavailable reserved address - REPLY NAK

Description: Requesting an unavailable reserved address - REPLY NAK

DHCP.208

Level: C-INFO

Short Syntax: DHCP.208 OFFER was not selected by client *client_string*

Long Syntax: DHCP.208 OFFER was not selected by client *client_string*

Description: OFFER was not selected by client

DHCP.209

Level: C-INFO

Short Syntax: DHCP.209 Address reservation has been canceled

Long Syntax: DHCP.209 Address reservation has been canceled

Description: Address reservation has been canceled

DHCP.210

Level: C-INFO

Short Syntax: DHCP.210 Client *client_string* did not id any server in response to an offer

Long Syntax: DHCP.210 Client *client_string* did not id any server in response to an offer

Description: Client did not id any server in response to an offer

DHCP.211

Level: C-INFO

Short Syntax: DHCP.211 Client *client_string* may have recvd addr from another srvr - REPLY NONE

Long Syntax: DHCP.211 Client *client_string* may have received address from another server - REPLY NONE

Description: Client may have received address from another server - REPLY NONE

DHCP.212

Level: C-INFO

Short Syntax: DHCP.212 Client *client_string* moved to another subnet -- REPLY NAK

Long Syntax: DHCP.212 Client *client_string* moved to another subnet -- REPLY NAK

Description: Client moved to another subnet -- REPLY NAK

DHCP.213

Level: C-INFO

Short Syntax: DHCP.213 NO REPLY - clnt *client_string* req diff addr *requested_ipaddress* fr what avail on srvr *available_ipaddress*

Long Syntax: DHCP.213 NO REPLY generated because client *client_string* requested a different address *requested_ipaddress* from what we have on the server *available_ipaddress*

Description: NO REPLY generated because client requested a different address from what we have on the server

DHCP.214

Level: C-INFO

Short Syntax: DHCP.214 Received a *type_of_packet* from client *client_string* who failed authentication

Long Syntax: DHCP.214 Received a *type_of_packet* from client *client_string* who failed authentication

Description: Received packet from client who failed authentication

DHCP.215

Level: C-INFO

Short Syntax: DHCP.215 Client *client_string* declined a non-existing lease

Long Syntax: DHCP.215 Client *client_string* declined a non-existing lease

Description: Client declined a non-existing lease

DHCP.216

Level: C-INFO

Short Syntax: DHCP.216 Clnt *client_string action* non-existing lease in another srvr *server_string* - no action

Long Syntax: DHCP.216 Client *client_string* attempts to *action* non-existing lease in another server *server_string* - no action

Description: Received from client to decline or release non-existing lease in another server, no reply

DHCP.217

Level: C-INFO

Short Syntax: DHCP.217 Client *client_string* sends a *action* pkt but didn't specify srvr

Long Syntax: DHCP.217 Client *client_string* sends a *action* packet but didn't specify server

Description: Client sends release or decline packet but does not specify the server, send no reply

DHCP.218

Level: C-INFO

Short Syntax: DHCP.218 Client *client_string* has *action* a reserved lease for address *lease_ipaddress*

Long Syntax: DHCP.218 Client *client_string* has *action* a reserved lease for address *lease_ipaddress*

Description: Client specified in the client id string has denied or release a reserved lease for ipaddress

DHCP.219

Level: UE-ERROR

Short Syntax: DHCP.219 Client *client_string server_string* a reserved lease in this srvr but specified another srvr *server_ip*

Long Syntax: DHCP.219 Client *client_string server_string* a reserved lease in this srvr but specified another srvr *server_ip*

Description: Client has declined or released a reserved lease in this server but specified another server in the option server id

DHCP.220

Level: C-INFO

Short Syntax: DHCP.220 Client *client_string Action* a binding for address *ipaddress*

Long Syntax: DHCP.220 Client *client_string Action* a binding for address *ipaddress*

Description: Client has declined or release a binding for the address specified

DHCP.221

Level: UE-ERROR

Short Syntax: DHCP.221 Client *client_string action* a lease in this srvr but has another srvr's id *server_ipaddress*

Long Syntax: DHCP.221 Client *client_string action* a running lease in this srvr but has another srvr's id *server_ipaddress*

Description: Client has declined or released a running lease in this server but has another server's id

DHCP.222

Level: UE-ERROR

Short Syntax: DHCP.222 Client *client_string* sends a decline after release - send no reply

Long Syntax: DHCP.222 Client *client_string* sends a decline after release - do not reply

Description: Client specified sends a decline packet after it has already released the lease, do not send reply

DHCP.223

Level: C-INFO

Short Syntax: DHCP.223 Client *client_string action* after lease has expired - no reply

Long Syntax: DHCP.223 Client *client_string action* after lease has expired - no reply

Description: Client specified has declined or released an address after the lease has expired

DHCP.224

Level: C-INFO

Short Syntax: DHCP.224 Failed to cancel after declined by client *client_string*

Long Syntax: DHCP.224 Failed to cancel after declined by client *client_string*

Description: Failed to cancel after declined by client

DHCP.225

Level: C-INFO

Short Syntax: DHCP.225 Client *client_string* has released a non-existing lease

Long Syntax: DHCP.225 Client *client_string* has released a non-existing lease

Description: Client has requested to release a non-existing lease

DHCP.226

Level: UE-ERROR

Short Syntax: DHCP.226 Client *client_string* RELEASEs a lease that has been released

Long Syntax: DHCP.226 Client *client_string* RELEASEs a lease that has been released

Description: Client requested to release a lease that has already been released

DHCP.227

Level: UI-ERROR

Short Syntax: DHCP.227 Address manager returned unknown status *status*

Long Syntax: DHCP.227 Address manager returned unknown status *status*

Description: Address manager returned unknown status

DHCP.228

Level: C-INFO

Short Syntax: DHCP.228 Failed to release a lease for client *client_string*

Long Syntax: DHCP.228 Failed to release a lease for client *client_string*

Description: Address manager fails request to release a lease for a client specified

DHCP.229

Level: C-INFO

Short Syntax: DHCP.229 BOOTP client *client_string* is authenticated

Long Syntax: DHCP.229 BOOTP client *client_string* is authenticated

Description: BOOTP client has passed authentication

DHCP.230

Level: C-INFO

Short Syntax: DHCP.230 DHCP client *client_string* has address *ipaddress* reserved, then became BOOTP

Long Syntax: DHCP.230 DHCP client *client_string* has address *ipaddress* reserved, then became BOOTP

Description: A dhcp client has reserved an address then changed from dhcp to bootp

---

DHCP.231

Level: C-INFO

Short Syntax: DHCP.231 BOOTP client *client_string* has an infinite lease

Long Syntax: DHCP.231 BOOTP client *client_string* has an infinite lease

Description: BOOTP client has an infinite lease

DHCP.232

Level: C-INFO

Short Syntax: DHCP.232 Client *client_string* has a previously *action* address

Long Syntax: DHCP.232 Client *client_string* has a previously *action* address

Description: Client has a previously released or expired an address

---

DHCP.233

Level: C-INFO

Short Syntax: DHCP.233 Offering a BOOTP clnt a permanent lease addr *ipaddress* - REPLY OFFER

Long Syntax: DHCP.233 Offering a BOOTP client a permanent lease addr *ipaddress* - REPLY OFFER

Description: Offering a BOOTP client a permanent lease address - REPLY OFFER

DHCP.234

Level: C-INFO

Short Syntax: DHCP.234 Failed to allocate buffer for *type_of_buffer*, size *size*

Long Syntax: DHCP.234 Failed to allocate buffer for *type_of_buffer*, size *size*

Description: Failed to allocate buffer for reply packet or tranmit buffer

---

DHCP.235

Level: UI-ERROR

Short Syntax: DHCP.235 Cannot find client *client_string* in client record

Long Syntax: DHCP.235 Cannot find client *client_string* in client record

Description: Cannot find client in client records

DHCP.236

Level: UI-ERROR

Short Syntax: DHCP.236 Address record has a bad client index *index*

Long Syntax: DHCP.236 Address record has a bad client index *index*

Description: Address record has a mapping for a client but the index is bad

---

DHCP.237

Level: C-INFO

Short Syntax: DHCP.237 Clnt *client_string* (*status*) has no addr mapped to it

Long Syntax: DHCP.237 Client *client_string* (status *status*) has no address mapped to it

Description: A request through the query mapper has been made about a client, and it is found to have no address associated with it, and the it shows the status of that client

---

DHCP.238

Level: UI-ERROR

Short Syntax: DHCP.238 Address *ipaddress* has no mapped client but in mapped state *state*

Long Syntax: DHCP.238 Address *ipaddress* has no mapped client but in mapped state *state*

Description: The address has no mapped client, but the status is indicating that it is mapped

---

DHCP.239

Level: UI-ERROR

Short Syntax: DHCP.239 Address *ipaddress* has a running lease for no client

Long Syntax: DHCP.239 Address *ipaddress* has a running lease for no client

Description: The address specified has a valid lease but has no client associated with it

---

DHCP.240

Level: C-INFO

Short Syntax: DHCP.240 No move check since addr *ipaddress1* is mapped to clnt and clue is *ipaddress2*

Long Syntax: DHCP.240 No check for movement since Address *ipaddress1* is mapped to client and clue is *ipaddress2*

Description: Client has an address mapped previously and it is of the same subnet as the clue, therefore no need to do subnet movement check

---

DHCP.241

Level: UI-ERROR

Short Syntax: DHCP.241 Unable to copy *numbytes* bytes of options due to not enough room

Long Syntax: DHCP.241 Unable to copy *numbytes* bytes of options due to not enough room

Description: Unable to copy number of bytes of options due to not enough room

---

DHCP.242

Level: C-INFO

Short Syntax: DHCP.242 Generating a *DHCPType* reply

Long Syntax: DHCP.242 Generating a *DHCPType* reply

Description: Generating a reply of type specified

DHCP.243

Level: UI-ERROR

Short Syntax: DHCP.243 Failed to allocate block of memory of *block_size* bytes

Long Syntax: DHCP.243 Failed to allocate block of memory of *block_size* bytes

Description: Failed to allocate memory for exchange block or address or client record

---

DHCP.244

Level: C-INFO

Short Syntax: DHCP.244 Client *client_string* is *action* clientele list

Long Syntax: DHCP.244 Client *client_string* is *action* clientele list

Description: Client specified is authenticated or rejected by clientel list

---

DHCP.245

Level: UI-ERROR

Short Syntax: DHCP.245 Failed to initialize the address pool

Long Syntax: DHCP.245 Failed to initialize the address pool

Description: Failed to initialize the address pool due to different reason such as memory allocation, validation of address and client in profile.

DHCP.246

Level: UI-ERROR

Short Syntax: DHCP.246 Number of pool size is *num_of_addresses*, number of actual addrs is *valid_addresses*

Long Syntax: DHCP.246 Number of pool size is *num_of_addresses*, number of actual addresses is *valid_addresses*

Description: The number which indicates the pool size is bigger then the actual number of address when finish initializing each one

---

DHCP.247

Level: UI-ERROR

Short Syntax: DHCP.247 Cannot allocate buffer for *num_of_client_records* client records, size *client_record_size*

Long Syntax: DHCP.247 Cannot allocate buffer for *num_of_client_records* client records, size *client_record_size*

Description: Cannot allocate buffer for client records

DHCP.248

Level: UI-ERROR

Short Syntax: DHCP.248 Failed to alloc buffer for addr pool of *num_of_addresses* address records

Long Syntax: DHCP.248 Failed to allocate buffer for addr pool of *num_of_addresses* address records

Description: Failed to allocate buffer for address pool of number of address records

---

DHCP.249

Level: UI-ERROR

Short Syntax: DHCP.249 Cannot *action* address records from profile

Long Syntax: DHCP.249 Cannot *action* address records from profile

Description: Cannot fill or classify address records from profile

---

DHCP.250

Level: UI-ERROR

Short Syntax: DHCP.250 Network tree has *records* addrs while *numRecords* recs were asked to fill

Long Syntax: DHCP.250 Network tree found *records* addresses while *numRecords* records were asked to fill

Description: Initialization routine asks to fill in different number of records than the number of address found in subnet tree

---

DHCP.251

Level: UI-ERROR

Short Syntax: DHCP.251 No subnet specified thru config, no addr record can be initialized

Long Syntax: DHCP.251 No subnet specified thru configuration, no address record can be initialized

Description: No subnet specified through configuration, no address record can be initialized

---

DHCP.252

Level: C-INFO

Short Syntax: DHCP.252 No subnet specified thru config; number of IP addresses is 0

Long Syntax: DHCP.252 No subnet specified thru config; number of IP addresses is 0

Description: The routine to count the number of addresses returns 0 since no subnet is specified in the configuration

---

DHCP.253

Level: UI-ERROR

Short Syntax: DHCP.253 Failed to initialize the address mapper, cannot proceed

Long Syntax: DHCP.253 Failed to initialize the address mapper, cannot proceed

Description: Initialization process of dhcp fails in init address

---

DHCP.254

Level: C-INFO

Short Syntax: DHCP.254 Client address index *client_address_index*

Long Syntax: DHCP.254 Client address index *client_address_index*

Description: Shows the specified address index which is mapped to a client

---

DHCP.255

Level: C-INFO

Short Syntax: DHCP.255 No client with status *status* is available, *action*

Long Syntax: DHCP.255 No client with status *status* is available, *action*

Description: Looking for a free client and record, and it searches through free ones then released ones then expired ones

---

DHCP.256

Level: C-INFO

Short Syntax: DHCP.256 Client requested address lease from wrong server - REPLY NAK

Long Syntax: DHCP.256 Client requested address lease from wrong server - REPLY NAK

Description: Client requested address lease from wrong server - REPLY NAK

DHCP.257

Level: C-INFO

Short Syntax: DHCP.257 failed to address client *client_id*

Long Syntax: DHCP.257 failed to address client *client_id*

Description: Failed to address the specified client

DHCP.258

Level: C-INFO

Short Syntax: DHCP.258 Cannot find client record *client_string* to release it's lease

Long Syntax: DHCP.258 Cannot find client record *client_string* to release it's lease

Description: Received command from address manager to release a lease for a client which does not exist in the record

---

DHCP.259

Level: C-INFO

Short Syntax: DHCP.259 *position*: find next *find_next*, cntl: *cntl_number*

Long Syntax: DHCP.259 *position*: find next *find_next*, cntl: *cntl_number*

Description: Displays the value of find_next and cntl

number which is used for balance and inorder processing

DHCP.260

Level: C-INFO

Short Syntax: DHCP.260 using first label in list cntl: *cntl_number*

Long Syntax: DHCP.260 using first label in list cntl: *cntl_number*

Description: Using the first label member in the list for balance or inorder, display control number

---

DHCP.261

Level: C-INFO

Short Syntax: DHCP.261 returning [0x *return_subnet*], cntl: *cntl_number*

Long Syntax: DHCP.261 returning [0x *return_subnet*], cntl: *cntl_number*

Description: Displays the returning subnet and cntl_number

DHCP.262

Level: C-INFO

Short Syntax: DHCP.262 Using first label in list, subnet is *return_subnet*

Long Syntax: DHCP.262 Using first label in list, subnet is *return_subnet*

Description: Display the subnet number of the first label in list

---

DHCP.263

Level: C-INFO

Short Syntax: DHCP.263 Using subnet from list *return_subnet* [0x *return_subnet_hex*] cntl: *cntl_number*

Long Syntax: DHCP.263 Using subnet from list *return_subnet* [0x *return_subnet_hex*] cntl: *cntl_number*

Description: Displays the subnet which is being used

DHCP.264

Level: C-INFO

Short Syntax: DHCP.264 Switching back to 1st subnet *return_subnet* [0x *return_subnet_hex*] cntl: *cntl_number*

Long Syntax: DHCP.264 Switching back to first subnet *return_subnet* [0x *return_subnet_hex*] cntl: *cntl_number*

Description: Switching back to the first subnet

DHCP.265

Level: C-INFO

Short Syntax: DHCP.265 returning twiddled subnet [0x *twiddled_subnet*]

Long Syntax: DHCP.265 returning twiddled subnet [0x *twiddled_subnet*]

Description: Returning the twiddled subnet

DHCP.266

Level: C-INFO

Short Syntax: DHCP.266 MSSW client of subnet *mssw_subnet_ip*

Long Syntax: DHCP.266 MSSW client of subnet *mssw_subnet_ip*

Description: Display subnet number of client on a mssw network

---

DHCP.267

Level: C-INFO

Short Syntax: DHCP.267 Check if address is in class returns *return_code*

Long Syntax: DHCP.267 Return code from routine check if an address is in a class returns *return_code*

Description: Showing return code from a routine which check through all the class to see if an address is defined within that class

---

DHCP.268

Level: C-INFO

Short Syntax: DHCP.268 returning subnet_to_use 0x *subnet_to_use*, cntl_no *cntl_number*

Long Syntax: DHCP.268 returning subnet_to_use 0x *subnet_to_use*, cntl_no *cntl_number*

Description: Shows the specified return values

DHCP.269

Level: C-INFO

Short Syntax: DHCP.269 No address is available for client (unknown error) - no reply

Long Syntax: DHCP.269 No address is available for client (unknown error) - no reply

Description: No address is available for client (unknown error) - no reply

---

DHCP.270

Level: C-INFO

Short Syntax: DHCP.270 Renewing a lease on another server - NO REPLY

Long Syntax: DHCP.270 Renewing a lease on another server - NO REPLY

Description: Renewing a lease on another server - NO REPLY

DHCP.271

Level: C-INFO

Short Syntax: DHCP.271 Call to pr_query_subnetlabel returned subnet 0x *mssw_subnet*

Long Syntax: DHCP.271 Call to pr_query_subnetlabel returned subnet 0x *mssw_subnet*

Description: Shows the specified return values

DHCP.272

Level: C-INFO

Short Syntax: DHCP.272 Not used

Long Syntax: DHCP.272 Not used

Description: Not used

DHCP.273

Level: C-INFO

Short Syntax: DHCP.273 Not used

Long Syntax: DHCP.273 Not used

Description: Not used

DHCP.274

Level: C-INFO

Short Syntax: DHCP.274 Override srvr lease (*recommended_lease_duration* secs) w/ infinite lease for bootp clnts

Long Syntax: DHCP.274 Server is overriding the server recommended lease (*recommended_lease_duration* seconds) with an INFINITE lease since BOOTP clients are supported

Description: Server is overriding the specified server recommended lease with an INFINITE lease since BOOTP clients are supported

DHCP.275

Level: C-INFO

Short Syntax: DHCP.275 Clnt req INFINITE lease instead of srvr rec lease (*recommended_lease_duration* seconds), this will only be honored if BOOTP clnts are supported

Long Syntax: DHCP.275 Client requested INFINITE lease instead of the server recommended lease (*recommended_lease_duration* seconds), this will only be honored if BOOTP clients are supported

Description: A client has requested an INFINITE lease instead of the server recommended lease. This will only be honored if BOOTP clients are supported.

DHCP.276

Level: C-INFO

Short Syntax: DHCP.276 stall exchange block with state (*exchange_block_state*)

Long Syntax: DHCP.276 stall exchange block with state (*exchange_block_state*)

Description: The specified exchange block is stalled

DHCP.277

Level: C-INFO

Short Syntax: DHCP.277 Not used

Long Syntax: DHCP.277 Not used

Description: Not used

DHCP.278

Level: C-INFO

Short Syntax: DHCP.278 DHCP client *binder_client_id* became BOOTP

Long Syntax: DHCP.278 DHCP client *binder_client_id* became BOOTP

Description: The specified DHCP client has become a BOOTP client

DHCP.279

Level: C-INFO

Short Syntax: DHCP.279 About to generate a reply with Unknown reply type *reply_type*

Long Syntax: DHCP.279 About to generate a reply with Unknown reply type *reply_type*

Description: Displays the value of an unknown reply type

DHCP.280

Level: C-INFO

Short Syntax: DHCP.280 Using subnet from list *return_subnet* (twiddled) cntl: *twiddled_cntl_number*

Long Syntax: DHCP.280 Using subnet from list

return_subnet (twiddled) cntl: *twiddled_cntl_number*

Description: Displays the twiddled subnet which is being used

---

DHCP.281

Level: C-INFO

Short Syntax: DHCP.281 Address *ipaddress* for client *client_string* has just expired

Long Syntax: DHCP.281 Address *ipaddress* for client *client_string* has just expired

Description: Address mapped to the client has just expired

DHCP.282

Level: C-INFO

Short Syntax: DHCP.282 Cannot locate mapped client for address *ipaddress*

Long Syntax: DHCP.282 Cannot locate mapped client for address *ipaddress*

Description: Cannot locate mapped client for address

DHCP.283

Level: UI-ERROR

Short Syntax: DHCP.283 Not enough mem to alloc for hostname or domainname str

Long Syntax: DHCP.283 Not enough memory to allocate for host name or domainname string

Description: There is not enough memory in the box to allocate for hostname or domainname string

---

DHCP.284

Level: C-INFO

Short Syntax: DHCP.284 No *type* specified

Long Syntax: DHCP.284 No *type* specified in requested packet

Description: No domainname or hostname specified, no need to proceed to check dns update

---

DHCP.285

Level: UE-ERROR

Short Syntax: DHCP.285 Invalid hostname [*hostname*] or domainname [*domainname*] entered, no update

Long Syntax: DHCP.285 Invalid hostname [*hostname*] or domainname [*domainname*] is entered, ignore the update

Description: Invalid hostname or domainname has been given by users, ignore furthur processing

---

DHCP.286

Level: UE-ERROR

Short Syntax: DHCP.286 Option 81 DHCP-DDNS specified with len (*length*) less than 4 bytes

Long Syntax: DHCP.286 Option 81 DHCP-DDNS is specified with length less than 4 bytes, length = *length*

Description: Option DHCP-DDNS in the packet is specified with length less than 4 bytes

---

DHCP.287

Level: UE-ERROR

Short Syntax: DHCP.287 Invalid char detected in *typeofname* name, char = [*character*]

Long Syntax: DHCP.287 Invalid character detected in *typeofname* name, char = [*character*]

Description: hostname or domainname has invalid character in it

---

DHCP.288

Level: UE-ERROR

Short Syntax: DHCP.288 *typeofname* name exceeds 63 char, name = [*name*]

Long Syntax: DHCP.288 *typeofname* name exceeds 63 characters, name = [*name*]

Description: hostname or domainname exceeds 63 characters

DHCP.289

Level: C-INFO

Short Syntax: DHCP.289 Pinging addr *ipaddress* (*am_addr_status*)

Long Syntax: DHCP.289 Pinging addr *ipaddress* (status = *am_addr_status*)

Description: Before giving out the address, the server checks the address record status and see if it should be pinged

---

DHCP.290

Level: C-INFO

Short Syntax: DHCP.290 Addr to be pinged *ipaddress* is not in server profile

Long Syntax: DHCP.290 Address to be pinged *ipaddress* does not belong to server profile

Description: Attempt to look up the status of the address to be pinged but the address is not in the server profile

DHCP.291

Level: C-INFO

Short Syntax: DHCP.291 Ping processing completes (*rping_return_code*), *ipaddress* is *isuse*

Long Syntax: DHCP.291 Ping processing completes (*rping_return_code*), address *ipaddress* is *isuse*

Description: Ping return code when checking if an address is in use by other station. The return code indicates if an address is in use or not

---

DHCP.292

Level: C-INFO

Short Syntax: DHCP.292 Found a match for subnet group name [*mssw_label*] in *string* list

Long Syntax: DHCP.292 A subnet with a group name of [*mssw_label*] has been found to be configured within the *string* list

Description: Found a match on the group name of a particular subnet within the BALANCE or INORDER list.

DHCP.293

Level: C-INFO

Short Syntax: DHCP.293 No match for subn gr name [*mssw_label*] in BALANCE or INORDER list

Long Syntax: DHCP.293 A subnet with a group name of [*mssw_label*] has not been found to be configured within the BALANCE or INORDER list

Description: Did not find a match on the group name of a particular subnet within the BALANCE or INORDER list.

DHCP.294

Level: UI-ERROR

Short Syntax: DHCP.294 Could not transmit DHCP to *dest*

Long Syntax: DHCP.294 A DHCP packet could not be transmitted to destination *dest*

Description: Cannot transmit packet to specified destination. This may be the result of improper IP configuration on the DHCP interface.

Chapter 30. Dials (DIAL)

This chapter describes Dials messages. For information on message content and how to use the message, refer to the Introduction.

DIAL.001

Level: C-TRACE

Short Syntax: DIAL.001 CML X31 DSIO: pkt xmted nt *network ID*

Long Syntax: DIAL.001 CML X31 DSIO transmitted a packet on network *network ID*

Description: Trace message for outgoing x.25 packet on a dial circuit over ISDN D-channel

DIAL.002

Level: C-TRACE

Short Syntax: DIAL.002 CML X31 RCV: pkt rcved nt *network ID*

Long Syntax: DIAL.002 CML X31 RCV received a packet on network *network ID*

Description: Trace message for incoming x.25 packet on a dial circuit over ISDN D-channel

DIAL.003

Level: UI-ERROR

Short Syntax: DIAL.003 No cnfg nt *network ID*

Long Syntax: DIAL.003 No configuration found for net *network ID*

Description: No SR_VRTBLK record found in SR_VNET block.

Cause: Incomplete configuration

Action: Review your configuration for this network.

DIAL.004

Level: UI-ERROR

Short Syntax: DIAL.004 bd dl net on nt *network ID*

Long Syntax: DIAL.004 Bad dial network specified in config, net *network ID*

Description: The base net configured is either not present, or not an ISDN BRI net.

Cause: Configuration error.

Action: Configure a valid base net.

DIAL.005

Level: U-INFO

Short Syntax: DIAL.005 Caller id *clid* match found in Authentication list, but no ANY_INBOUND found

Long Syntax: DIAL.005 Caller Id *clid* match found on Authentication list but no net with ANY_INBOUND configured

Description: Checking Callerid for entries in callback table

Cause: Q931 setup received, match clid with entry in auth list, but no any inbound was available

Action: None

DIAL.009

Level: UI-ERROR

Description: There is a software problem.

Cause: software error.

Action: Contact support.

DIAL.010

Level: UI-ERROR

Short Syntax: DIAL.010 X.31 TEI mismatch: rcv= *rcvTEI*,cfg/negot= *cfg_ngotTEI* on nt int /

Long Syntax: DIAL.010 X.31 TEI mismatch: received tei= *rcvTEI*,configured or negotiated tei= *cfg_ngotTEI* on net interface /

Description: The dial circuit is misconfigured.

Cause: Configuration error.

Action: Review your configuration for this dial circuit.

DIAL.011

Level: C-TRACE

Short Syntax: DIAL.011 CML state *state_string*,, event *event_string*, nt *network ID*

Long Syntax: DIAL.011 CML state *state_string*,, event *event_string*,, net *network ID*

Description: FSM trace event.

DIAL.012

Level: UI-ERROR

Short Syntax: DIAL.012 X.31 bad TEI state: tei= *rcvTEI* on nt int /

Long Syntax: DIAL.012 X.31 TEI state is not multi frame for tei= *rcvTEI*, on net interface /

Description: The dial circuit's self test is not successful.

Cause: Network/Configuration error.

Action: Review your configuration for this dial circuit.

DIAL.013

Level: U-INFO

Short Syntax: DIAL.013 Query Caller id table for *clid* from ISDN/ *interface*

Long Syntax: DIAL.013 A setup was recived with Calling Party number *clid* on interface *interface*

Description: Checking Callerid for entries in callback table

Cause: Q931 setup received

Action: None

DIAL.014

Level: U-INFO

Short Syntax: DIAL.014 Found Caller id match for *clid* for callback on net *interface*

Long Syntax: DIAL.014 Caller Id matched in the table *clid* for dial circuit interface *interface*

Description: Clid match callback that destination

Cause: Q931 setup received

Action: None

DIAL.016

Level: U-INFO

Short Syntax: DIAL.016 No match found Caller id *clid* for callback on specific net

Long Syntax: DIAL.016 Caller Id not matched in the table *clid* for dial circuit interface

Description: No Clid match to callback that destination

Cause: Q931 setup received

Action: None

DIAL.017

Level: U-INFO

Short Syntax: DIAL.017 Caller id *clid* no match found in Authentication list,

Long Syntax: DIAL.017 Caller Id *clid* no match found on Authentication list to callback

Description: Checking Callerid for entries in callback table

Cause: Q931 setup received

Action: None

DIAL.018

Level: U-INFO

Short Syntax: DIAL.018 Caller id callback on net *interface*

Long Syntax: DIAL.018 Caller Id callback on interface *interface*

Description: callback the destination

Cause: Callback Timer expired

Action: None

DIAL.019

Level: U-INFO

Short Syntax: DIAL.019 Caller id callback on any_inbound net *interface*

Long Syntax: DIAL.019 Caller Id callback on any inbound interface *interface*

Description: callback the destination

Cause: Callback Timer expired

Action: None

DIAL.020

Level: U-INFO

Short Syntax: DIAL.020 Caller id *clid* - Call blocked on net ISDN/ *net*

Long Syntax: DIAL.020 Caller Id *clid* match found on call block table for net *net*

Description: Checking Callerid for entries in callblock table

Cause: Q931 setup received

Action: None

DIAL.021

Level: C-TRACE

Short Syntax: DIAL.021 CML state *state_string,*, event *event_string,* nt *network ID*

Long Syntax: DIAL.021 CML state *state_string,*, event *event_string,*, net *network ID*

Description: FSM trace event.

DIAL.022

Level: UI_ERROR

Short Syntax: DIAL.022 LID no bf, *message_type,* not snt nt *network ID*

Long Syntax: DIAL.022 LID no buffer, *message_type,* msg not sent on net *network ID*

Description: Line ID code couldn't allocate a buffer to send a message.

DIAL.023

Level: UE_ERROR

Short Syntax: DIAL.023 LID NAK rcv nt *network ID*

Long Syntax: DIAL.023 LID NAK received net *network ID*

Description: The other end of the switched circuit didn't like the LINE ID we sent, and returned a NAK.

Action: Check configuration on both sides. Remote side does not think we should be calling it.

DIAL.024

Level: C-INFO

Short Syntax: DIAL.024 LID ACK rcv nt *network ID*

Long Syntax: DIAL.024 LID ACK received net *network ID*

Description: The other end of the switched circuit liked our line ID.

DIAL.025

Level: UE_ERROR

Short Syntax: DIAL.025 LID tmo on mdm sgs nt *network ID*

Long Syntax: DIAL.025 LID timeout waiting for modem signals to come up on net *network ID*

Description: Either an inbound or outbound call, the V.25bis modem signals did not come up after the call was connected.

Action: Check line and modems. Line quality may be insufficient.

DIAL.026

Level: UE_ERROR

Short Syntax: DIAL.026 LID tmo on id nt *network ID*

Long Syntax: DIAL.026 LID timeout waiting for line ID from other side, net *network ID*

Description: Timed out waiting for line ID from remote side.

Action: Check configuration of whoever is calling into this router. They are not sending line ID message. Might be an incompatible router.

DIAL.027

Level: UE_ERROR

Short Syntax: DIAL.027 LID unkn id [*bad_lineid_string,*]; nk snt, nt *network ID*

Long Syntax: DIAL.027 LID unknown line ID [*bad_lineid_string,*] received; NAK sent, net *network ID*

Description: An ID message was received corresponding to a phone number from which you do not want any calls, that is, a phone number that does not exist, or the number exists but is configured for no inbound calls.

Action: Check configuration of both routers.

DIAL.028

Level: UE_ERROR

Short Syntax: DIAL.028 LID no dflt circt; data ign nt *network ID*

Long Syntax: DIAL.028 LID no default circuit; received data was ignored, net *network ID*

Description: Received data from other side rather than line ID, but had no default circuit to assign the data to.

Action: Check configuration of whoever is calling into this router. They are not sending line ID message. Might be an incompatible router.

DIAL.029

Level: C-INFO

Short Syntax: DIAL.029 No dl crct inc call on nt *switched network ID*

Long Syntax: DIAL.029 No dial circuit configured for inbound calls on switched network *switched network ID*

Description: An inbound call was received over the switched network, and there isn't a dial circuit configured to take it.

Cause: Misconfiguration.

Action: A dial circuit needs to be configured to accept inbound calls.

Cause: Wrong number.

Action: If this persists, you may want to pursue what avenues you can to identify a possible security break-in.

DIAL.030

Level: C-TRACE

Short Syntax: DIAL.030 nt *dial network ID* st *cml_state*; cnt acpt call on nt *switched network ID*

Long Syntax: DIAL.030 net *dial network ID* is in state *cml_state*; can't acpt call on network *switched network ID*

Description: A dial circuit was found that would take the incoming call, but it is not in a state where it can do so.

DIAL.031

Level: C-TRACE

Short Syntax: DIAL.031 Inbnd dsbl nt *dial network ID*; cnt acpt call on nt *switched network ID*

Long Syntax: DIAL.031 Inbound calls disabled on net *dial network ID*; can't acpt call on network *switched network ID*

Description: The network would accept a call from a specified caller, but it is configured not to accept inbound calls.

DIAL.032

Level: C-TRACE

Short Syntax: DIAL.032 LID st *old_state*,-> *new_state*, nt *network ID*

Long Syntax: DIAL.032 Line ID state *old_state*, changed to *new_state*,, net *network ID*

Description: FSM trace event.

DIAL.033

Level: C-TRACE

Short Syntax: DIAL.033 LID ID rcv: *line_id_string* nt *network ID*

Long Syntax: DIAL.033 Line ID received: *line_id_string*, net *network ID*

Description: A Line ID message was received containing the specified address. Note: Only the digits 0-9 are printed, since only they are significant.

DIAL.034

Level: C-TRACE

Short Syntax: DIAL.034 nt *dial network ID* acptd call on nt *switched network ID*

Long Syntax: DIAL.034 net *dial network ID* accepted call on network *switched network ID*

Description: The specified network has accepted the inbound call.

DIAL.035

Level: C-TRACE

Short Syntax: DIAL.035 No avl net for inbound call on nt *switched network ID*

Long Syntax: DIAL.035 No available net for call on network *switched network ID*

Description: There is no network that can take the inbound call.

DIAL.036

Level: C-TRACE

Short Syntax: DIAL.036 ISDN inb Caller Id addr [*address*] nt *switched network ID*

Long Syntax: DIAL.036 ISDN inbound address [*address*] network *switched network ID*

Description: The router passed the specified address and subaddress of the caller in an ISDN setup message.

DIAL.037

Level: C-TRACE

Short Syntax: DIAL.037 LID ID snt: *line_id_string* nt *network ID*

Long Syntax: DIAL.037 Line ID sent: *line_id_string*, net *network ID*

Description: We sent the specified line ID message to the destination.

DIAL.038

Level: C-INFO

Short Syntax: DIAL.038 Too many circuits nt *base network ID*

Long Syntax: DIAL.038 Too many circuits on net *base network ID*

Description: There are more virtual circuits that are active than the interface type supports.

DIAL.039

Level: C-INFO

Short Syntax: DIAL.039 Higher pri conn nt *preempted network ID* preempts nt *higher-priority network ID*

Long Syntax: DIAL.039 Higher priority connection request for net *preempted network ID* preempts net *higher-priority network ID*

Description: A connection request for a higher-priority dial circuit caused the specified lower-priority circuit to terminate.

DIAL.040

Level: C-INFO

Short Syntax: DIAL.040 Disc ind on pri conn nt *network ID*; retry

Long Syntax: DIAL.040 Disconnect indication received for priority connection network *network ID*; retry

Description: The router received a disconnect indication for the specified network, but the base network did not actually attempt the connection. The router rejected the connection because the base network was not ready. The router will retry the connection shortly.

DIAL.041

Level: C-INFO

Short Syntax: DIAL.041 outbound call denied, configed inbound nt *network ID*

Long Syntax: DIAL.041 Outbound calls denied network *network ID*

Description: Router would like to place outbound call, but configuration prevents it.

DIAL.042

Level: C-INFO

Short Syntax: DIAL.042 idle exp nt *network ID*

Long Syntax: DIAL.042 idle timer expired and call cleared, net *network ID*

Description: The idle timer of a demand-based net expired, and the call was cleared.

DIAL.043

Level: C-TRACE

Short Syntax: DIAL.043 Match dial addr [*dial_address*] to nt *switched network ID*

Long Syntax: DIAL.043 Matched inbound destination dial address [*dial_address*] to network *switched network ID*

Description: An inbound call arrived and the specified network is configured to match it. Match the dial_address address string in hex. Empty string is a wildcard and will match a network with any_inbound setting.

DIAL.044

Level: C-TRACE

Short Syntax: DIAL.044 No usbl match dial addr [*dial_address*]

Long Syntax: DIAL.044 No useable match dial addr [*dial_address*]

Description: No more dial circuits match inbound address.

DIAL.045

Level: C-TRACE

Short Syntax: DIAL.045 Dialing dest < *dest_name*>, DTE number [*dte_addr*], nt *network ID*

Long Syntax: DIAL.045 Dialing destination < *dest_name*>, DTE number [*dte_addr*], net *network ID*

Description: The Connection Management Library (CML) is dialing the specified destination end point using the specified DTE number. This message occurs for every DTE number the CML actually dials.

DIAL.046

Level: C-TRACE

Short Syntax: DIAL.046 CMLB net # *net_num*, dest < *dest*>, indest < *in_dest*>, net *network ID*

Long Syntax: DIAL.046 CMLB dump: net # *net_num*, dest < *dest*>, indest < *in_dest*>, net *network ID*

Description: Traces the contents of the Connection Management Library control Block (CMLB) chain as an inbound connection vector to the correct CMLB.

DIAL.047

Level: C-TRACE

Short Syntax: DIAL.047 Source DTE addr # *index*: [*addr_str*]

Long Syntax: DIAL.047 Source DTE address # *index*: [*addr_str*]

Description: The router called once for each DTE address string found in a CMLB's src_addrs.

DIAL.048

Level: UI-ERROR

Short Syntax: DIAL.048 Bad MP config nt *network ID*

Long Syntax: DIAL.048 Bad MP config for net *network ID*

Description: The MP net configured is invalid or BRS is on the link.

Cause: Configuration error.

Action: Configure a valid MP net or turn off BRS on the link.

DIAL.049

Level: UI-ERROR

Short Syntax: DIAL.049 Invalid destination addr on nt *network ID*

Long Syntax: DIAL.049 Bad dialer destination name specified in config, net *network ID*

Description: The specified destination name was not added with the ″add address″ command.

Cause: Configuration error.

Action: Configure a destination name using the ″add address″ command.

DIAL.050

Level: UI-ERROR

Short Syntax: DIAL.050 Swcthd net (*switched network ID*) rjctd rgstrtn for nt *network ID*

Long Syntax: DIAL.050 The switched network (network *switched network ID*) rejected the registration request for this dial circuit: net *network ID*

Description: The dial circuit is misconfigured.

Cause: Configuration error.

Action: Review your configuration for this dial circuit.

Chapter 31. Dialout (DOUT)

This chapter describes Dialout (DOUT) messages. For information on message content and how to use the message, refer to the Introduction.

DOUT.001

Level: U-INFO

Short Syntax: DOUT.001 Schedule a Listen for TCP open on tcp port *portnum*

Long Syntax: DOUT.001 Schedule a Listen for TCP open on tcp port *portnum*

Description: DIALOUT: For each dialout circuit present, telnet registers with tcp to listen for open requests on tcp port 1000.

DOUT.002

Level: U-INFO

Short Syntax: DOUT.002 A TCP connection to the telnet modem server has been opened

Long Syntax: DOUT.002 A TCP connection to the telnet modem server has been opened

Description: DIALOUT: A TCP connection to the telnet modem server has been opened, next step is to register with a virtual net, if one is available.

DOUT.003

Level: UE-ERROR

Short Syntax: DOUT.003 Initializing telnet queues failed, can't open telnet modem connection

Long Syntax: DOUT.003 Initializing telnet queues failed, can't open telnet modem connection

Description: DIALOUT: tel_qinit() failed. The telnet modem server tried to initialize the queues associated with this session. This attempt failed as a result of not being able to allocate the queue.

Action: You may be running low on memory, check your memory statistics. Report this error to customer service.

DOUT.004

Level: UE-ERROR

Short Syntax: DOUT.004 Telnet read buffer allocation failed, can't open telnet modem conn

Long Syntax: DOUT.004 Telnet read buffer allocation failed, can't open telnet modem conn

Description: DIALOUT: tel_start_init failed as a result of the read buffer not being able to be allocated.

Action: You may be running low on memory, check your memory statistics. Report this error to customer service.

DOUT.005

Level: UE-ERROR

Short Syntax: DOUT.005 Telnet couldn't register with a virtual net

Long Syntax: DOUT.005 Telnet couldn't register with a virtual net

Description: The telnet modem server could not register with a virtual net. The reason for this is most likely that all the base nets that have dialout circuits configured for them are in use.

Action: Make sure a dialout circuit and its corresponding base net are both available and try again.

DOUT.006

Level: C-INFO

Short Syntax: DOUT.006 Dialout server registered new session with net number *netnum*

Long Syntax: DOUT.006 Dialout server registered new session with net number *netnum*

Description: The new telnet session was successfully registered with a virtual net. All data that arrives to this telnet session will be sent to the virtual net that was listed.

DOUT.010

Level: UE-ERROR

Short Syntax: DOUT.010 Dialout server session closing

Long Syntax: DOUT.010 Dialout server session closing

Description: Dialout server session closed, most likely as a result of not being able to register with a virtual net.

Action: Make sure a dialout circuit and its corresponding base net are both available and try again.

DOUT.011

Level: C-INFO

Short Syntax: DOUT.011 Dialout server session on net *netnum* closing

Long Syntax: DOUT.011 Dialout server session on net *netnum* closing

Description: Dialout server session closed, most likely as a result the client terminating the session.

DOUT.012

Level: C-INFO

Short Syntax: DOUT.012 Dialout server rcvd *count* byte(s) from modem on net *netnum*

Long Syntax: DOUT.012 Dialout server rcvd *count* byte(s) from modem on net *netnum*

Description: Dialout server in data xfer state and received bytes from modem.

DOUT.013

Level: UE-ERROR

Short Syntax: DOUT.013 Dialout server rcvd packet from modem with errors on net *netnum*

Long Syntax: DOUT.013 Dialout server rcvd packet from modem with errors on net *netnum*

Description: Dialout server received bytes from modem and the packet had errors in it.

Action: This error occured from some event in the net handler or driver. In the monitor console, check the error statistics for this net. If the problem persists, report this problem to customet service.

DOUT.014

Level: UE-ERROR

Short Syntax: DOUT.014 Dialout server could not xmit packet, net *netnum* was down

Long Syntax: DOUT.014 Dialout server could not xmit packet, net *netnum* was down

Description: Dialout server tried to transmit bytes from telnet but the v34 handler reported the net was down and did not xmit the bytes.

Action: This error occured from some event in the net handler or driver. In the monitor console, check the error statistics for this net. If the problem persists, report this problem to customet service.

DOUT.015

Level: C-INFO

Short Syntax: DOUT.015 Dialout server received *bytes* data bytes via the *name* interface

Long Syntax: DOUT.015 Dialout server received *bytes* data bytes via the *name* interface

Description: Dialout server received x number of bytes via either the telnet or DIALs application.

Action: No action just information

DOUT.016

Level: UE-ERROR

Short Syntax: DOUT.016 Dialout server could not xmit packet, problem with net *netnum*

Long Syntax: DOUT.016 Dialout server could not xmit packet, problem with net *netnum*

Description: Dialout server tried to xmit a packet but could not because of some internal error in the driver.

Action: This error occured from some event in the net handler or driver. In the monitor console, check the error statistics for this net. If the problem persists, report this problem to customer service.

DOUT.017

Level: C-INFO

Short Syntax: DOUT.017 Dialout server received a telnet option for *command option* packet

Long Syntax: DOUT.017 Dialout server received a telnet option for *command option* packet

Description: Dialout server received a telnet command option.

Action: No action, just information

DOUT.018

Level: UE-ERROR

Short Syntax: DOUT.018 Dialout server did not receive an end of Suboption

Long Syntax: DOUT.018 Dialout server did not receive an end of Suboption

Description: Dialout server received a telnet suboption command but never received the suboption end byte.

Action: This is an error that may cause the dialout server to become out of sync. Operation will continue with unexpected results. If this problem persists contact customer service.

DOUT.019

Level: UE-ERROR

Short Syntax: DOUT.019 Dialout server, net *netnum*, failure during CML init

Long Syntax: DOUT.019 Dialout server, net *netnum*, failure during CML init

Description: Dialout server did not install correctly due to a error during cml_init. This is most likely a configuration problem.

Action: Please check to see that the configuration is correct. If you feel everything is configured correctly, please contact customer service.

DOUT.020

Level: UE-ERROR

Short Syntax: DOUT.020 Dialout server *timer_type* timer expired, net *netnum* going down

Long Syntax: DOUT.020 Dialout server *timer_type* timer expired, net *netnum* going down

Description: Either the keepalive timer or the inactivity timer on the dialout server expired. If the inactivity timer expired, this means that data has not been transmitted or received within the configured amount of time. If the keepalive timer has expired this means that the dialout client has not sent a keepalive packet in the timeout period of four minutes. They are supposed to be sent every 2 minutes. Please check to make sure the Shiva client is operating correctly.

Action: Increase the inactivity timer if this is causing a problem for your clients. For keepalive timer expirations, please make sure the client is operating correctly. If the problem persists, please contact IBM customer service.

DOUT.021

Level: C-INFO

Short Syntax: DOUT.021 Dialout server received keepalive pkt on net *netnum*

Long Syntax: DOUT.021 Dialout server received keepalive pkt on net *netnum*

Description: Information - received a keepalive packet on network interface.

Action: No action, just information

DOUT.022

Level: C_INFO

Short Syntax: DOUT.022 *buffer*

Long Syntax: DOUT.022 *buffer*

Description: Information - if you are having problems, report message to IBM Customer Service.

Action: For problems, report this message to customer service.

DOUT.023

Level: CE_ERROR

Short Syntax: DOUT.023 Dialout server could not xmt *numbytes* bytes from modem to TCP on net *netnum*

Long Syntax: DOUT.023 Dialout server could not xmt *numbytes* bytes from modem to TCP on net *netnum*

Description: Error - Dialout server could not transmit bytes received from modem to TCP. The reason for this is that TCP buffers are full and as a result the dialout server cannot put any more data into this buffer. This most likely is a result of a slow dialout client or network congestion, or a heavy loaded router. Contact customer service for more help.

Action: For problems, report this message to customer service.

DOUT.024

Level: UE_ERROR

Short Syntax: DOUT.024 Could not add modem pool tcp session, maximum number of *num* exceeded

Long Syntax: DOUT.024 Could not add modem pool tcp session, maximum number of *num* exceeded

Description: Error - User has added more dialout nets than are allowed. This error will not hurt anything, however only MAXTNMDMS can be utilized. This number is limited to the number of physical modems in the device. For VL3 platforms, this number is 12, for VL platforms, this number is 2.

Action: Delete excess dialout nets

Chapter 32. Digital Modem Adapter

This chapter describes Digital Modem Adapter messages. For information on message content and how to use the message, refer to the Introduction.

MDM.001

Level: C-INFO

Short Syntax: MDM.001 *indication* indication from modem on port *port* chan *channel* nt *network ID*

Long Syntax: MDM.001 *indication* indication from modem on port *port* channel *channel* net *network ID*

Description: The digital modem adapter received a terse indication from the modem.

MDM.002

Level: C-INFO

Short Syntax: MDM.002 Init strng *init_string* rcvd response OK frm TRIPAC *port* DSP *channel* nt *network ID*

Long Syntax: MDM.002 Init string *init_string* received a response of OK from TRIPAC *port* DSP *channel* net *network ID*

Description: The digital modem adapter received the indication shown from the modem.

MDM.003

Level: C-INFO

Short Syntax: MDM.003 CONNECT from modem on port *port* chan *channel* nt *network ID*

Long Syntax: MDM.003 CONNECT from modem on port *port* channel *channel* net *network ID*

Description: The digital modem adapter received the indication shown from the modem.

MDM.004

Level: C-INFO

Short Syntax: MDM.004 RING from modem on port *port* chan *channel* nt *network ID*

Long Syntax: MDM.004 RING from modem on port *port* channel *channel* net *network ID*

Description: The digital modem adapter received the indication shown from the modem.

MDM.005

Level: C-INFO

Short Syntax: MDM.005 NO CARRIER from modem on port *port* chan *channel* nt *network ID*

Long Syntax: MDM.005 NO CARRIER from modem on port *port* channel *channel* net *network ID*

Description: The digital modem adapter received the indication shown from the modem.

MDM.006

Level: C-INFO

Short Syntax: MDM.006 Init string *init_string* received a response of ERROR from modem on tripac *port* dsp *channel* nt *network ID*

Long Syntax: MDM.006 Init string *init_string* received a response of ERROR from modem on tripac *port* dsp *channel* net *network ID*

Description: The digital modem adapter received the indication shown from the modem.

MDM.007

Level: C-INFO

Short Syntax: MDM.007 NO DIALTONE from modem on port *port* chan *channel* nt *network ID*

Long Syntax: MDM.007 NO DIALTONE from modem on port *port* channel *channel* net *network ID*

Description: The digital modem adapter received the indication shown from the modem.

MDM.008

Level: C-INFO

Short Syntax: MDM.008 BUSY from modem on port *port* chan *channel* nt *network ID*

Long Syntax: MDM.008 BUSY from modem on port *port* channel *channel* net *network ID*

Description: The digital modem adapter received the indication shown from the modem.

MDM.009

Level: C-INFO

Short Syntax: MDM.009 NO ANSWER from modem on port *port* chan *channel* nt *network ID*

Long Syntax: MDM.009 NO ANSWER from modem on port *port* channel *channel* net *network ID*

Description: The digital modem adapter received the indication shown from the modem.

MDM.010

Level: C-INFO

Short Syntax: MDM.010 CARRIER *baud* from modem on port *port* chan *channel* nt *network ID*

Long Syntax: MDM.010 CARRIER *baud* from modem on port *port* channel *channel* net *network ID*

Description: The digital modem adapter received the indication shown from the modem.

MDM.011

Level: C-INFO

Short Syntax: MDM.011 COMPRESSION: CLASS 5 from modem on port *port* chan *channel* nt *network ID*

Long Syntax: MDM.011 COMPRESSION: CLASS 5 from modem on port *port* channel *channel* net *network ID*

Description: The digital modem adapter received the indication shown from the modem.

MDM.012

Level: C-INFO

Short Syntax: MDM.012 COMPRESSION: V.42 bis from modem on port *port* chan *channel* nt *network ID*

Long Syntax: MDM.012 COMPRESSION: V.42 bis from modem on port *port* channel *channel* net *network ID*

Description: The digital modem adapter received the indication shown from the modem.

MDM.013

Level: C-INFO

Short Syntax: MDM.013 COMPRESSION: NONE from modem on port *port* chan *channel* nt *network ID*

Long Syntax: MDM.013 COMPRESSION: NONE from modem on port *port* channel *channel* net *network ID*

Description: The digital modem adapter received the indication shown from the modem.

MDM.014

Level: C-INFO

Short Syntax: MDM.014 PROTOCOL: NONE from modem on port *port* chan *channel* nt *network ID*

Long Syntax: MDM.014 PROTOCOL: NONE from modem on port *port* channel *channel* net *network ID*

Description: The digital modem adapter received the indication shown from the modem.

MDM.015

Level: C-INFO

Short Syntax: MDM.015 PROTOCOL: LAPM from modem on port *port* chan *channel* nt *network ID*

Long Syntax: MDM.015 PROTOCOL: LAPM from modem on port *port* channel *channel* net *network ID*

Description: The digital modem adapter received the indication shown from the modem.

MDM.016

Level: C-INFO

Short Syntax: MDM.016 Digital modem adapter load starting for slot *slot*.

Long Syntax: MDM.016 Digital modem adapter load starting for slot *slot*.

Description: The system is starting to load the digital modem adapter in specified slot.

MDM.017

Level: C-INFO

Short Syntax: MDM.017 Digital modem adapter load completed successfully for slot *slot*.

Long Syntax: MDM.017 Digital modem adapter load completed successfully for slot *slot*.

Description: The system successfully completed loading the adapter in the specified slot.

MDM.018

Level: ALWAYS

Short Syntax: MDM.018 Digital modem adapter load failed for slot *slot*, reason

Long Syntax: MDM.018 Digital modem adapter load failed for slot *slot*, reason

Description: The adapter in the specified slot failed to load.

MDM.019

Level: C-INFO

Short Syntax: MDM.019 No modems available to accept the call for port *port* chan *channel* nt *network ID*

Long Syntax: MDM.019 No modems available to accept the call for port *port* chan *channel* nt *network ID*

Description: The call could not be accepted since there were no modems available.

MDM.020

Level: ALWAYS

Short Syntax: MDM.020 Error: Dbl rcv buf rtn to dgtl mdm adap slot *slot* chan *channel* nt *network ID*

Long Syntax: MDM.020 Double receive buffer return to digital modem adapter slot *slot* channel *channel* net *network ID*

Description: A protocol returned a private receive buffer twice (an error). This is generally a code problem. This interface is dead until the router is restarted.

MDM.021

Level: C-INFO

Short Syntax: MDM.021 CONNECT str *string* rcvd from mdm *modem* on port *port* chan *channel* nt *network ID*

Long Syntax: MDM.021 CONNECT string *string* received from modem *modem* on port *port* channel *channel* net *network ID*

Description: Connect string received from Tripac when modems finish training.

Chapter 33. Digital Network Architecture Phase IV (DN)

This chapter describes Digital Network Architecture Phase IV (DN) messages. For information on message content and how to use the message, refer to the Introduction.

DN.001

Level: UE-ERROR

Short Syntax: DN.001 event 4.0: Aged pkt loss; *source_area*. *source_node* -> *destination_area*. *destination_node*

Long Syntax: DN.001 event 4.0: Aged packet loss; packet from *source_area*. *source_node* to *destination_area*. *destination_node*

Description: A packet has had too many visits through routers going between the specified nodes. If return to sender was requested, the packet will be returned to the originator. Otherwise, it will be dropped.

Cause: The router's EXECUTOR MAXIMUM VISITS is too small.

Action: Increase EXECUTOR MAXIMUM VISITS to be larger the number of hops between the two most distant nodes in the network.

Cause: There is a temporary routing loop due to an unreachable node.

Action: Unless the problem is persistent, there should be no need for corrective action. Routing loops usually go away within a minute when a node goes down.

DN.002

Level: CE-ERROR

Short Syntax: DN.002 event 4.1: Node unreach pkt loss; *source_area*. *source_node* -> *destination_area*. *destination_node*, cir *number* net *network_name*

Long Syntax: DN.002 event 4.1: Node unreachable packet loss; packet from *source_area*. *source_node* to *destination_area*. *destination_node*, circuit *number* network *network_name*

Description: Packet was received on the specified network for unreachable destination. If return to sender was requested, the packet will be returned to the originator. Otherwise, it will be dropped.

Cause: The originator is attempting to contact a non-existent node.

Action: If the originator supplied a host address, it should be corrected. If the originator supplied a host name, the node name to the address translation may be out of date. Use the DEFINE NODE ″name″ ADDRESS command on the originating node to correct the permanent database.

Cause: There is no route to the destination node in the routing database.

Action: Do a SHOW ACTIVE NODES to see if the destination node is reachable. Check the circuit(s) that could be used to reach this node.

Cause: There is no route to the destination area in the routing database.

Action: Do a SHOW ACTIVE AREA to see if the area of the destination node is reachable. Check the circuit(s) that could be used to reach this area.

DN.003

Level: UI-ERROR

Short Syntax: DN.003 event 4.2: Node out-of-range pkt loss; *source_area*. *source_node* -> *destination_area*. *destination_node*, cir *number* net *network_name*

Long Syntax: DN.003 event 4.2: Node out-of-range packet loss; packet from *source_area*. *source_node* to *destination_area*. *destination_node*, circuit *number* network *network_name*

Description: Packet was received on the specified network for node address beyond EXECUTOR MAXIMUM ADDRESS. If return to sender was requested, the packet will be returned to the originator. Otherwise, it will be dropped.

Cause: EXECUTOR MAXIMUM ADDRESS set too low.

Action: Increase EXECUTOR MAXIMUM ADDRESS.

Cause: Destination node's EXECUTOR NODE ADDRESS set too high.

Action: Decrease destination node's EXECUTOR NODE ADDRESS.

Cause: The originator is attempting to contact a non-existent node, which also has too high an address.

Action: If the originator supplied a host address, it should be corrected. If the originator supplied a host name, the node name to address translation may be out of date. Use the DEFINE NODE ″name″ ADDRESS command on the originating node to correct the permanent database.

DN.004

Level: UE-ERROR

Short Syntax: DN.004 event 4.3: Ovsize pkt loss; *source_area*. *source_node* -> *destination_area*. *destination_node*, cir *number* net *network_name*

Long Syntax: DN.004 event 4.3: Oversized packet loss; packet from *source_area*. *source_node* to *destination_area*. *destination_node*, circuit *number* network *network_name*

Description: Packet was received that is larger than the blocksize of the output circuit chosen to the destination. The packet will be dropped.

Cause: Originating host has a larger EXECUTOR BUFFER SIZE than the receiving host can accept.

Action: Correct EXECUTOR BUFFER SIZE on originating host.

Cause: Intervening circuit has too small a packet size.

Action: Ensure that originating host's EXECUTOR BUFFER SIZE is smaller than the circuit with the lowest packet size. (Since Ethernet has the smallest blocksize, this is not likely.)

DN.005

Level: UE-ERROR

Short Syntax: DN.005 event 4.4: Pkt format err; data packet *source_area*. *source_node* -> *destination_area*. *destination_node*, cir *number* net *network_name*

Long Syntax: DN.005 event 4.4: Packet format error; long data packet from *source_area*. *source_node* to *destination_area*. *destination_node*, circuit *number* network *network_name*

Description: A Long Data Packet was received with invalid header data, on the specified circuit. The packet will be dropped.

Cause: First 4 bytes of source or destination ID are not HIORD.

Action: Correct programming error in sending node, or find source of data corruption.

Cause: The reserved D-AREA or S-AREA fields of the long data packet are not zero.

Action: Correct programming error in sending node, or find source of data corruption.

DN.006

Level: UE-ERROR

Short Syntax: DN.006 event 4.4: Pkt format err; endnode hello from *source_area*. *source_node*, cir *number* net *network_name*

Long Syntax: DN.006 event 4.4: Packet format error; endnode hello message from *source_area*. *source_node*, circuit *number* network *network_name*

Description: An Endnode Hello Message was received with invalid header data on the specified circuit. The packet will be dropped.

Cause: The node type in the IINFO field is not endnode, or the first 4 bytes of the ID field are not HIORD.

Action: Correct programming error in sending node, or find source of data corruption.

DN.007

Level: UE-ERROR

Short Syntax: DN.007 event 4.4: Pkt format err; lvl *router_level* route from *source_area*. *source_node*, cir *number* net *network_name*

Long Syntax: DN.007 event 4.4: Packet format error; level *router_level* routing message from *source_area*. *source_node* circuit *number* network *network_name*

Description: A Level 1 or 2 Routing Message was received with a formatting error within the routing data. The packet will be dropped. In the case of an error in the routing data, the data up to the error will be processed.

Cause: The packet ends with a SEGMENT that does not contain as many RTGINFO entries as the COUNT claims.

Action: Correct programming error in sending node, or find source of data corruption.

DN.008

Level: UE-ERROR

Short Syntax: DN.008 event 4.4: Pkt format err; short pkt from *source_area*. *source_node*, cir *number* net *network_name*

Long Syntax: DN.008 event 4.4: Packet format error; packet too short from *source_area*. *source_node*, circuit *number* network *network_name*

Description: A packet too short to contain its header was received. The packet will be dropped.

Cause: Long Data Packet less than 21 bytes long (excluding padding).

Cause: Endnode Hello Message less than 31 bytes long.

Cause: Endnode Hello Message not long enough to contain the test data indicated by the byte count in the test data.

Cause: Router Hello Message less than 27 bytes long.

Cause: Routing Message less than 6 bytes long.

Action: Correct programming error in sending node, or find source of data corruption.

DN.009

Level: UE-ERROR

Short Syntax: DN.009 event 4.4: Pkt format err; router hello from *source_area*. *source_node* cir *number* net *network_name*

Long Syntax: DN.009 event 4.4: Packet format error; router hello message from *source_area*. *source_node* circuit *number* network *network_name*

Description: A Router Hello Message was received with invalid header data. The packet will be dropped.

Cause: The node type in the INFO field is not level 1 or 2 router, or the first 4 bytes of the ID field are not HIORD.

Action: Correct programming error in sending node, or find source of data corruption.

DN.010

Level: UE-ERROR

Short Syntax: DN.010 event 4.4: Pkt format err; unkn typ, cir *number* net *network_name*, hdr *first 21 bytes*

Long Syntax: DN.010 event 4.4: Packet format error; unknown type, circuit *number* network *network_name*, header *first 21 bytes*

Description: A packet with an invalid or unsupported flags field was received. The first 21 bytes of the header are dumped.

Cause: The first byte of the message is not one of Long Data Packet, Endnode Hello, Router Hello, Level 1 Routing, or Level 2 Routing.

Action: Correct programming error in sending node, or find source of data corruption.

DN.012

Level: UE-ERROR

Short Syntax: DN.012 event 4.4: Pkt format err; vers skew, flags *FLAGS*, cir *number* net *network_name*

Long Syntax: DN.012 event 4.4: Packet format error; version skew in long data packet, flags *FLAGS*, circuit *number* network *network_name*

Description: A Long Format Data Packet was received with the version bit set in the flags field. The packet will be dropped.

Cause: Programming error in sending node, or data corruption.

DN.013

Level: CI-ERROR

Short Syntax: DN.013 event 4.5: Part rting upd loss; area *area_number* from *source_area*. *source_node*, cir *number* net *network_name*

Long Syntax: DN.013 event 4.5: Partial routing update loss; area *area_number* in level 2 routing message from *source_area*. *source_node*, circuit *number* network *network_name*

Description: A Level 2 Routing Message contained reachable routes to area(s) higher than this router's EXECUTOR MAXIMUM AREA. Only the highest reachable area will be logged. Routes to unreachable (infinite cost) areas are not complained about.

Cause: This routers EXECUTOR MAXIMUM AREA is lower than the highest reachable area in the network.

Action: Correct EXECUTOR MAXIMUM AREA, or change area number of offending area.

DN.014

Level: CI-ERROR

Short Syntax: DN.014 event 4.5: Part rting upd loss; node *node_number* from *source_area*. *source_node*, cir *number* net *network_name*

Long Syntax: DN.014 event 4.5: Partial routing update loss; node *node_number* in level 1 routing message from *source_area*. *source_node*, circuit *number* network *network_name*

Description: A Level 1 Routing Message contained reachable routes to node(s) higher than this router's EXECUTOR MAXIMUM ADDRESS. Only the highest reachable node will be logged. Routes to unreachable (infinite cost) nodes are not complained about.

Cause: This routers EXECUTOR MAXIMUM ADDRESS is lower than the highest reachable node in the network.

Action: Correct EXECUTOR MAXIMUM ADDRESS, or change node number of offending node.

DN.015

Level: UE-ERROR

Short Syntax: DN.015 event 4.11: Init fail; inval data from *source_area*. *source_node* cir *number* net *network_name*

Long Syntax: DN.015 event 4.11: Initialization failure, line fault; adjacent node listener received invalid data from node *source_area*. *source_node* circuit *number* network *network_name*

Description: The (optional) test data in an Endnode Hello Message was not valid, differing from the

expected test pattern of 252 octal. The adjacency will not be accepted.

Cause: Data corruption on network.

DN.016

Level: UE-ERROR

Short Syntax: DN.016 event 4.13: Init fail; endnode *source_area*. *source_node* out of range, cir *number* net *network_name*

Long Syntax: DN.016 event 4.13: Initialization failure, operator initiated; adjacent endnode *source_area*. *source_node* out of range, circuit *number* network *network_name*

Description: An Endnode Hello Message was received from the specified node, but its node address exceeds the EXECUTOR MAXIMUM ADDRESS. No adjacency will be created.

Cause: Endnode node address too high.

Action: Correct endnode node address.

Cause: Router's EXECUTOR MAXIMUM ADDRESS too low.

Action: Increase router's EXECUTOR MAXIMUM ADDRESS.

DN.017

Level: UE-ERROR

Short Syntax: DN.017 event 4.13: Init fail; router *area*. *node* out of range, cir *number* net *network_name*

Long Syntax: DN.017 event 4.13: Initialization failure, operator initiated; adjacent router *area*. *node* out of range, circuit *number* network *network_name*

Description: A Router Hello Message was received from the specified node, but there is a problem with it's node address. The node address exceeds the EXECUTOR MAXIMUM ADDRESS or the area address exceeds the EXECUTOR MAXIMUM AREA or the node or area number is zero. No adjacency will be created.

Cause: Source router's node address too high.

Action: Correct source router's node address.

Cause: This router's EXECUTOR MAXIMUM ADDRESS too low.

Action: Increase this router's EXECUTOR MAXIMUM ADDRESS.

Cause: Source router's area address too high.

Action: Correct source router's area address.

Cause: This router's EXECUTOR MAXIMUM AREA too low.

Action: Increase this router's EXECUTOR MAXIMUM AREA.

Cause: Remote router using node or area 0.

Action: Correct programming error on remote node.

DN.018

Level: UE-ERROR

Short Syntax: DN.018 event 4.13: Init fail; blck sz *size* too sm frm *area*. *node*, cir *number* net *network_name*

Long Syntax: DN.018 event 4.13: Initialization failure, operator initiated; adjacent node block size *size* too small from router *area*. *node*, circuit *number* network *network_name*

Description: A router hello is offering a blocksize that is too small to support area routing. The blocksize must be large enough to receive a Level 2 Routing Message with all 63 areas in it. The adjacency will be rejected.

Cause: Adjacent router has a block size less than 80.

Action: Correct block size on adjacent router.

Cause: Software error in adjacent router.

Action: Correct software error.

Cause: Line error causing data corruption.

Action: Examine network error counters.

DN.019

Level: UE-ERROR

Short Syntax: DN.019 event 4.13: Init fail; vers skew (*Version_number*. *ECO_number*. *user_ECO_number*) node *area*. *node*, cir *number* net *network_name*

Long Syntax: DN.019 event 4.13: Initialization failure; version skew (*Version_number*. *ECO_number*. *user_ECO_number*) node *area*. *node*, cir *number* net *network_name*

Description: A Router Hello Message was received with a Routing Layer version number lower than 2.0.0. No adjacency will be created. (Messages with version numbers exceeding 2.0.0 are dropped silently, per the DECnet specifications.)

Cause: Attempt to have adjacency with Phase III router.

Action: Adjacencies with Phase III routers are not supported, reconfigure network.

DN.020

Level: U-INFO

Short Syntax: DN.020 event 4.14: Node reach change; node *area*. *node* reachable

Long Syntax: DN.020 event 4.14: Node reachability change; node *area*. *node* reachable

Description: The specified node is now reachable,

either due to an endnode adjacency with the node, or by being included in a Level 1 Routing Message.

DN.021

Level: U-INFO

Short Syntax: DN.021 event 4.14: Node reach change; node *area*. *node* unreachable

Long Syntax: DN.021 event 4.14: Node reachability change; node *area*. *node* unreachable

Description: The specified node is now unreachable.

Cause: Circuit to the node down.

Action: See if earlier message was circuit down (Event 5.0).

Cause: Endnode adjacency down.

Action: See if earlier message was adjacency down (Event 4.18). Could be due to node down, or due to failure of network connection on that machine.

Cause: Intervening node down.

Action: See if the necessary routers are reachable.

Cause: Node down.

Action: Verify whether node is up.

Cause: Cost to node exceeds EXECUTOR MAXIMUM COST.

Action: Verify that EXECUTOR MAXIMUM COST is large enough to span the network.

Cause: Cost to node exceeds EXECUTOR MAXIMUM HOPS.

Action: Verify that EXECUTOR MAXIMUM HOPS is large enough to span the network.

DN.022

Level: C-INFO

Short Syntax: DN.022 event 4.15: Adj up; new endnode *area*. *node* cir *number* net *network_name*

Long Syntax: DN.022 event 4.15: Adjacency up; new endnode *area*. *node* circuit *number* network *network_name*

Description: There is now an adjacency with the specified endnode on the specified network.

Cause: Received valid endnode hello message.

DN.023

Level: C-INFO

Short Syntax: DN.023 event 4.15: Adj up; new router *area*. *node* cir *number* net *network_name*

Long Syntax: DN.023 event 4.15: Adjacency up; new router *area*. *node* circuit *number* network *network_name*

Description: There is now an adjacency with the specified router on one of the directly connected networks. Level 1 (and 2) Routing Messages will now be accepted from this node.

Cause: Received valid router hello message containing this router's node-id in the R/S-LIST.

DN.024

Level: UI-ERROR

Short Syntax: DN.024 event 4.16: Adj rej; table full for endnode *area*. *node*, cir *number* net *network_name*

Long Syntax: DN.024 event 4.16: Adjacency rejected; table too full for endnode *area*. *node*, circuit *number* network *network_name*

Description: An Endnode Hello Message has been received from a new endnode, but there are too many endnode adjacencies, and the table is full. No adjacency will be created until another endnode adjacency times out.

Cause: There are more than EXECUTOR MAXIMUM BROADCAST NONROUTERS endnodes with adjacencies to this router.

Action: Increase EXECUTOR MAXIMUM BROADCAST NONROUTERS.

DN.025

Level: UI-ERROR

Short Syntax: DN.025 event 4.16: Adj rej; table full for rtr *source*, cir *number* net *network_name*

Long Syntax: DN.025 event 4.16: Adjacency rejected; table too full for router *source*, circuit *number* network *network_name*

Description: A Router Hello Message has been received from a new router, but there are too many router adjacencies, and the table is full. No adjacency will be created until another router adjacency times out. No routes will be accepted from this router, since there is no adjacency.

Cause: There are more than EXECUTOR MAXIMUM BROADCAST ROUTERS endnodes with adjacencies to this router.

Action: Increase EXECUTOR MAXIMUM BROADCAST ROUTERS.

DN.026

Level: UI-ERROR

Short Syntax: DN.026 event 4.16: Adj rej; too many rtrs for node *source*, cir *number* net *network_name*

Long Syntax: DN.026 event 4.16: Adjacency rejected; too many routers for node *source*, circuit *number* network *network_name*

Description: A Router Hello Message has been received from a new router on the specified circuit, but there are too many router adjacencies on this circuit, and the table is full. No adjacency will be created until another router adjacency on this circuit times out. No routes will be accepted from this router, since there is no adjacency.

Cause: There are more than CIRCUIT MAXIMUM ROUTERS endnodes with adjacencies to this router.

Action: Increase CIRCUIT MAXIMUM ROUTERS.

DN.027

Level: U-INFO

Short Syntax: DN.027 event 4.17: Area reach change; area *area* reachable

Long Syntax: DN.027 event 4.17: Area reachability change; area *area* reachable

Description: The specified area is now reachable due to being included in a Level 2 Routing Message.

DN.028

Level: U-INFO

Short Syntax: DN.028 event 4.17: Area reach change; area *area* unreachable

Long Syntax: DN.028 event 4.17: Area reachability change; area *area* unreachable

Description: The specified area is now unreachable, due to a circuit going down, a router adjacency timing out, an endnode adjacency timing out, or by the cost to that node exceeding EXECUTOR MAXIMUM COST. A preceding message should indicate the cause.

Cause: Circuit to the area down.

Action: See if earlier message was circuit down (Event 5.0).

Cause: Adjacent router down.

Action: See if earlier message was adjacency down (Event 4.18) for the router to the area.

Cause: Intervening router down.

Action: See if the necessary routers are reachable.

Cause: Cost to area exceeds EXECUTOR AREA MAXIMUM COST.

Action: Verify that EXECUTOR AREA MAXIMUM COST is large enough to span the network.

Cause: Hops to area exceeds EXECUTOR AREA MAXIMUM HOPS.

Action: Verify that EXECUTOR AREA MAXIMUM HOPS is large enough to span the network.

DN.029

Level: UE-ERROR

Short Syntax: DN.029 event 4.18: Adj dwn; invalid data from *area*. *node* cir *number* net *network_name*

Long Syntax: DN.029 event 4.18: Adjacency down, line fault; adjacent node listener received invalid data from node *area*. *node* circuit *number* network *network_name*

Description: The (optional) test data in an Endnode Hello Message was not valid, differing from the expected test pattern of 252 octal. The adjacency will be taken down.

Cause: Data corruption on network, or software error in remote node.

DN.030

Level: UE-ERROR

Short Syntax: DN.030 event 4.18: Adj dwn: node *area*. *node* chng to endnode, cir *number* net *network_name*

Long Syntax: DN.030 event 4.18: Adjacency down: node *area*. *node* changed to endnode, circuit *number* network *network_name*

Description: An Endnode Hello Message has been received from a node that had previously been a router adjacency. The existing router adjacency will be taken down, and an endnode adjacency created.

Cause: This would occur if the type of the of the adjacent node was changed quickly.

Action: Do not change node types without taking them down first.

Cause: Two nodes of different type at the same address.

Action: Ensure that node ID's are unique.

DN.031

Level: UE-ERROR

Short Syntax: DN.031 event 4.18: Adj dwn: router *area*. *node* chng type, cir *number* net *network_name*

Long Syntax: DN.031 event 4.18: Adjacency down: router *area*. *node* changed type, cir *number* net *network_name*

Description: A Router Hello Message has been received from a node whose existing adjacency was for the other type of router (level 1 versus level 2). The existing router adjacency will be taken down, and a new router adjacency created.

Cause: The type of the adjacent node was chnaged quickly.

Action: Do not change node types without taking them down first.

Cause: Two nodes of different type at the same address.

Action: Ensure that node ID's are unique.

DN.032

Level: C-INFO

Short Syntax: DN.032 event 4.18: Adj dwn; cir *number* net *network_name* down to node *area*. *node*

Long Syntax: DN.032 event 4.18: Adjacency down; circuit *number* network *network_name* down to node *area*. *node*

Description: The specified adjacency has gone down. All routes through this adjacency will be deleted.

Cause: The associated circuit has gone down.

Action: See if earlier message was circuit down (Event 5.0).

Cause: A Router Hello Message was received from a higher priority router.

Action: See if earlier message was adjacency reject (Event 4.16).

DN.033

Level: C-INFO

Short Syntax: DN.033 event 4.18: Adj dwn; node *area*. *node*, cir *number* net *network_name* timed out

Long Syntax: DN.033 event 4.18: Adjacency down; node *area*. *node*, circuit *number* network *network_name* timed out

Description: The specified adjacency has gone down because a Router Hello Message has not been heard from the adjacency for three times the hello time presented in the adjacency's Router Hello Message (the adjacency's CIRCUIT HELLO TIMER). All routes through this adjacency will be deleted.

Cause: Node down.

Action: Check status of node.

Cause: Node disconnected from network.

Action: Check circuit and line status on node.

DN.034

Level: UE-ERROR

Short Syntax: DN.034 event 4.18: Adj dwn; lvl 1 route from *area*. *node*, cir *number* net *network_name*, cksum *received_checksum*, expct *correct_checksum*

Long Syntax: DN.034 event 4.18: Adjacency down; lvl 1 route from *area*. *node*, circuit *number* network *network_name*, checksum *received_checksum*, expected *correct_checksum*

Description: A Level 1 Routing Message was received with an invalid checksum. The packet will be dropped, and the adjacency with the router will be taken down.

Cause: Data corruption error.

Action: Check network error counters.

Cause: Programming error at remote node.

Action: See if error is consistent from a particular node.

DN.035

Level: UE-ERROR

Short Syntax: DN.035 event 4.18: Adj dwn; lvl 2 route from *area*. *node*, cir *number* net *network_name*, cksum *received_checksum*, expct *correct_checksum*

Long Syntax: DN.035 event 4.18: Adjacency down; lvl 2 route from *area*. *node*, circuit *number* network *network_name*, checksum *received_checksum*, expected *correct_checksum*

Description: A Level 2 Routing Message was received with an invalid checksum. The packet will be dropped, and the adjacency with the router will be taken down.

Cause: Data corruption error.

Action: Check network error counters.

Cause: Programming error at remote node.

Action: See if error is consistent from a particular node.

DN.036

Level: C-INFO

Short Syntax: DN.036 event 4.19: Adj dwn: dropped by rtr *area*. *node*, cir *number* net *network_name*

Long Syntax: DN.036 event 4.19: Adjacency down, operator initiated: dropped by router *area*. *node*, circuit *number* network *network_name*

Description: A Router Hello Message has been received from a router that we have an adjacency with, but does not include our address in the router state list. The adjacency will be taken down, and will not come back up until our address is in the router state list.

Cause: Adjacent router restarted.

Cause: One-way communication. While this router can receive packets from the adjacent router, the adjacent router cannot receive packets from this router.

Action: Ensure that there is two-way communication on the circuit.

DN.037

Level: U-INFO

Short Syntax: DN.037 event 5.0: Circ dwn; cir *number* net *network_name*

Long Syntax: DN.037 event 5.0: Circuit down; cir *number* net *network_name*

Description: A circuit has gone down. All adjacencies via this circuit will be taken down.

Cause: Self-test failure.

Action: Look for self-test error messages, check status of interface.

Cause: Disabling circuit via CGWCON, by the SET CIRCUIT STATE OFF command, or by the SET EXECUTOR STATE OFF command.

DN.038

Level: U-INFO

Short Syntax: DN.038 event 5.0: Circ up; cir *number* net *network_name*

Long Syntax: DN.038 event 5.0: Circuit up; cir *number* net *network_name*

Description: A circuit has gone up, due either to enabling the circuit via CGWCON, due to a self-test success, by the NCP SET CIRCUIT STATE ON command, or by the NCP SET EXECUTOR STATE ON command. The router will start sending router hellos on the circuit.

DN.039

Level: UI-ERROR

Short Syntax: DN.039 event 5.14: Send fail; rsn *reason_code*, *source* -> *destination* cir *number* net *network_name*

Long Syntax: DN.039 event 5.14: Send failure on line; reason *reason_code*, packet from *source* to *destination* cir *number* net *network_name*

Description: The sending of a packet being forwarded failed. The reason_code is the internal error code for the failure.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

DN.040

Level: P-TRACE

Short Syntax: DN.040 *source* -> *destination*

Long Syntax: DN.040 Forwarding packet from *source* to *destination*

Description: Forwarding a packet from one node to another.

DN.041

Level: P-TRACE

Short Syntax: DN.041 MOP Req ID pkt rcvd frm *MAC_address* cir *number* net *network_name*

Long Syntax: DN.041 MOP Request ID packet received from node *MAC_address* circuit *number* network *network_name*

Description: A DECnet Maintenance Operations Protocol MOP Request System ID packet was received from the specified node. A MOP System ID packet will be sent to the requester's address.

DN.042

Level: P-TRACE

Short Syntax: DN.042 MOP Sys ID pkt rcvd frm *MAC_address* cir *number* net *network_name*

Long Syntax: DN.042 MOP System ID packet received from node *MAC_address* circuit *number* network *network_name*

Description: A DECnet MOP System ID packet was received from the specified node.

DN.043

Level: UE-ERROR

Short Syntax: DN.043 MOP pkt rcvd unk opc *opcode* frm *MAC_address* cir *number* net *network_name*

Long Syntax: DN.043 MOP packet received unknown opcode *opcode* from node *MAC_address* cir *number* net *network_name*

Description: DECnet MOP (Maintenance Operations Protocol) packet received with unsupported opcode from specified node. The packet will be ignored.

Cause: Programming error on remote note.

Cause: Data corruption.

DN.045

Level: UI-ERROR

Short Syntax: DN.045 acc cnt bad rec, cir *number* net *network_name*, purge

Long Syntax: DN.045 Access control bad SRAM record, circuit *number* network *network_name*, purge

Description: There is a faulty access control record in the permanent database for this circuit.

Action: Do a PURGE MODULE ACCESS CONTROL CIRCUIT.

DN.046

Level: C-INFO

Short Syntax: DN.046 acc cont fail *source* -> *destination* cir *number* net *network_name*

Long Syntax: DN.046 Access control failed, packet from *source* to *destination* circuit *number* network *network_name*

Description: A packet was not forwarded between the two hosts due to access control restrictions. If Request Return to Sender was set in the header, the packet will be returned to the sender, otherwise it will be dropped.

Cause: User attempting to contact host is restricted by access control.

DN.047

Level: C-INFO

Short Syntax: DN.047 desig router chng frm *old_router* to *new_router*, cir *number* net *network_name*

Long Syntax: DN.047 Designated router changed from *old_router* to *new_router*, circuit *number* network *network_name*

Description: Designated router for this circuit has changed.

Cause: New router adjacency with higher router priority on circuit, or same router priority and higher node address.

DN.048

Level: C-INFO

Short Syntax: DN.048 desig router *address* select, cir *number* net *network_name*

Long Syntax: DN.048 Designated router *address* selected, circuit *number* network *network_name*

Description: There is now a designated router for this circuit, where there had not been one before.

DN.049

Level: P-TRACE

Short Syntax: DN.049 endnode hello len *packet_length* from *node*, cir *number* net *network_name*

Long Syntax: DN.049 endnode hello length *packet_length* from *node*, circuit *number* network *network_name*

Description: Received endnode hello message from specified endnode.

DN.050

Level: ALWAYS

Short Syntax: DN.050 executor node address *area*. *node* exceeds MAX ADDRESS *max_address*

Long Syntax: DN.050 executor node address *area*. *node* exceeds EXECUTOR MAX ADDRESS *max_address*

Description: The EXECUTOR ADDRESS stored in the permanent database exceeds the EXECUTOR MAXIMUM ADDRESS stored in the permanent database. DECnet will be left off, but the database will be allocated.

Action: Either correct EXECUTOR ADDRESS or EXECUTOR MAX ADDRESS.

DN.051

Level: ALWAYS

Short Syntax: DN.051 executor node address *area*. *node* exceeds MAX AREA *max_node*

Long Syntax: DN.051 executor node address *area*. *node* exceeds EXECUTOR MAX AREA *max_node*

Description: The area of the EXECUTOR ADDRESS stored in the permanent database exceeds the EXECUTOR MAXIMUM AREA stored in the permanent database. DECnet will be left off, but the database will be allocated.

Action: Either correct EXECUTOR ADDRESS or EXECUTOR MAX AREA.

DN.053

Level: CI-ERROR

Short Syntax: DN.053 inp que ovflow data *source* -> *destination* cir *number* net *network_name*

Long Syntax: DN.053 Input queue overflow data packet from *source* to *destination* circuit *number* network *network_name*

Description: The DECnet input queue overflowed for incoming Long Format Data packet. The packet will be dropped.

Cause: Too much traffic for forwarder to forward.

Action: Adjust circuit costs to balance traffic between paths. Reconfigure network. Increase speed of router.

Cause: Inadquate buffer resources.

Action: Examine memory statistics in GWCON. More buffers can be made available by ensuring that DECnet configuration does not have excess adjacency memory allocated.

Action: Increase memory.

DN.054

Level: CI-ERROR

Short Syntax: DN.054 inp que ovflow multicast from *source* cir *number* net *network_name*

Long Syntax: DN.054 Input queue overflow multicast from *source* circuit *number* network *network_name*

Description: The DECnet input queue overflowed for incoming routing or hello multicast packet. The packet will be dropped.

Cause: Too much traffic for forwarder to forward.

Action: Adjust circuit costs to balance traffic between paths. Reconfigure network. Increase speed of router.

Cause: Inadequate buffer resources.

Action: Examine memory statistics in GWCON. More buffers can be made available by ensuring that DECnet configuration does not have excess adjacency memory allocated.

Action: Increase memory.

DN.055

Level: U-TRACE

Short Syntax: DN.055 lvl 1 rte pkt from *source* ign, cir *number* net *network_name*, no adjacency

Long Syntax: DN.055 Level 1 routing message from *source* ignored, circuit *number* network *network_name*, no adjacency with router

Description: A Level 1 Routing Message was received from a router that does not have an active adjacency with this router. The routing packet will not be processed.

Cause: This will happen occasionally when the other router develops an adjacency with this router before this one does.

Action: No action needed unless message is persistent.

DN.056

Level: P-TRACE

Short Syntax: DN.056 lvl 1 rte pkt len *received_length* from *source*, cir *number* net *network_name*

Long Syntax: DN.056 Level 1 routing packet length *received_length* from *source*, circuit *number* network *network_name*

Description: A Level 1 Routing Message was received from the specified router.

DN.057

Level: U-TRACE

Short Syntax: DN.057 lvl 2 rte pkt from *source* ign, cir *number* net *network_name*, no adjacency

Long Syntax: DN.057 Level 2 routing message from *source* ignored, circuit *number* network *network_name*, no adjacency with router

Description: A Level 2 Routing Message was received from a router that does not have an active adjacency with this router. The routing packet will not be processed.

Cause: This will happen occassionally when the other router develops an adjacency with this router before this one does.

Action: No action needed unless message is persistent.

Cause: Level 2 routing message sent by level 1 router.

Action: Correct software error at sending router.

DN.058

Level: P-TRACE

Short Syntax: DN.058 lvl 2 rte pkt len *received_length* from *source*, cir *number* net *network_name*

Long Syntax: DN.058 Level 2 routing packet length *received_length* from *source*, circuit *number* network *network_name*

Description: A Level 2 Routing Message was received from the specified router.

DN.059

Level: UI-ERROR

Short Syntax: DN.059 no buffer for hello on cir *number* net *network_name*

Long Syntax: DN.059 No buffer to build hello packet to send on circuit *number* network *network_name*

Description: No packet buffer was available to construct and send a Router Hello Message.

Cause: Severe packet buffer shortage.

Action: Check memory statistics in GWCON to verify packet buffer level. If possible, make routing tables smaller. In DECnet, this is done by minimizing the number of adjacencies allowed. (Configure EXECUTOR MAXIMUM BROADCAST ROUTERS and EXECUTOR MAXIMUM BROADCAST NONROUTERS to

minimum appropriate values.) If routing tables cannot be made smaller, increase memory size.

Cause: Traffic peak using all available buffers.

Action: This is the problem if this message occurs very infrequently.

DN.060

Level: UI-ERROR

Short Syntax: DN.060 no buffer for lvl 1 rte on cir *number* net *network_name*

Long Syntax: DN.060 No buffer to build level 1 routing message to send on circuit *number* network *network_name*

Description: No packet buffer was available to construct and send a Level 1 Routing Message.

Cause: Severe packet buffer shortage.

Action: Check memory statistics in GWCON to verify packet buffer level. If possible, make routing tables smaller. In DECnet, this is done by minimizing the number of adjacencies allowed. (Configure EXECUTOR MAXIMUM BROADCAST ROUTERS and EXECUTOR MAXIMUM BROADCAST NONROUTERS to minimum appropriate values.) If routing tables cannot be made smaller, increase memory size.

Cause: Traffic peak using all available buffers.

Action: This is the problem if this message occurs very infrequently.

DN.061

Level: UI-ERROR

Short Syntax: DN.061 no buffer for lvl 2 rte on cir *number* net *network_name*

Long Syntax: DN.061 No buffer to build level 2 routing message to send on circuit *number* network *network_name*

Description: No packet buffer was available to construct and send a Level 2 Routing Message.

Cause: Severe packet buffer shortage.

Action: Check memory statistics in GWCON to verify packet buffer level. If possible, make routing tables smaller. In DECnet, this is done by minimizing the number of adjacencies allowed. (Configure EXECUTOR MAXIMUM BROADCAST ROUTERS and EXECUTOR MAXIMUM BROADCAST NONROUTERS to minimum appropriate values.) If routing tables cannot be made smaller, increase memory size.

Cause: Traffic peak using all available buffers.

Action: This is the problem if this message occurs very infrequently.

DN.062

Level: ALWAYS

Short Syntax: DN.062 no memory for NCP circuit name table

Long Syntax: DN.062 No memory for building NCP circuit name table

Description: No memory was available to build the circuit name table for NCP at start time.

Cause: There is some configuration error causing a grave memory shortage.

Action: Reduce memory demand by making routing tables smaller, or getting more memory.

DN.063

Level: ALWAYS

Short Syntax: DN.063 no memory for routing tables (*number* bytes req), DECnet disabled

Long Syntax: DN.063 No Memory for building routing tables (*number* bytes required), DECnet disabled

Description: The routing tables required more memory than was available. DECnet is disabled.

Cause: Parameters that determine size of routing database are too large for actual network configuration.

Action: The following parameters should be reduced as appropriate using the DEFINE commands, and the gateway restarted: EXECUTOR MAXIMUM BROADCAST ROUTERS, EXECUTOR MAXIMUM BROADCAST NONROUTERS, CIRCUIT MAXIMUM ROUTERS, EXECUTOR MAXIMUM ADDRESS, EXECUTOR MAXIMUM AREA,

Cause: Inadequate memory size.

Action: Upgrade for more memory.

DN.064

Level: CI-ERROR

Short Syntax: DN.064 packet received on down cir *number* net *network_name*, dropped

Long Syntax: DN.064 Packet received on down circuit *number* network *network_name*, packet dropped

Description: Received a data packet on a circuit or rotuer that does not have DECnet enabled. The packet will be dropped.

DN.066

Level: U-TRACE

Short Syntax: DN.066 returning packet to sender *sender* <- *original_destination*

Long Syntax: DN.066 returning packet to sender *sender* from *original_destination*

Description: A data packet could not reach the destination, and had the Request Return to Sender bit set in the header. It is being returned to the sender.

Cause: Should be explained by a previous message, such as Events 4.1, 4.2, and 4.3.

Action: See action in causative message.

DN.067

Level: P-TRACE

Short Syntax: DN.067 router hello len *received_length* from *source*, cir *number* net *network_name*

Long Syntax: DN.067 Router hello length *received_length* received from *source*, circuit *number* network *network_name*

Description: A Router Hello Message was received from the specified router.

DN.068

Level: P-TRACE

Short Syntax: DN.068 sending desig rtr hello on cir *number* net *network_name*

Long Syntax: DN.068 Sending designated router hello on circuit *number* network *network_name*

Description: A Router Hello Message is being sent to the ALLENDNODES address, as this router is the designated router on the specified circuit.

DN.069

Level: P-TRACE

Short Syntax: DN.069 sending hello on cir *number* net *network_name*

Long Syntax: DN.069 Sending router hello on circuit *number* network *network_name*

Description: A Router Hello Message is being sent to the ALLROUTERS address on the specified circuit.

DN.070

Level: P-TRACE

Short Syntax: DN.070 sending lvl 1 rte on cir *number* net *network_name*

Long Syntax: DN.070 Sending level 1 routing message on circuit *number* network *network_name*

Description: A Level 1 Routing Message is being sent to the ALLROUTERS address on the specified circuit.

DN.071

Level: P-TRACE

Short Syntax: DN.071 sending lvl 2 rte on cir *number* net *network_name*

Long Syntax: DN.071 Sending level 2 routing message on circuit *number* network *network_name*

Description: A Level 2 Routing Message is being sent to the ALLROUTERS address on the specified circuit.

DN.072

Level: ALWAYS

Short Syntax: DN.072 too many router adjacencies *total_adjacencies*, NBRA = *maximum_adjacencies*

Long Syntax: DN.072 Too many router adjacencies configured, sum = *total_adjacencies*, NBRA = *maximum_adjacencies*

Description: The permanent database has been configured such that the sum of CIRCUIT MAXIMUM ROUTERS for all circuits exceeds EXECUTOR MAXIMUM BROADCAST ROUTERS. This error is non-fatal, but new values should be DEFINED, and the gateway restarted.

Cause: CIRCUIT MAXIMUM ROUTERS too large.

Action: This is the usual problem, especially on Serial Line interfaces, where there can only be one router adjacency.

Cause: EXECUTOR MAXIMUM BROADCAST ROUTERS too small.

Action: This is not normally the problem, as the default is 32, which is quite generous.

DN.073

Level: C-INFO

Short Syntax: DN.073 new 1-way adj *sender* cir *number* net *network_name*

Long Syntax: DN.073 new 1-way adjacency with node *sender* on circuit *number* network *network_name*

Description: We have just received a router hello message from the specified router, but our address is not in the router/state list of the hello message. We have a one-way adjacency with this router, it will not be two-way until our address is in the router/state list.

Cause: New node came up.

Action: None required unless adjacency never reaches two way. This should happen shortly. If it does not, it may indicate that our address is beyond the other routers EXECUTOR MAXIMUM ADDRESS.

DN.074

Level: C-INFO

Short Syntax: DN.074 1-way adj *sender* timed out cir *number* net *network_name*

Long Syntax: DN.074 1-way adjacency with node *sender* timed out on circuit *number* network *network_name*

Description: We have stopped receiving router hellos without our node address in the router/state list from the specified router. The timeout is three times the hello timer that was specified in the last router hello from this router. The partial adjacency with this router will be eliminated.

Cause: New node never came up all the way.

DN.075

Level: P-TRACE

Short Syntax: DN.075 Pkt for me frm *sender*

Long Syntax: DN.075 Packet for me from node *sender*

Description: We have received a packet addressed to us. It will be checked to see what transport protocol it is for.

DN.076

Level: U-TRACE

Short Syntax: DN.076 NSP unsupp msg type *msgflg* frm *sender*

Long Syntax: DN.076 NSP unsupported message type *msgflg* from node *sender*

Description: We have received an NSP packet of a message type that we do not process. Only Connect Initiate Messages are processed.

DN.077

Level: CE-ERROR

Short Syntax: DN.077 Unk trans type *msgflg* from *sender*

Long Syntax: DN.077 Unknown transport protocol type *msgflg*from ndoe *sender*

Description: We have received a data packet that is not for the NSP transport protocol.

DN.078

Level: C-INFO

Short Syntax: DN.078 NSP conn init from *sender*, reject

Long Syntax: DN.078 NSP Connect Initiate Message received from node *sender*, rejecting

Description: An NSP Connect Initiate or Retransmitted Connect Initiate Message was received from the specified node. A Disconnect Initiate message will be sent in return, with a Session Reject error code of 4 (destination end user does not exist).

Cause: User on remote machine attempted to initiate an NSP connection, but there are no Session clients supported in the router.

DN.079

Level: UE-ERROR

Short Syntax: DN.079 endnode hello from *sender* cir *number* net *network_name* dup addr w/self, ign

Long Syntax: DN.079 endnode hello from node *sender* circuit *number* network *network_name*, duplicate address with self, ignoring

Description: An Endnode Hello Message was received from a node with the same DECnet address as this router. Since duplicate node addresses are not allowed, and the router is more important, the hello message will be ignored.

Cause: User configuration error.

Action: Change DECnet node address.

DN.080

Level: P-TRACE

Short Syntax: DN.080 MOP Req Cnt pkt rcvd frm *MAC_address* cir *number* net *network_name*

Long Syntax: DN.080 MOP Request Counters packet received from node *MAC_address* circuit *number* network *network_name*

Description: A DECnet Maintenance Operations Protocol (MOP) Request Counters packet was received from the specified node. A MOP Counters packet will be sent to the requester's address.

DN.081

Level: P-TRACE

Short Syntax: DN.081 MOP Cnt pkt snt to *MAC_address* cir *number* net *network_name*

Long Syntax: DN.081 MOP Counters packet sent to node *MAC_address* circuit *number* network *network_name*

Description: A DECnet Maintenance Operations Protocol (MOP) Counters packet is being sent to the specified address.

DN.082

Level: P-TRACE

Short Syntax: DN.082 MOP Sys ID pkt snt to *MAC_address* cir *number* net *network_name*

Long Syntax: DN.082 MOP System ID packet sent to node *MAC_address* circuit *number* network *network_name*

Description: A DECnet Maintenance Operations System ID packet is being sent to the specified address.

DN.083

Level: P-TRACE

Short Syntax: DN.083 MOP Sys ID pkt snt to MOP cir *number* net *network_name*

Long Syntax: DN.083 MOP System ID packet sent to MOP circuit *number* network *network_name*

Description: A DECnet Maintenance Operations Protocol System ID packet is being sent to the MOP multicast address AB-00-00-02-00-00.

DN.084

Level: UI-ERROR

Short Syntax: DN.084 MOP Cnt Req frm *MAC_address* not supp on cir *number* net *network_name*

Long Syntax: DN.084 MOP Cnt Req from node *MAC_address* not supported on circuit *number* network *network_name*

Description: A DECnet Maintenance Operations Protocol (MOP) Request Counters was received from the specified host, but there is no support for the MOP Counters message on this circuit.

DN.085

Level: UI-ERROR

Short Syntax: DN.085 Ph IV rtr hlo wo bilingual rtr frm *node_number* on cir *number* net *network_name*

Long Syntax: DN.085 Ph IV router hello without bilingual router from *node_number* on circuit *number* network *network_name*

Description: A DECnet Phase IV broadcast router hello was received on a circuit that was configured for Phase IV′ only.

Cause: Router is receiving Phase IV broadcast router hello packets on a network that should only have Phase IV′ packets

Action: Router must be configured for both Phase IV and Phase IV′ to receive the broadcast router hello packets from a Phase IV router.

DN.086

Level: UI-ERROR

Short Syntax: DN.086 Ph IV ennd hlo wo bilingual rtr frm *node_number* on cir *circut_number* net *node_name*

Long Syntax: DN.086 Ph IV endnode hello without bilingual router from *node_number* on circuit *circut_number* network *node_name*

Description: A DECnet Phase IV broadcast endnode hello was received on a circuit that was configured for Phase IV′ only.

Cause: The router is receiving Phase IV broadcast endnode hello packets on a network that should only have Phase IV′ packets.

Action: The router must be configured for both Phase IV and Phase IV′ to receive the broadcast endnode hello packets from a Phase IV endnode.

DN.087

Level: UI-ERROR

Short Syntax: DN.087 Ph IV′ rtr hlo wo bilingual or ama rtr frm *node_number* on cir *circut_number* net *node_name*

Long Syntax: DN.087 Ph IV′ router hello without bilingual or ama router from *node_number* on circuit *circut_number* network *node_name*

Description: A DECnet Phase IV′ broadcast router hello was received on a circuit that was configured for Phase IV only.

Cause: The router is receiving Phase IV′ broadcast router hello packets on a network that should only have Phase IV packets.

Action: The router must be configured for Phase IV′ to receive the broadcast endnode hello packets from a Phase IV′ endnode.

DN.088

Level: UI-ERROR

Short Syntax: DN.088 Ph IV′ ennd hlo wo bilingual or ama rtr frm *node_number* on cir *circut_number* net *node_name*

Long Syntax: DN.088 Ph IV′ endnode hello without bilingual or ama router from *node_number* on circuit *circut_number* network *node_name*

Description: A DECnet Phase IV′ broadcast endnode hello was received on a circuit that was configured for Phase IV only.

Cause: The router is receiving Phase IV′ broadcast endnode hello packets on a network that should only have Phase IV packets.

Action: The router must be configured for Phase IV′

to receive the broadcast endnode hello packets from a Phase IV′ endnode.

DN.089

Level: UI-ERROR

Short Syntax: DN.089 Unkn ennd hlo format frm *node_number* on cir *circut_number* net *node_name*

Long Syntax: DN.089 Unknown endnode hello message format from *node_number* on circuit *circut_number* network *node_name*

Description: The router received an Endnode Hello Message with unknown format.

Cause: Some station is sending a message with this format.

Action: Determine the errant node from this message and inform the manufacturer that this node is sending hello messages of unknown format.

DN.090

Level: UI-ERROR

Short Syntax: DN.090 Cannot bld lvl 1 rte on cir *number* net *network_name*, blk sz too small - *block_size*

Long Syntax: DN.090 Cannot build level 1 routing message on circuit *number*, network *network_name*, block size too small - *block_size*

Description: A Level 1 Routing Message cannot be built because the circuit's minimum block size is too small.

DN.091

Level: UI-ERROR

Short Syntax: DN.091 Send fail for hello, rsn *reason_code*, cir *number* net *network_name*

Long Syntax: DN.091 Send failed for router hello packet, reason *reason_code*, on circuit *number* network *network_name*

Description: The transmission of a router hello packet failed on the specified circuit for the reason number given in reason_code. Occasional occurrences of this will not disrupt the protocol, but continuing occurrences will disrupt the protocol.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

DN.092

Level: UI-ERROR

Short Syntax: DN.092 Send fail for lvl 1 rte, rsn *reason_code*, cir *number* net *network_name*

Long Syntax: DN.092 Send failed for level 1 routing message, reason *reason_code*, on circuit *number* network *network_name*

Description: The transmission of a Level 1 Routing Message failed on the specified circuit for the reason number given in reason_code. Occasional occurrences of this will not disrupt the protocol, but continuing occurrences will disrupt the protocol.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

DN.093

Level: UI-ERROR

Short Syntax: DN.093 Send fail for lvl 2 rte, rsn *reason_code*, cir *number* net *network_name*

Long Syntax: DN.093 Send failed for level 2 routing message, reason *reason_code*, on circuit *number* network *network_name*

Description: The transmission of a Level 2 Routing Message failed on the specified circuit for the reason number given in reason_code. Occasional occurrences of this will not disrupt the protocol, but continuing occurrences will disrupt the protocol.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

DN.094

Level: UI-ERROR

Short Syntax: DN.094 Send fail for MOP *message_type*, rsn *reason_code*, cir *number* net *network_name*

Long Syntax: DN.094 Send failed for MOP *message_type* message, reason *reason_code*, on circuit *number* network *network_name*

Description: The transmission of a MOP message failed on the specified circuit for the reason number given in reason_code. The message_type is one of ''System ID'' or ''Counters.'' Occasional occurrences of this will not disrupt the protocol, but continuing occurrences will disrupt the protocol.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

Check dnrouttype

Short Syntax: Unknown circuit router type.

Description: The circuit router type is unknown.

Cause: Data corruption, probably from coding error.

Check dnrfgtl

Short Syntax: DN routes() called with first > last

Description: The dnroute routine was called with an invalid node address range.

Cause: Internal consistency error.

Action: Report to customer service, preferably with a core dump.

Check dnrbeaf

Short Syntax: DN routes() BEA optimization failed

Description: The dnroute routine has computed a route via a broadcast circuit, rather than via a router or endnode.

Cause: Internal consistency error.

Action: Report to customer service, preferably with a core dump.

Check dnarfgtl

Short Syntax: DN Aroutes() called with first > last

Description: The dnAroute routine was called with an invalid area range.

Cause: Internal consistency error.

Action: Report to customer service, preferably with a core dump.

Check dnmembug

Short Syntax: DN no memory for table

Description: An allocation of memory for the routing tables failed, but a check of free memory had indicated that there should be enough memory.

Cause: Internal consistency check.

Action: Report to customer service, preferably with a core dump.

Fatal dnadbadarg

Short Syntax: DN bad arg to dnadjdown()

Description: The dnadjdown routine was asked to remove an adjacency that was not a router or an endnode.

Cause: Internal consistency error.

Action: Report to customer service, preferably with a core dump.

Fatal dnacnmr

Short Syntax: DN no mem to read acc cntl

Description: There is no memory available to read the access control lists from the permanent database.

Cause: Severe memory shortage.

Action: Reduce sizes of routing tables to use less memory, or add additional memory.

Fatal dnacnmsac

Short Syntax: DN no mem to store acc cntl

Description: There is no memory available to store the access control lists for use.

Cause: Severe memory shortage.

Action: Reduce sizes of routing tables to use less memory, or add addtional memory.

Fatal dnacnmcac

Short Syntax: DN no mem for acc cntl

Description: There is no memory available to build the access control list.

Cause: Severe memory shortage.

Action: Reduce sizes of routing tables to use less memory, or add additional memory.

Fatal dncnmrfi

Short Syntax: DN no mem for dnrfin

Description: There is no memory available to build the circuit input routing filter table.

Cause: Severe memory shortage.

Action: Reduce sizes of routing tables to use less memory, or add additional memory.

Fatal dncnmrfo

Short Syntax: DN no mem for dnrfout

Description: There is no memory available to build the circuit output routing filter table.

Cause: Severe memory shortage.

Action: Reduce sizes of routing tables to use less memory, or add additional memory.

Fatal dncnmci

Short Syntax: DN no mem for dnccti init

Description: There is no memory available to build the circuit volatile database.

Cause: Severe memory shortage.

Action: Reduce sizes of routing tables to use less memory, or add additional memory.

Panic dnrtcrtos

Short Syntax: DN routing table corrupt: routes to self

Description: The routing database consistency checker has detected an inconsistency in the routing database. The router will be restarted.

Cause: Memory corruption.

Action: Configure for core dump, and report to customer service.

Cause: Internal software error.

Action: Configure for core dump, and report to customer service.

Panic dnrtcarts

Short Syntax: DN routing table corrupt: area routes to self

Description: The routing database consistency checker has detected an inconsistency in the routing database. The router will be restarted.

Cause: Memory corruption.

Action: Configure for core dump, and report to customer service.

Cause: Internal software error.

Action: Configure for core dump, and report to customer service.

Panic dnrtcrths

Short Syntax: DN routing table corrupt: routes through self

Description: The routing database consistency checker has detected an inconsistency in the routing database. The router will be restarted.

Cause: Memory corruption.

Action: Configure for core dump, and report to customer service.

Cause: Internal software error.

Action: Configure for core dump, and report to customer service.

Panic dnrtcrtas

Short Syntax: DN routing table corrupt: route to area self

Description: The routing database consistency checker has detected an inconsistency in the routing database. The router will be restarted.

Cause: Memory corruption.

Action: Configure for core dump, and report to customer service.

Cause: Internal software error.

Action: Configure for core dump, and report to customer service.

Panic dnrtcartas

Short Syntax: DN routing table corrupt: area route to area self

Description: The routing database consistency checker has detected an inconsistency in the routing database. The router will be restarted.

Cause: Memory corruption.

Action: Configure for core dump, and report to customer service.

Cause: Internal software error.

Action: Configure for core dump, and report to customer service.

DN.095

Level: CI-ERROR

Short Syntax: DN.095 inp que ovflow data *source* -> *destination* cir *number* net *network_name*

Long Syntax: DN.095 Input queue overflow data packet from *source* to *destination* circuit *number* network *network_name*

Description: The DECnet input queue overflowed for incoming Short Format Data packet. The forwarder drops the packet.

Cause: There is too much traffic for the forwarder.

Action: Adjust circuit costs to balance traffic between paths. Reconfigure network. Increase speed of router.

Cause: Inadquate buffer resources.

Action: Examine memory statistics in GWCON. To make more buffers available, ensure that the DECnet configuration does not have excess adjacency memory allocated.

Action: Increase memory.

DN.096

Level: CI-ERROR

Short Syntax: DN.096 inp que ovflow Init Msg *source* cir *number* net *network_name*

Long Syntax: DN.096 Input queue overflow Initialization Message from *source* circuit *number* network *network_name*

Description: The DECnet input queue overflowed for incoming Initialization Message. The forwarder drops the packet.

Cause: There is too much traffic for the forwarder.

Action: Adjust circuit costs to balance traffic between paths. Reconfigure network. Increase speed of router.

Cause: Inadquate buffer resources.

Action: Examine memory statistics in GWCON. To make more buffers available, ensure that the DECnet configuration does not have excess adjacency memory allocated.

Action: Increase memory.

DN.097

Level: CI-ERROR

Short Syntax: DN.097 inp que ovflow Verif Msg *source* cir *number* net *network_name*

Long Syntax: DN.097 Input queue overflow Verification Message from *source* circuit *number* network *network_name*

Description: The DECnet input queue overflowed for incoming Verification Message. The forwarder drops the packet.

Cause: There is too much traffic for the forwarder.

Action: Adjust circuit costs to balance traffic between paths. Reconfigure network. Increase speed of router.

Cause: Inadquate buffer resources.

Action: Examine memory statistics in GWCON. To make more buffers available, ensure that the DECnet configuration does not have excess adjacency memory allocated.

Action: Increase memory.

DN.098

Level: CI-ERROR

Short Syntax: DN.098 inp que ovflow Hlo/Tst Msg *source* cir *number* net *network_name*

Long Syntax: DN.098 Input queue overflow

Hello/Test Message from *source* circuit *number* network *network_name*

Description: The DECnet input queue overflowed for incoming Hello/Test Message. The forwarder drops the packet.

Cause: Too much traffic for forwarder to forward.

Action: Adjust circuit costs to balance traffic between paths. Reconfigure network. Increase speed of router.

Cause: Inadquate buffer resources.

Action: Examine memory statistics in GWCON. To make more buffers available, ensure that the DECnet configuration does not have excess adjacency memory allocated.

Action: Increase memory.

DN.099

Level: ALWAYS

Short Syntax: DN.099 max rclls rchd cir *number* net *network_name*

Long Syntax: DN.099 Maximum recalls attempts reached on circuit *number* network *network_name*

Description: An outgoing circuit reached it maximum allowed retries to set up an X.25 virtual circuit to the remote node. This circuit places no more calls until you take the required action.

Action: Check connectivity to the X.25 switch. Then disable and enable the circuit to try calling again.

DN.100

Level: UE-ERROR

Short Syntax: DN.100 Init Msg err; cir *number* net *network_name*

Long Syntax: DN.100 Initialization Message format error; circuit *number* network *network_name*

Description: The router received an Initialization Message with invalid header information. The forwarder drops the packet.

DN.101

Level: UE-ERROR

Short Syntax: DN.101 Init Msg err - wrg ver; *source_node*, cir *number* net *network_name*

Long Syntax: DN.101 Received Initialization Message specifying unsupported version; from *source_node*, circuit *number* network *network_name*

Description: The router received an Initialization Message that specified an unsupported version number. The forwarder drops the packet.

DN.102

Level: ALWAYS

Short Syntax: DN.102 Init Msg rcvd; *source_node*, cir *number* net *network_name*

Long Syntax: DN.102 Received Initialization Message; from *source_node*, circuit *number* network *network_name*

Description: The router received an Initialization Message.

DN.103

Level: ALWAYS

Short Syntax: DN.103 Verif Msg rcvd; *source_node*, cir *number* net *network_name*

Long Syntax: DN.103 Received Verification Message; from *source_node*, circuit *number* network *network_name*

Description: The router received a Verification Message.

DN.104

Level: UE-ERROR

Short Syntax: DN.104 Verif fail; *source_node*, cir *number* net *network_name*

Long Syntax: DN.104 Verification failure; from *source_node*, circuit *number* network *network_name*

Description: Verification failure. The router detected an error in the Verification Message.

DN.105

Level: UE-ERROR

Short Syntax: DN.105 Hlo/tst fail; *source_node*, cir *number* net *network_name*

Long Syntax: DN.105 Error detected in processing Hello/Test Message; from *source_node*, circuit *number* network *network_name*

Description: The router detected an error in processing the Hello/Test Message. The forwarder drops the packet.

DN.106

Level: ALWAYS

Short Syntax: DN.106 Hlo/Tst Msg rcvd; *source_node*, cir *number* net *network_name*

Long Syntax: DN.106 Received Hello/Test Message; from *source_node*, circuit *number* network *network_name*

Description: The router received a Hello/Test Message from a neighbor.

DN.107

Level: UI-ERROR

Short Syntax: DN.107 no buffer for Init Msg on cir *number* net *network_name*

Long Syntax: DN.107 No buffer to build Initialization Message to send on circuit *number* network *network_name*

Description: No packet buffer was available to construct and send an Initialization Message.

Cause: Severe packet buffer shortage.

Action: Check memory statistics in GWCON to verify packet buffer level. If possible, make routing tables smaller. To do this in DECnet, minimize the number of adjacencies allowed. (Configure EXECUTOR MAXIMUM BROADCAST ROUTERS and EXECUTOR MAXIMUM BROADCAST NONROUTERS to minimum appropriate values.) If the router cannot make the routing tables smaller, increase memory size.

Cause: Traffic peak using all available buffers.

Action: If this message occurs infrequently, this is the problem.

DN.108

Level: UI-ERROR

Short Syntax: DN.108 Snd fail for Init Msg; cir *number* net *network_name*

Long Syntax: DN.108 Send failed for Initialization Message on circuit *number* network *network_name*

Description: The transmission of a router Initialization Message failed on the specified circuit. Occasional occurrences of this will not disrupt the protocol, but continuing occurrences will disrupt the protocol.

DN.109

Level: ALWAYS

Short Syntax: DN.109 snd Init Msg; cir *number* net *network_name*

Long Syntax: DN.109 Sending Initialization Message on circuit *number* network *network_name*

Description: The router is sending an Initialization Message on the indicated circuit.

DN.110

Level: UI-ERROR

Short Syntax: DN.110 Snd fail for Verif Msg; cir *number* net *network_name*

Long Syntax: DN.110 Send failed for Verification Message on circuit *number* network *network_name*

Description: The transmission of a router Verification Message failed on the specified circuit.

DN.111

Level: ALWAYS

Short Syntax: DN.111 snd Verif Msg; cir *number* net *network_name*

Long Syntax: DN.111 Sending Verification Message on circuit *number* network *network_name*

Description: The router is sending a Verification Message on the indicated circuit.

DN.112

Level: UI-ERROR

Short Syntax: DN.112 Snd fail for Hlo/TstMsg; cir *number* net *network_name*

Long Syntax: DN.112 Send failed for Hello/Test Message on circuit *number* network *network_name*

Description: The transmission of a router Hello/Test Message failed on the specified circuit.

DN.113

Level: ALWAYS

Short Syntax: DN.113 snd Hlo/Tst Msg; cir *number* net *network_name*

Long Syntax: DN.113 Sending Hello/Test Message on circuit *number* network *network_name*

Description: The router is sending a Hello/Test Message on the indicated circuit.

DN.114

Level: UI-ERROR

Short Syntax: DN.114 x25 reg fail

Long Syntax: DN.114 Registration with X25 service failed

Description: The forwarder could not register with X.25 services on the interface.

DN.115

Level: UI-ERROR

Short Syntax: DN.115 call req to x25 fail; intf *number* net *network_name*

Long Syntax: DN.115 Call request to X25 service failed on interface *number* network *network_name*

Description: The forwarder's call request to X.25 services failed on the indicated network.

Chapter 34. Digital Network Architecture Phase V (DNAV)

This chapter describes Digital Network Architecture Phase V (DNAV) messages. For information on message content and how to use the message, refer to the Introduction.

DNAV.001

Level: P-TRACE

Short Syntax: DNAV.001 DNA pkt forwarded via OSI at level *rtg_lvl*

Long Syntax: DNAV.001 DNA packet forwarded via OSI at level *rtg_lvl*

Description: A DNA packet was received and then passed to OSI for forwarding.

DNAV.002

Level: P-TRACE

Short Syntax: DNAV.002 DNA pkt translated to OSI pkt *source_NSAP -> destination_NSAP*

Long Syntax: DNAV.002 DNA pkt translated to OSI pkt: *source_NSAP -> destination_NSAP*

Description: A DNA data packet was successfully translated to an OSI data packet.

DNAV.003

Level: P-TRACE

Short Syntax: DNAV.003 Translation of DNA pkt to OSI pkt failed

Long Syntax: DNAV.003 Translation of DNA pkt to OSI pkt failed

Description: An attempt to translate a DNA data packet to an OSI data packet failed.

DNAV.004

Level: P-TRACE

Short Syntax: DNAV.004 OSI pkt translated to DNA pkt *src -> dst*

Long Syntax: DNAV.004 OSI pkt translated to DNA pkt: *src -> dst*

Description: An OSI data packet was successfully translated to a DNA data packet.

DNAV.005

Level: P-TRACE

Short Syntax: DNAV.005 Translation of OSI pkt to DNA pkt failed

Long Syntax: DNAV.005 Translation of OSI pkt to DNA pkt failed

Description: An attempt to translate an OSI data packet to a DNA data packet failed.

DNAV.006

Level: P-TRACE

Short Syntax: DNAV.006 OSI pkt forwarded via DNA at level *rtg_lvl*

Long Syntax: DNAV.006 OSI packet forwarded via DNA at level *rtg_lvl*

Description: An OSI packet was received and then passed to DNA for forwarding.

DNAV.007

Level: UE-ERROR

Short Syntax: DNAV.007 timed out route to DNA IV ES reactivated *src_area. src_node*

Long Syntax: DNAV.007 timed out route to DNA IV ES reactivated *src_area. src_node*

Description: A DNA endnode hello packet was received with a route that had been previously timed out in the OSI database.

DNAV.008

Level: P-TRACE

Short Syntax: DNAV.008 ISIS hello from distance vector router funnelled to DNA

Long Syntax: DNAV.008 ISIS hello from distance vector router funnelled to DNA

Description: An ISIS hello was received from a router running distance vector - the hello was passed to DNA IV to establish a router adjacency.

DNAV.009

Level: C-INFO

Short Syntax: DNAV.009 new 1-way adj w/ phase V dist vect router *sender* cir *number* net *network_name*

Long Syntax: DNAV.009 new 1-way adjacency with phase V distance vector router *sender* on circuit *number* network *network_name*

Description: We have just received an ISIS Hello Message from the specified router, but our address is not in the IS neighbor list of the hello message. We have a one-way adjacency with this router, it will not be two-way until our address is in the IS neighbor list.

DNAV.010

Level: C-INFO

Short Syntax: DNAV.010 Adj up; new phase V dist vect rtr *area*. *node* cir *number* net *network_name*

Long Syntax: DNAV.010 Adjacency up; new phase V distance vector router *area*. *node* circuit *number* network *network_name*

Description: There is now an adjacency with the specified router on one of the directly connected networks. Level 1 (and 2) Routing Messages will now be accepted from this node.

DNAV.011

Level: C-INFO

Short Syntax: DNAV.011 Adj dwn: dropped by phase V dist vect rtr *area*. *node*, cir *number* net *network_name*

Long Syntax: DNAV.011 Adjacency down, operator initiated: dropped by phase V distance vector router *area*. *node*, circuit *number* network *network_name*

Description: An ISIS Hello Message has been received from a router that we have an adjacency with, but does not include our address in the IS neighbor list. The adjacency will be taken down, and will not come back up until our address is in the IS neighbor list.

Cause: Adjacent router restarted.

Cause: One-way communication. While this router can receive packets from the adjacent router, the adjacent router cannot receive packets from this router.

Action: Ensure that there is two-way communication on the circuit.

DNAV.012

Level: UE-ERROR

Short Syntax: DNAV.012 pkt trans V to IV err - segmentation needed but not permitted

Long Syntax: DNAV.012 packet translation V to IV error - segmentation needed but not permitted

Description: An OSI data packet could not be translated to a DNA IV data packet because it needs to be segmented - segmentation of it is not permitted.

DNAV.013

Level: UE-ERROR

Short Syntax: DNAV.013 pkt trans V to IV err - src or dst addr not translatable

Long Syntax: DNAV.013 packet translation V to IV error - source or destination address not translatable

Description: An OSI data packet could not be translated to a DNA IV data packet because either the source or destination address is not Phase IV translatable.

DNAV.014

Level: UE-ERROR

Short Syntax: DNAV.014 Validation of phase IV info in ISIS hello PDU failed

Long Syntax: DNAV.014 Validation of phase IV info in ISIS hello PDU failed

Description: An ISIS hello PDU was received with an invalid Phase IV information option.

DNAV.015

Level: UE-ERROR

Short Syntax: DNAV.015 Phase IV hello from Phase V system dropped

Long Syntax: DNAV.015 Phase IV hello from Phase V system dropped

Description: A Phase IV hello PDU is dropped because it was sent by a Phase V system - adjacencies with Phase V systems are established using Phase V hellos.

DNAV.016

Level: UE-ERROR

Short Syntax: DNAV.016 L1 LSP from DNA system dropped - running dist vect at level 1

Long Syntax: DNAV.016 L1 LSP from DNA system dropped - running dist vect at level 1

Description: A level 1 link state packet received from a DNA system is dropped because this router is running distance vector at level 1.

DNAV.017

Level: UE-ERROR

Short Syntax: DNAV.017 L2 LSP from DNA system dropped - running dist vect at level 2

Long Syntax: DNAV.017 L2 LSP from DNA system dropped - running dist vect at level 2

Description: A level 2 link state packet received from a DNA system is dropped because this router is running distance vector at level 2.

DNAV.018

Level: UE-ERROR

Short Syntax: DNAV.018 ISIS hello dropped - nonmatching Phase IV areas

Long Syntax: DNAV.018 ISIS hello dropped - nonmatching Phase IV areas

Description: An ISIS hello PDU is dropped because the Phase IV area address in the area address option does not match this router's Phase IV area address.

DNAV.019

Level: C-INFO

Short Syntax: DNAV.019 Adj up; new DNA V endnode *area*. *node* cir *number* net *network_name*

Long Syntax: DNAV.019 Adjacency up; new DNA V endnode *area*. *node* circuit *number* network *network_name*

Description: There is now an adjacency with the specified DNA Phase V endnode on the specified network.

Cause: Received valid ISO ESIS hello message.

DNAV.020

Level: UE-ERROR

Short Syntax: DNAV.020 Trans DNIV pkt not forwarded - mapping of out adj ID *area*. *node* to SNPA add failed

Long Syntax: DNAV.020 Translated DECnet IV packet not forwarded - mapping of output adjacency's Phase IV ID *area*. *node* to an OSI SNPA address failed.

Description: The translation of a DECnet IV packet to a DECnet V packet failed because a mapping couldn't be found between the output adjacency's DECnet IV ID and an OSI SNPA address.

Cause: An end system adjacency doesn't exist in the OSI database for the next hop system.

DNAV.021

Level: UE-ERROR

Short Syntax: DNAV.021 verify fail on cir (*routing-circuit*)

Long Syntax: DNAV.021 verification failure on circuit (*routing-circuit*)

Description: There was a verification failure during link initialization on the circuit.

DNAV.022

Level: UE-ERROR

Short Syntax: DNAV.022 link init timeout on cir (*routing-circuit*)

Long Syntax: DNAV.022 link initialization timeout on circuit (*routing-circuit*)

Description: The link-initialization timer expired before the router successfully initialized the link.

DNAV.023

Level: UE-ERROR

Short Syntax: DNAV.023 init min timeout on cir(*routing-circuit*)

Long Syntax: DNAV.023 Initial Minimum Timer expired on circuit (*routing-circuit*)

Description: The Initial Minimum Timer expired before the router successfully initialized the link.

DNAV.024

Level: UE-ERROR

Short Syntax: DNAV.024 link init fail on cir (*routing-circuit*)

Long Syntax: DNAV.024 link initialization failure on circuit (*routing-circuit*)

Description: The link initialization failed on the circuit.

DNAV.025

Level: C-INFO

Short Syntax: DNAV.025 Adj up; new DNA IV VAXcluster alias *area*. *node* cir *number* net *network_name*

Long Syntax: DNAV.025 Adjacency up; new DNA IV VAXcluster alias *area*. *node* circuit *number* network *network_name*

Description: There is now a DNA Phase IV end node adjacency representing a VAXcluster alias address on the specified network.

Cause: The router received a valid DNA IV Level 1

Routing message, which advertises a VAXcluster alias address.

DNAV.026

Level: C-INFO

Short Syntax: DNAV.026 Adj dwn; DNA IV VAXcluster alias *area*. *node* cir *number* net *network_name*

Long Syntax: DNAV.026 Adjacency down; DNA IV VAXcluster alias *area*. *node* circuit *number* network *network_name*

Description: A DNA Phase IV end node adjacency representing a VAXcluster alias address went down.

Cause: The adjacency to the DNA IV router that was advertising the alias address timed out.

Cause: The router received a valid DNA IV Level 1 Routing message from the router that was advertising the alias address. The adjacency now advertises a different alias address or no alias address.

Chapter 35. Distance Vector Multicast Routing Protocol (DVM)

This chapter describes Distance Vector Multicast Routing Protocol (DVM) messages. For information on message content and how to use the message, refer to the Introduction.

DVM.001

Level: UE-ERROR

Short Syntax: DVM.001 Unknown DVMRP code from *IP_source*, code= *message_code*

Long Syntax: DVM.001 Received unknown DVMRP code from *IP_source*, code= *message_code*

Description: A DVMRP message was received from the specified source, however it has an unrecognized IGMP code value. The packet is discarded.

DVM.002

Level: UE-ERROR

Short Syntax: DVM.002 No matching VIF for pkt from *IP_source*, code = *message_code*

Long Syntax: DVM.002 No matching DVMRP interface for packet from *IP_source*, code= *message_code*

Description: A DVMRP message was received from the specified source, however, no matching DVMRP interface could be found. This probably indicates a configuration error (either in the source, or in the logging router). The packet is discarded.

DVM.003

Level: P-TRACE

Short Syntax: DVM.003 Rcvd DVMRP Report from *IP_source*

Long Syntax: DVM.003 Received DVMRP Report from *IP_source*

Description: A DVMRP report (routing update) has been received from the specified source. This is a normal, periodic event, and can cause additions to the DVMRP routing table.

DVM.004

Level: U-TRACE

Short Syntax: DVM.004 Rcvd DVMRP probe from *IP_source*

Long Syntax: DVM.004 Received DVMRP probe from *IP_source*

Description: A DVMRP probe has been received from the specified source. This is somewhat unusual, and should only happen when the DVMRP conversation on the interface is just beginning.

DVM.005

Level: UE-ERROR

Short Syntax: DVM.005 Rcvd bad DVMRP update from *IP_source*

Long Syntax: DVM.005 Received bad DVMRP update from *IP_source*

Description: A DVMRP update has been received from the specified source. The update was improperly formatted, and at least part of its contents were discarded.

DVM.006

Level: U-TRACE

Short Syntax: DVM.006 Add phyint *IP_interface_address* cost *cost* thresh *threshold*

Long Syntax: DVM.006 Add physical interface *IP_interface_address* cost *cost* thresh *threshold*

Description: DVMRP has been enabled on the specified physical interface, with the given cost and threshold parameters.

DVM.007

Level: U-TRACE

Short Syntax: DVM.007 Add tunnel *tunnel_source*-> *tunel_destination* cost *cost* thresh *threshold*

Long Syntax: DVM.007 Add tunnel *tunnel_source*-> *tunel_destination* cost *cost* thresh *threshold*

Description: A DVMRP tunnel has been configured between the given source and destination, with the specified cost and threshold parameters.

DVM.008

Level: U-TRACE

Short Syntax: DVM.008 Add MOSPF cost *cost* thresh *threshold*

Long Syntax: DVM.008 Add MOSPF cost *cost* thresh *threshold*

Description: Tunneling of DVMRP through the MOSPF cloud has been enabled, with the given cost and threshold paramters.

DVM.009

Level: U-TRACE

Short Syntax: DVM.009 Add/update route to *source_network* via *neighbor_IP_address*

Long Syntax: DVM.009 Add route to source *source_network* via neighbor *neighbor_IP_address*

Description: Processing a DVMRP update, or the fact that an interface came up, has caused us to either create or revise a routing table entry for the particular source.

DVM.010

Level: U-TRACE

Short Syntax: DVM.010 Delete route to *source_network*

Long Syntax: DVM.010 Delete route to source *source_network*

Description: A neighbor has informed us that the source is no longer reachable.

DVM.011

Level: U-TRACE

Short Syntax: DVM.011 Add neighbor *neighbor_IP_address*

Long Syntax: DVM.011 Add neighbor *neighbor_IP_address*

Description: A new DVMRP neighbor has been discovered, through the receipt of a probe or update message.

DVM.012

Level: U-TRACE

Short Syntax: DVM.012 Delete neighbor *neighbor_IP_address*

Long Syntax: DVM.012 Delete neighbor *neighbor_IP_address*

Description: Neighbor is no longer reachable. It has either timed out or its associated interface has gone down.

DVM.013

Level: C-TRACE

Short Syntax: DVM.013 Sending DVMRP probe to *neighbor_IP_address*, VIF: *VIF_index*

Long Syntax: DVM.013 Sending probe to *neighbor_IP_address*, VIF *VIF_index*

Description: Sent a DVMRP neighbor probe to the specified address.

DVM.014

Level: C-TRACE

Short Syntax: DVM.014 Sending DVMRP update to *neighbor_IP_address*, VIF: *VIF_index*

Long Syntax: DVM.014 Sending probe to *neighbor_IP_address*, VIF: *VIF_index*

Description: Sent a DVMRP routing update to the specified address.

DVM.015

Level: U-TRACE

Short Syntax: DVM.015 Route to *source_network* timed out

Long Syntax: DVM.015 Route to source *source_network* timed out

Description: Route to a particular source has timed out.

DVM.016

Level: U-TRACE

Short Syntax: DVM.016 Neighbor *neighbor_IP_address* timed out

Long Syntax: DVM.016 Neighbor *neighbor_IP_address* has timed out

Description: A neighbor has timed out. We did not get any updates from it lately.

DVM.017

Level: UI-ERROR

Short Syntax: DVM.017 No mem for source *source_network*

Long Syntax: DVM.017 No memory for source network *source_network*

Description: Either a) we don't have enough heap memory to allocate a DVMRP routing table entry or b) the IP routing table has overflowed. In any case, we cannot recognize the new source. If this source is a directly connected subnet, we won't be able to run IGMP on the subnet either.

DVM.018

Level: U-TRACE

Short Syntax: DVM.018 Added MOSPF route *source_network*

Long Syntax: DVM.018 Added MOSPF route *source_network*

Description: Started advertising a MOSPF route via DVMRP.

DVM.019

Level: U-TRACE

Short Syntax: DVM.019 Deleted MOSPF route *source_network*

Long Syntax: DVM.019 Deleted MOSPF route *source_network*

Description: Stopped advertising a MOSPF route via DVMRP.

DVM.020

Level: UI-ERROR

Short Syntax: DVM.020 No room for neighbor *neighbor_IP_address*

Long Syntax: DVM.020 No room for neighbor *neighbor_IP_address*

Description: There was no room to allocate the data structure for a new neighbor. DVMRP routes form the neighbor will be ignored.

DVM.021

Level: P-TRACE

Short Syntax: DVM.021 Packet rcvd from mis/unconfigured tunnel *source_IP_address*

Long Syntax: DVM.021 Packet received from mis/unconfigured tunnel *source_IP_address*

Description: A packet has been received via protocol 4 (IP encapsulation). The packet should be source by the other end of a tunnel. Either the tunnel has not been configured, or it has been configured to be source-routed instead of encapsulated.

DVM.022

Level: C-INFO

Short Syntax: DVM.022 Snt PRUNE (src= *source_net* grp= *group_address* life= *prune_lifetime*) to *neighbor_IP_address*, VIF= *VIF_index*

Long Syntax: DVM.022 Sent PRUNE (source= *source_net* group= *group_address*, lifetime= *prune_lifetime*) to *neighbor_IP_address*, VIF= *VIF_index*

Description: Sent a DVMRP PRUNE message to the specified IP address on the specified virutual interface. The packet contained the given source network, group address, and prune lifetime, in seconds. A PRUNE is being sent because this router has determined that receiving this traffic would be wasteful since this router will just discard it. The PRUNE informs the neighbors on the interface that multicast traffic, destined for this group address and sent from this source network, should not be forwarded to this router for next prune_lifetime number of seconds. Large amounts of wasteful multicast traffic are avoided as PRUNE information is exchanged among the routers forming the multicast DVMRP internet.

DVM.023

Level: U-INFO

Short Syntax: DVM.023 Snt GRAFT (src= *source_net* grp= *group_address*) to *neighbor_IP_address*, VIF= *VIF_index*

Long Syntax: DVM.023 Sent GRAFT (source= *source_net* group= *group_address*) to *neighbor_IP_address*, VIF= *VIF_index*

Description: Sent a DVMRP GRAFT message to the specified IP address on the specified virtual interface. The packet contained the given source network and group address. A GRAFT is being sent because this router has determined that receiving this multicast traffic, which was previously determined to be wasteful, is now no longer wasteful. The GRAFT informs the neighbors on the interface to cancel any previous PRUNE, if any, that the neighbor may have in effect for the specified group address, source network from this router.

DVM.024

Level: C-TRACE

Short Syntax: DVM.024 Snt GRAFT ACK (src= *source_net* grp= *group_address*) to *neighbor_IP_address*, VIF= *VIF_index*

Long Syntax: DVM.024 Sent GRAFT ACK (source= *source_net* group= *group_address*) to *neighbor_IP_address*, VIF= *VIF_index*

Description: Sent a DVMRP GRAFT ACK message to the specified IP address on the specified virtual interface. The packet contained the given source network and group address. A GRAFT ACK message is used to acknowledge a GRAFT message.

DVM.025

Level: C-INFO

Short Syntax: DVM.025 Rec PRUNE (src= *source_net* grp= *group_address* life= *prune_lifetime*) frm *neighbor_IP_address*, VIF= *VIF_index*, act= *action_taken*

Long Syntax: DVM.025 Rec PRUNE (source= *source_net* group= *group_address* lifetime= *prune_lifetime*) frm *neighbor_IP_address*, VIF= *VIF_index* action= *action_taken*

Description: Received a PRUNE, containing the

displayed source, group, and lifetime values, from an IP neighbor on a virutual interface. The PRUNE was accepted. The only possible action taken values is: (1) ″accepted″, meaning that the PRUNE was accepted.

DVM.026

Level: UI-ERROR

Short Syntax: DVM.026 Rec PRUNE (src= *source_net* grp= *group_address* life= *prune_lifetime*) frm *neighbor_IP_address*, VIF= *VIF_index*, rej= *action_taken*

Long Syntax: DVM.026 Rec PRUNE (source= *source_net* group= *group_address* lifetime= *prune_lifetime*) frm *neighbor_IP_address*, VIF= *VIF_index* reject= *action_taken*

Description: Received a PRUNE, containing the displayed source, group, and lifetime values, from an IP neighbor on a virutual interface. The PRUNE was not accepted because of the displayed reason. Possible action taken values are: (1) ″pkt trunc″ - the PRUNE packet was too short, (2) ″unknown nbr″ - the sender was not a known DVMRP neighbor, (3) ″unreach src net″ - the router can not route to the source_net, (4) ″no mem for Mfcache″ - the router had no memory to build a multicast forwarding entry, (5) ″no DVMRP rte″ - the router has no DVMRP route to the this source_net, (6) ″not from dependent nbr″ - the source_net was not a downstream neighbor to which the router forwards, (7) ″no mem for group″ - the router had no memory to build a group entry, (8) ″no mem for prune″ - the router had no memory to build a prune entry,

---

DVM.027

Level: UI-ERROR

Short Syntax: DVM.027 No mem for prune mgnt, grp *neighbor_IP_address*

Long Syntax: DVM.027 No memory for prune management, group *neighbor_IP_address*

Description: There was no room to allocate a group pr prune data structure to manage the pruning and grafting of the specified group. Some functinonaly for prune and graft logic will be lost until memory can be allocated. If this message is not persistent, then the situation has recovered; otherwise DVMRP functionality is being lost.

DVM.028

Level: UI-ERROR

Short Syntax: DVM.028 No buf for msg *message_code*, frm *src_IP*

Long Syntax: DVM.028 No buffer for message *message_code*, frm *src_IP*

Description: There was io buffer available to send a DVMRP message. This means that a DVMRP message that should have been sent is not sent. If this message is not persistent, the situation has recovered; otherwise DVMRP functionality is being lost.

DVM.029

Level: C-INFO

Short Syntax: DVM.029 Rec GRAFT (src= *source_net* grp= *group_address*) frm *neighbor_IP_address*, VIF= *VIF_index*, act= *action_taken*

Long Syntax: DVM.029 Rec GRAFT (source= *source_net* group= *group_address*) frm *neighbor_IP_address*, VIF= *VIF_index* action= *action_taken*

Description: Received a GRAFT, containing the displayed source and group values, from an IP neighbor on a virutual interface. Possible action taken values are: (1) ″accepted″ - the GRAFT matched a PRUNE in the router's prune database, (2) ″pkt trunc″ - the PRUNE packet was too short, (3) ″unknown nbr″ - the sender was not a known DVMRP neighbor, (4) ″unreach src net″ - the router can not route to the source_net, (5) ″no mem for Mfcache″ - the router had no memory to build a multicast forwarding entry, (6) ″no DVMRP rte″ - the router has no DVMRP route to the this source_net, (7) ″not from dependent nbr″ - the source_net was not a downstream neighbor to which the router forwards, (8) ″no group found″ - the router did not find a group entry for this group. group entry, (9) ″no prune found″ - the router did not find a matching prune. prune entry, Note that although many of these cases sound unsuccessful, a GRAFT ACK is sent for all of these except ″pkt trunc″ and ″unknown nbr″.

DVM.030

Level: C-INFO

Short Syntax: DVM.030 Rec GRAFT ACK (src= *source_net* grp= *group_address*) frm *neighbor_IP_address*, VIF= *VIF_index*, act= *action_taken*

Long Syntax: DVM.030 Rec GRAFT ACK (source= *source_net* group= *group_address*) frm *neighbor_IP_address*, VIF= *VIF_index* action= *action_taken*

Description: Received a GRAFT ACK, containing the displayed source and values, from an IP neighbor on a virutual interface. Possible action taken values are: (1) ″accepted″ - the GRAFT ACK successfully matched a GRAFT on this router that needed acknowledgment, (2) ″pkt trunc″ - the GRAFT ACK was too short, (3) ″unknown nbr″ - the sender was not a known DVMRP neighbor, (4) ″unreach src net″ - the router can not route to the source_net, (5) ″no mem for Mfcache″ - the router had no memory to build a multicast forwarding entry, (6) ″no DVMRP rte″ - the router has no DVMRP route to the this source_net, (7) ″no group found″ - the router did not find a group entry for this group. (8) ″not upstream nbr″ - the GRAFT ACK was not from the upstream neighbor. The normal case is accepted. But

other values can norally occur as tables change dynamically. If this message action persistently is not the accepted case, there may be a protocol problem. The problem is probably on the router sending the GRAFT ACK.

DVM.031

Level: C-TRACE

Short Syntax: DVM.031 Unsupported Default Route from *neighbor_IP_address*, VIF= *VIF_index*

Long Syntax: DVM.031 Received a default route entry, which is not supported, from *neighbor_IP_address*, VIF= *VIF_index*

Description: The router received a default route advertisement on the specified virtual interface. This DVMRP implementation does not support default routes for multicast.

Chapter 36. DS

This chapter describes DS messages. For information on message content and how to use the message, refer to the Introduction.

DS.001

Level: C-INFO

Short Syntax: DS.001 Initializing DS; status= *init_status*.

Long Syntax: DS.001 Initializing DS function; result status is *init_status*.

Description: The DS init routine is called to perform initialization functions. The result status is displayed (NOT CONFIGURED, STARTED, CONFIG ERROR).

DS.002

Level: C-INFO

Short Syntax: DS.002 DS stopped at clock *time_now*.

Long Syntax: DS.002 DS function is stopped by operator at system clock *time_now*.

Description: The operator stopped DS function using op-console command.

DS.003

Level: U-INFO

Short Syntax: DS.003 DS Init

Long Syntax: DS.003 DS initialization

Description: This message is printed when DiffServe has been enabled.

DS.004

Level: U-INFO

Short Syntax: DS.004 DS Int up; net: *net_number*; net-type: *net-type* speed: *net_speed*

Long Syntax: DS.004 DiffServ interface up on on net *net_number* of type *net-type* with speed *net_speed*

Description: This message is printed when DiffServe initalizes on interface.

DS.005

Level: U-INFO

Short Syntax: DS.005 DS net dn; net *net_number*

Long Syntax: DS.005 DiffServ interface down on net *net_number*

Description: This message is printed when DiffServe interface goes down.

DS.006

Level: U-INFO

Short Syntax: DS.006 in *net_number*, out *net_number*, strm_id *stream_id*, Got Neg Pol Ptr

Long Syntax: DS.006 in net *net_number*, out net *net_number*, stream id *stream_id*. Got Negotiated Policy pointer in Decision Object.

Description: This message is printed when Decision Object has Negotiated Policy pointer.

DS.007

Level: UI-ERROR

Short Syntax: DS.007 in *net_number*, out *net_number*, NULL Decision pntr from CPE

Long Syntax: DS.007 in net *net_number*, out net *net_number*. NULL Decision pointer returned from CPE

Description: This message is printed when CPE returns OK with NULL decision pointer.

DS.008

Level: UI-ERROR

Short Syntax: DS.008 Run out buckets in free pools

Long Syntax: DS.008 Run out buckets in istream, ostream or timer fbl's.

Description: This message is printed in ds_add_xxx routines.

DS.009

Level: UI-ERROR

Short Syntax: DS.009 Delete fixed streams

Long Syntax: DS.009 Delete fixed streams such as control, resv and best-effort.

Description: This message is printed in ds_delete_xxx routines.

DS.010

Level: UI-ERROR

Short Syntax: DS.010 Fail to add an istream id net

Long Syntax: DS.010 Fail to add an istream id net

Description: This message is printed in ds_add_istream() routine.

DS.011

Level: UI-ERROR

Short Syntax: DS.011 Fail to add an ostream net

Long Syntax: DS.011 Fail to add an ostream net

Description: This message is printed in ds_add_ostream() routine.

DS.012

Level: UI-ERROR

Short Syntax: DS.012 Fail to chain an istream id net and an ostream net

Long Syntax: DS.012 Fail to chain an istream id net and an ostream net

Description: This message is printed in ds_io_chain() routine.

DS.013

Level: UI-ERROR

Short Syntax: DS.013 Fail to add a timer entry net

Long Syntax: DS.013 Fail to add a timer entry net

Description: This message is printed in ds_reg_timer() routine.

DS.014

Level: UI-ERROR

Short Syntax: DS.014 Fail to un-chain an istream net and an ostream net

Long Syntax: DS.014 Fail to un-chain an istream net and an ostream net

Description: This message is printed in ds_io_unchain() routine.

DS.015

Level: UI-ERROR

Short Syntax: DS.015 Fail to delete a timer entry net

Long Syntax: DS.015 Fail to delete a timer entry net

Description: This message is printed in ds_reg_timer() routine.

DS.016

Level: UI-ERROR

Short Syntax: DS.016 Fail to delete an istream id net

Long Syntax: DS.016 Fail to delete an istream id net

Description: This message is printed in ds_del_istream() routine.

DS.017

Level: UI-ERROR

Short Syntax: DS.017 Fail to delete an ostream net

Long Syntax: DS.017 Fail to delete an ostream net

Description: This message is printed in ds_del_ostream() routine.

DS.018

Level: UI-ERROR

Short Syntax: DS.018 Inconsistant qid during stream checks net

Long Syntax: DS.018 The new stream has a different queue id with the existing one net

Description: This message is printed in ds_check_spec() routine.

DS.019

Level: UI-ERROR

Short Syntax: DS.019 Net q no out buff need curr max

Long Syntax: DS.019 Net queue has no output buffer needed current alloc max alloc

Description: This message is printed in ds_check_spec() and ds_add_ostream_buf routines

DS.020

Level: UI-ERROR

Short Syntax: DS.020 Unknown service type

Long Syntax: DS.020 Unknown service type for the new sessions

Description: This message is printed in ds_check_spec() routine.

DS.021

Level: UI-ERROR

Short Syntax: DS.021 Net q no in buff need curr max

Long Syntax: DS.021 Net Queue has no input buffer needed current alloc max alloc

Description: This message is printed in

ds_check_spec() and ds_add_istream_buf routines.

DS.022

Level: UI-ERROR

Short Syntax: DS.022 Bad stream type during timer checking

Long Syntax: DS.022 Bad stream type (that is, not a regular one) during timer checking

Description: This message is printed in ds_check_timeout() routine.

DS.023

Level: UI-ERROR

Short Syntax: DS.023 Bad stream status during timer checking

Long Syntax: DS.023 Bad stream type (that is, not full nor idle) during timer checking

Description: This message is printed in ds_check_timeout() routine.

DS.024

Level: U-INFO

Short Syntax: DS.024 dsact: QPri= *que_pri*, RateT= *rate_type*, Rate= *rate*

Long Syntax: DS.024 DiffServ action: Queue Priority= *que_pri*, Rate Type= *rate_type* and Rate= *rate*

Description: This message is printed when parsing the DiffServ action from Decision Object.

DS.025

Level: UI-ERROR

Short Syntax: DS.025 Init DS, net *net_number*: dlci: *init_status*; status= .

Long Syntax: DS.025 Initializing DS function on net *net_number*, dlci *init_status*; result status is .

Description: This message is printed when DiffServe initalizes on interface.

DS.026

Level: U-INFO

Short Syntax: DS.026 DS Int dn; net *net_number*; not enabled.

Long Syntax: DS.026 DiffServ interface on net *net_number*; not PPP/FR or not enabled or not configured.

Description: This message is printed when DiffServe initalizes on interface.

DS.027

Level: UI-ERROR

Short Syntax: DS.027 Bad rc *return_code* from CPE

Long Syntax: DS.027 Return Code from CPE is not CPE_OK, rc = *return_code*

Description: This message is printed when CPE returns with bad return code.

DS.028

Level: UI-ERROR

Short Syntax: DS.028 dsapi: in *net_number*, out *net_number*. *error_msg*.

Long Syntax: DS.028 DiffServ API: Error from DS API functions; input net *net_number*, output net *net_number*; status = *error_msg*.

Description: This message is printed when there is an error in DSAPI functions.

DS.029

Level: UI-ERROR

Short Syntax: DS.029 dsapi: out *net_number*. *error_msg*

Long Syntax: DS.029 DiffServ API: Error from DS API functions; output net *net_number*; status = *error_msg*.

Description: This message is printed when there is an error in DSAPI functions.

DS.030

Level: U-INFO

Short Syntax: DS.030 dsapi: in *net_number*, out *net_number*. Added Stream *stream_ptr*

Long Syntax: DS.030 DiffServ API: Stream added from in net *net_number* to out net *net_number*. Stream id *stream_ptr*.

Description: This message is printed when a DS stream is added.

DS.031

Level: U-INFO

Short Syntax: DS.031 ds_fr_out: ovifp *ovifp*, dlci *dlci*, istream id *istream_id*, Got istream, out vif, and out dlci from iob

Long Syntax: DS.031 ds_fr_out: Got out vif Ptr *ovifp*, out dlci# *dlci* and istream_id *istream_id* from iob.

Description: This message is printed when Frame Relay calls ds_fr_out.

DS.032

Level: U-INFO

Short Syntax: DS.032 ds_fr_nget: dlci *dlci*, pkt lgt *lgt*, q weight *weight* Xmitting pkt..

Long Syntax: DS.032 ds_fr_nget: Xmitting packet from DLCI *dlci*, length *lgt*, from q weight *weight*.

Description: This message is printed when Frame Relay traffic shaper gives the go-ahead for a packet.

DS.033

Level: U-INFO

Short Syntax: DS.033 ds_fr_nget: dlci *dlci*, CIR *cir*, Not Xmitting pkt..

Long Syntax: DS.033 ds_fr_nget: Packet from dlci *dlci*, CIR *cir* (Bytes/s) did not pass FR traffic shaping.

Description: This message is printed when Frame Relay traffic shaper fails on a packet.

DS.034

Level: UI-ERROR

Short Syntax: DS.034 ds_update_stream: neg alloc_buff *alloc_buff*, last *last-change* sp *streamptr* type *stream-type* reset

Long Syntax: DS.034 ds_update_stream: negative alloc_buff amount *alloc_buff* last change *last-change* streamptr *streamptr* type *stream-type* reset to zero.

Description: This message is printed when the alloc buffer amount goes negative.

DS.035

Level: U-INFO

Short Syntax: DS.035 ds_nout: In stream: in_shr_buf *in_shr_buf*, in_extra *in_extra*

Long Syntax: DS.035 ds_nout: Input stream: in_shr_buf *in_shr_buf*, in_extra *in_extra*.

Description: diffserv_out or ds_fr_nout dropped a packet D_OVFL instream

---

DS.036

Level: U-INFO

Short Syntax: DS.036 ds_nout: Output stream: out_max_buf *out_max_buf*, out_curr_buff *out_curr_buf*, lgt *lgt*

Long Syntax: DS.036 ds_nout: Output stream: out_max_buf *out_max_buf*, out_curr_buff *out_curr_buf*, length *lgt*

Description: diffserv_out or ds_fr_nout dropped a packet D_OVFL out_max_buff used

---

DS.037

Level: U-INFO

Short Syntax: DS.037 ds_nout: Output stream: out_shr_buf *out_max_buf*, out_extra *out_extra*, buff_diff *buff_diff*

Long Syntax: DS.037 ds_nout: Output stream: out_shr_buf *out_max_buf*, out_extra *out_extra*, buff_diff *buff_diff*

Description: diffserv_out or ds_fr_nout dropped a packet D_OVFL out_share_buff used

---

DS.038

Level: UI-ERROR

Short Syntax: DS.038 dsapi: istr *istream_number* buff *curr_buff*: *error_msg*.

Long Syntax: DS.038 DiffServ API: Error from DS API functions; istream *istream_number*, curr_buff *curr_buff*; status = *error_msg*.

Description: This message is printed when outstanding buff exists at istream del time.

DS.039

Level: U-INFO

Short Syntax: DS.039 ds_api: Sid: *StreamId Queue* Policer: rate: *averageRate* peak: *peakRate* token: *TokenBucketSize* action: *PolicerAction*

Long Syntax: DS.039 ds_api: Stream ID: *StreamId Queue* Policer: avg rate: *averageRate* peak rate: *peakRate* token bucket (B): *TokenBucketSize* action: *PolicerAction*

Description: report policer parameters and actions

Chapter 37. Dynamic Host Configuration Protcol for IPv6 Relay Agent (DHCPv6)

This chapter describes Dynamic Host Configuration Protcol for IPv6 Relay Agent (DHCPv6) messages. For information on message content and how to use the message, refer to the Introduction.

DHC6.001

Level: U-INFO

Short Syntax: DHC6.001 DHCPv6 Relay Agent Enabled

Long Syntax: DHC6.001 The DHCPv6 Relay Agent has been enabled

Description: This message is generated when the DHCPv6 Relay Agent subsystem is enabled, due to the use of the Stateful Configuration option in the Neighbor Discovery Protocol for IPv6.

DHC6.002

Level: U-INFO

Short Syntax: DHC6.002 DHCPv6 Relay Agent Disabled

Long Syntax: DHC6.002 The DHCPv6 Relay Agent has been disabled

Description: This message is generated when the DHCPv6 Relay Agent subsystem is disabled, due to the discontinued use of the Stateful Configuration option in the Neighbor Discovery Protocol for IPv6.

DHC6.003

Level: UI-ERROR

Short Syntax: DHC6.003 pkt rcvd, bad nt, src *Packet_source_address*

Long Syntax: DHC6.003 The DHCPv6 Relay Agent received a pkt internally, src *Packet_source_address*

Description: The DHCPv6 Relay Agent received a packet from an unspecified interface. The Relay Agent requires the receiving interface to be indicated prior to the packet being processed.

DHC6.004

Level: U-INFO

Short Syntax: DHC6.004 rcv Sol from host *Packet_source_address*

Long Syntax: DHC6.004 The DHCPv6 Relay Agent received a Solicit Message from *Packet_source_address*

Description: The DHCPv6 Relay Agent has received a Solicit Message from the specified client. The address given is the IPv6 link local address of the client making the request.

DHC6.005

Level: U-INFO

Short Syntax: DHC6.005 fwd Sol, *Packet_source_address* --> *Packet_dest_address*

Long Syntax: DHC6.005 The DHCPv6 Relay Agent forwarded a Solicit Message from *Packet_source_address* to *Packet_dest_address*

Description: The DHCPv6 Relay Agent has forwarded a Solicit Message from the specified client.

DHC6.006

Level: U-INFO

Short Syntax: DHC6.006 rcv Adv from server *Packet_source_address*

Long Syntax: DHC6.006 The DHCPv6 Relay Agent received an Advertisement Message from *Packet_source_address*

Description: The DHCPv6 Relay Agent has received an Advertisement Message from the specified server.

DHC6.007

Level: U-INFO

Short Syntax: DHC6.007 fwd Adv, *Packet_source_address* --> *Packet_dest_address*

Long Syntax: DHC6.007 The DHCPv6 Relay Agent forwarded an Advertisement Message from *Packet_source_address* to *Packet_dest_address*

Description: The DHCPv6 Relay Agent has forwarded an Advertisement Message from the specified server to the specified client.

DHC6.008

Level: U-INFO

Short Syntax: DHC6.008 rcv Req from host *Packet_source_address*

Long Syntax: DHC6.008 The DHCPv6 Relay Agent received a Request Message from *Packet_source_address*

Description: The DHCPv6 Relay Agent has received a Request Message from the specified client.

DHC6.009

Level: UI-ERROR

Short Syntax: DHC6.009 dsc Req from *Packet_source_address*, not link local

Long Syntax: DHC6.009 The DHCPv6 Request Message from *Packet_source_address* did not use a link local address

Description: The DHCPv6 Relay Agent has discarded a Request Message that did not use a link local address as the source address in the IPv6 header.

DHC6.010

Level: UI-ERROR

Short Syntax: DHC6.010 dsc Req, hdr *Packet_source_address* != data *Packet_data_address*

Long Syntax: DHC6.010 The DHCPv6 Request Message from *Packet_source_address* had a bad client address *Packet_data_address*

Description: The DHCPv6 Relay Agent has discarded a Request Message that did not specify the same link local address in the client address field of the request as in the IPv6 header source address.

DHC6.011

Level: UI-ERROR

Short Syntax: DHC6.011 dsc Req, pkt rcv wrong nt *IPv6_interface* for relay *Packet_data_address*

Long Syntax: DHC6.011 The DHCPv6 Request Message was received on the wrong net (*IPv6_interface*) for relay *Packet_data_address*

Description: The DHCPv6 Relay Agent has discarded a Request Message that was received on the wrong interface for the relay agent address specified in the request.

DHC6.012

Level: UI-ERROR

Short Syntax: DHC6.012 dsc Req, srvr not specified

Long Syntax: DHC6.012 The DHCPv6 Request Message did not have the Server Address Present bit on

Description: The DHCPv6 Relay Agent has discarded a Request Message that was received because the Server Address Present bit was not turned on in the request.

DHC6.013

Level: U-INFO

Short Syntax: DHC6.013 fwd Req, *Packet_source_address* --> *Packet_dest_address*

Long Syntax: DHC6.013 The DHCPv6 Relay Agent forwarded a Request Message from *Packet_source_address* to *Packet_dest_address*

Description: The DHCPv6 Relay Agent has forwarded a Request Message from the specified client to the specified server.

DHC6.014

Level: U-INFO

Short Syntax: DHC6.014 rcv Rpl from srvr *Packet_source_address*

Long Syntax: DHC6.014 The DHCPv6 Relay Agent received a Reply Message from *Packet_source_address*

Description: The DHCPv6 Relay Agent has received a Reply Message from the specified server.

DHC6.015

Level: UI-ERROR

Short Syntax: DHC6.015 dsc Rpl, link local addr not present

Long Syntax: DHC6.015 The DHCPv6 Reply Message did not have the Link Local Address Present bit on

Description: The DHCPv6 Relay Agent has discarded a Reply Message that was received because the Link Local Address Present bit was not turned on in the reply.

DHC6.016

Level: UI-ERROR

Short Syntax: DHC6.016 dsc Rpl, clnt addr not link local (*Packet_data_address*)

Long Syntax: DHC6.016 The DHCPv6 Reply did not have a link local address in the client field (*Packet_data_address*)

Description: The DHCPv6 Relay Agent has discarded a Reply Message that was received because the client address specified in the packet was not a link local address.

DHC6.017

Level: U-INFO

Short Syntax: DHC6.017 fwd Rpl, *Packet_source_address* --> *Packet_dest_address*

Long Syntax: DHC6.017 The DHCPv6 Relay Agent

forwarded a Reply Message from *Packet_source_address* to *Packet_dest_address*

Description: The DHCPv6 Relay Agent has forwarded a Reply Message from the specified server to the specified client.

DHC6.018

Level: UI-ERROR

Short Syntax: DHC6.018 dsc pkt, unknown msg *msg_type* from *Packet_data_address*

Long Syntax: DHC6.018 The DHCPv6 Relay Agent received an unrecognized message (*msg_type*) from *Packet_data_address*

Description: The DHCPv6 Relay Agent has discarded a message with an unknown message type.

DHC6.019

Level: UI-ERROR

Short Syntax: DHC6.019 No mem for DHCPv6 servers

Long Syntax: DHC6.019 There is not enough memory to allocate storage for the list of DHCPv6 servers

Description: This message is generated when not enough memory can be allocated to store the list of DHCPv6 servers configured. The box will operate in multicast mode.

DHC6.020

Level: U-INFO

Short Syntax: DHC6.020 DHCPv6 Relay Agent not Enabled, IPv6 unavailable

Long Syntax: DHC6.020 The DHCPv6 Relay Agent has not been enabled, no IPv6 addresses configured

Description: This message is generated when the user tries to enable the DHCPv6 Relay Agent subsystem and there are no configured IPv6 addresses.

Chapter 38. Easy Start Functions (EZ)

This chapter describes Easy Start Functions (EZ) messages. For information on message content and how to use the message, refer to the Introduction.

EZ.001

Level: ALWAYS

Short Syntax: EZ.001 Starting.

Long Syntax: EZ.001 Starting.

Description: EasyStart process has begun.

EZ.002

Level: ALWAYS

Short Syntax: EZ.002 Changed one or more cfg params.

Long Syntax: EZ.002 Changed one or more configuration parameters.

Description: EasyStart changed a data link (e.g., PPP to FR), or changed a data link parameter in permanent configuration. EasyStart restarts for the changes to take effect.

EZ.003

Level: ALWAYS

Short Syntax: EZ.003 Bootp failed.

Long Syntax: EZ.003 Called bootp client and it failed

Description: EasyStart called BOOTP and it failed either because there is no BOOTP server on the attached (working) segment or because you did not configure the BOOTP server correctly.

EZ.004

Level: ALWAYS

Short Syntax: EZ.004 Rcvd boot info: ipAddr *ipAddr*, ipMask *ipMask* on intf *interfaceNumber*

Long Syntax: EZ.004 Received boot info: IPaddr: *ipAddr*, mask: *ipMask* on interface: *interfaceNumber*

Description: EasyStart called BOOTP and received necessary information to perform a TFTP download of needed configuration parameters. EasyStart will update the IP configuration with an IP address and a mask. Then, EasyStart will reboot the system for the changes to take effect.

EZ.005

Level: ALWAYS

Short Syntax: EZ.005 TFTP failed. Backing up to device configuration step.

Long Syntax: EZ.005 TFTP failed. Backing up to device configuration step.

Description: EasyStart TFTP transfer failed. EasyStart will go back to the device configuration step and try again.

EZ.006

Level: ALWAYS

Short Syntax: EZ.006 All dlinks/parameters tried but failed; resetting to def values.

Long Syntax: EZ.006 All datalinks and parameters tried but failed; resetting; restarting.

Description: The router tried all data links and parameters but failed. EasyStart will reset the configuration and start from the beginning.

EZ.007

Level: ALWAYS

Short Syntax: EZ.007 Waiting up to *seconds* seconds for devices to pass self-test.

Long Syntax: EZ.007 Waiting up to *seconds* seconds for devices to pass self-test.

Description: EasyStart is waiting for devices to pass self-test. The result is to have the device in the up or down state. Since some devices may come up quickly, there is a variable timer to avoid waiting.

EZ.008

Level: ALWAYS

Short Syntax: EZ.008 TFTP transfer completed successfully. *** EasyStart Completed Successfully ***

Long Syntax: EZ.008 TFTP transfer completed successfully.

Description: EasyStart succeeded. The router is restarting to the operational configuration which was downloaded.

EZ.009

Level: ALWAYS

Short Syntax: EZ.009 *** Restarting Router ***

Long Syntax: EZ.009 Restarting router

Description: Parameters have changed. EasyStart is restarting to have the changes take effect.

Chapter 39. End System Intermediate-System Protocol (ESIS)

This chapter describes End System Intermediate-System Protocol (ESIS) messages. For information on message content and how to use the message, refer to the Introduction.

ESIS.001

Level: UE-ERROR

Short Syntax: ESIS.001 ESIS input que ovflw

Long Syntax: ESIS.001 ESIS input queue overflow

Description: The ESIS task input queue has overflowed, packet is dropped.

ESIS.002

Level: UE-ERROR

Short Syntax: ESIS.002 rcvd incmplt pkt

Long Syntax: ESIS.002 received incomplete packet

Description: A packet fragment recognized as an ESIS packet was received.

ESIS.003

Level: UE-ERROR

Short Syntax: ESIS.003 rcvd pkt bad chksm= *pkt_chksum*

Long Syntax: ESIS.003 received packet with a bad checksum = *pkt_chksum*

Description: An ESIS packet was received but had a bad checksum.

ESIS.004

Level: UE-ERROR

Short Syntax: ESIS.004 rcvd pkt bad vers # = *version_number*

Long Syntax: ESIS.004 received packet with a bad version number (vers = *version_number*)

Description: An ESIS packet was received but had a bad or unsupported version number.

ESIS.005

Level: UE-ERROR

Short Syntax: ESIS.005 rcvd pkt bad typ # = *type_field*

Long Syntax: ESIS.005 received packet with a bad type field (vers = *type_field*)

Description: An ESIS packet was received but had a bad or unsupported type field.

ESIS.006

Level: UE-ERROR

Short Syntax: ESIS.006 no iob avail to snd hello

Long Syntax: ESIS.006 no i/o buffer available to send hello

Description: An attempt to send an ESIS hello failed because of a lack of system i/o buffers.

ESIS.007

Level: UE-ERROR

Short Syntax: ESIS.007 cnnt snt hello pkt hndlr err

Long Syntax: ESIS.007 cannot send a hello packet, handler error

Description: An ESIS hello packet coundn't be sent because of a handler error.

ESIS.008

Level: P-TRACE

Short Syntax: ESIS.008 sent hello *source_NSAP* on int *interface_#*

Long Syntax: ESIS.008 sent hello packet with source nsap *source_NSAP* on int *interface_#*

Description: An ESIS hello packet was sent out on an interface.

ESIS.009

Level: UE-ERROR

Short Syntax: ESIS.009 rcvd hello packet with a bad header

Long Syntax: ESIS.009 rcvd hello packet with a bad header

Description: Received hello packet with a holding time or reserved field.

ESIS.010

Level: UE-ERROR

Short Syntax: ESIS.010 rcvd hello bad nsap *source_NSAP*

Long Syntax: ESIS.010 received hello with bad nsap *source_NSAP*

Description: An ESIS hello packet was received with a bad nsap or one that overran the packet.

ESIS.011

Level: UE-ERROR

Short Syntax: ESIS.011 rcvd hello pkt bad opt

Long Syntax: ESIS.011 received packet with a bad optional parameter

Description: An ESIS CLNP data packet was received with bad option parameter(s).

ESIS.012

Level: P-TRACE

Short Syntax: ESIS.012 rcvd hello from *source_NSAP* int *interface* net *network_name*

Long Syntax: ESIS.012 rcvd hello packet with source nsap *source_NSAP* on int *interface*, net *network_name*

Description: An ESIS hello packet was received on the specified interface.

ESIS.013

Level: UE-ERROR

Short Syntax: ESIS.013 rcvd hello unsp dom src *source_NSAP*

Long Syntax: ESIS.013 rcvd hello packet unsupported domain *source_NSAP*

Description: An ESIS hello packet was received with an unrecognized IDI.

ESIS.016

Level: UE-ERROR

Short Syntax: ESIS.016 tmd out rte reac *source_NSAP*

Long Syntax: ESIS.016 timed out route reactivated *source_NSAP*

Description: An ESIS hello packet was received with a route that had been previously timed out.

ESIS.017

Level: UE-ERROR

Short Syntax: ESIS.017 no rsrc to snd rdrct

Long Syntax: ESIS.017 no resources to send redirect

Description: An ESIS redirect packet could not be sent due to a lack of resources.

ESIS.018

Level: UE-ERROR

Short Syntax: ESIS.018 rdrct nt snt hndlr err

Long Syntax: ESIS.018 redirect not sent, handler error

Description: An ESIS redirect packet could not be sent due to a handler error.

ESIS.019

Level: P-TRACE

Short Syntax: ESIS.019 sent rdrct to: *dest_NSAP*

Long Syntax: ESIS.019 sent redirect packet to: *dest_NSAP*

Description: An ESIS redirect packet was sent out on an interface.

ESIS.020

Level: UE-ERROR

Short Syntax: ESIS.020 tmd out rte *source_NSAP*

Long Syntax: ESIS.020 timed out route *source_NSAP*

Description: An ESIS hello route has been timed out.

ESIS.021

Level: UI_ERROR

Short Syntax: ESIS.021 Unable to allocate resources for a new ES adjacency

Long Syntax: ESIS.021 Unable to allocate resources for a new ES adjacency

Description: We were unable to get an adjacency structure for a new end system adjacency.

ESIS.022

Level: UE_ERROR

Short Syntax: ESIS.022 hello PDU dropped, rcvd over p-to-p cir *cct_num*

Long Syntax: ESIS.022 hello PDU dropped, received over point-to-point circ *cct_num*

Description: An ESIS hello PDU was received over a point-to-point circuit - the packet was dropped because

ESIS does not run over point-to-point circuits.

ESIS.023

Level: UE_ERROR

Short Syntax: ESIS.023 hello PDU dropped, no matching area address

Long Syntax: ESIS.023 ESIS hello PDU dropped, no matching area address

Description: An ESIS hello PDU was dropped because the area address portion of its source NSAP didn't match one of the router's manual area addresses.

ESIS.024

Level: P-TRACE

Short Syntax: ESIS.024 dropped hello from *source_NSAP* int *interface* net *network_name* manual ES adjacency exists

Long Syntax: ESIS.024 dropped hello packet with source nsap *source_NSAP* on int *interface*, net *network_name* - manual ES adjacency exists

Description: An ESIS hello packet was dropped on the specified interface because a manual adjacency exists for the ES.

Chapter 40. Environment Functions (ENV)

This chapter describes Environment Functions (ENV) messages. For information on message content and how to use the message, refer to the Introduction.

ENV.001

Level: C-TRACE

Short Syntax: ENV.001 current temp *temperature_celsius*C (*temperature_fahrenheit*F)

Long Syntax: ENV.001 Current ambient temperature: *temperature_celsius*C (*temperature_fahrenheit*F)

Description: The router generates this message each time it recalculates the current ambient temperature.

ENV.002

Level: U-TRACE

Short Syntax: ENV.002 hi temp thresh active: *threshold_temperature_celsius*C (*threshold_temperature_fahrenheit*F)

Long Syntax: ENV.002 High temperature threshold is active. Threshold: *threshold_temperature_celsius*C (*threshold_temperature_fahrenheit*F)

Description: Trace message indicating that the router passed the high temperature threshold and the high temperature threshold is active. The router generates this message each time it recalculates the current ambient temperature and the current ambient temperature surpasses the high temperature threshold.

ENV.003

Level: U-TRACE

Short Syntax: ENV.003 low temp thresh active: *threshold_temperature_celsius*C (*threshold_temperature_fahrenheit*F)

Long Syntax: ENV.003 Low temperature threshold is active. Threshold: *threshold_temperature_celsius*C (*threshold_temperature_fahrenheit*F)

Description: Trace message indicating that the router passed the low temperature threshold and the low temperature threshold is active. The router generates this message each time it recalculates the current ambient temperature and the current ambient temperature is below the low temperature threshold.

ENV.004

Level: UE-ERROR

Short Syntax: ENV.004 hi temp thresh exceeded: *threshold_temperature_celsius*C (*threshold_temperature_fahrenheit*F)

Long Syntax: ENV.004 High temperature threshold has been exceeded. Threshold: *threshold_temperature_celsius*C (*threshold_temperature_fahrenheit*F)

Description: Trace message indicating that the router passed the high temperature threshold and the high temperature threshold is active. The router generates this message when it first detects that the high temperature condition is active. If the ambient temperature of the router exceeds its stated operational maximum (set at the factory, independent of the configured high temperature threshold), it automatically shuts down until the ambient temperature returns to within its stated operational range. This prevents damage to the router and the data flow.

Cause: The value of the high temperature threshold is configured too low for the site's average operational ambient temperature.

Action: Verify that the high temperature threshold is set to the correct desired temperature, in consideration of the particular site's normal ambient temperature range.

Cause: Possible failure of the router's internal fan.

Action: Verify the operation of the router's internal fan. If the internal fan is not operational, contact customer service.

Cause: Possible failure of the environmental control system of the site where the router resides.

Action: Verify the operation of the site's environmental control system.

ENV.005

Level: UE-ERROR

Short Syntax: ENV.005 low temp thresh exceeded: *threshold_temperature_celsius*C (*threshold_temperature_fahrenheit*F)

Long Syntax: ENV.005 Low temperature threshold has been exceeded. Threshold: *threshold_temperature_celsius*C (*threshold_temperature_fahrenheit*F)

Description: Trace message indicating that the router passed the low temperature threshold and the low temperature threshold is active. The router generates this message when it first detects that the low

temperature condition is active.

Cause: The value of the low temperature threshold is configured too high for the site's average operational ambient temperature.

Action: Verify that the low temperature threshold has been set to the correct desired temperature, with consideration of the site's normal ambient temperature range.

Cause: Possible failure of the environmental control system of the site where the router resides.

Action: Verify the operation of the site's environmental control system.

Chapter 41. ESCON Network Interface (ESC)

This chapter describes ESCON Network Interface (ESC) messages. For information on message content and how to use the message, refer to the Introduction.

ESC.001

Level: ALWAYS

Short Syntax: ESC.001 bd frm LANtype *lan_type* LANnum *lan_num* on nt *network*

Long Syntax: ESC.001 frame received for unknown LAN type *lan_type*, LAN number *lan_num* on network *network*

Description: A frame was received from the channel destined for an unknown LAN type or LAN number.

ESC.002

Level: ALWAYS

Short Syntax: ESC.002 bd not *not_id* on nt *network*

Long Syntax: ESC.002 unknown notification *not_id* received from device driver on network *network*

Description: A notification was received from the device driver that was unknown.

ESC.003

Level: UE-ERROR

Short Syntax: ESC.003 bd 8232 cmd *cmd* on nt *network*

Long Syntax: ESC.003 unknown 8232 command *cmd* received on network *network*

Description: An 8232 command was received that was unknown.

ESC.004

Level: ALWAYS

Short Syntax: ESC.004 bd cmd *cmd* on nt *network*

Long Syntax: ESC.004 unknown IORB command *cmd* received on network *network*

Description: An IORB was received that contained an unknown command.

ESC.005

Level: ALWAYS

Short Syntax: ESC.005 no subch on nt *network*

Long Syntax: ESC.005 no subchannels are defined on network *network*, cannot pass self-test

Description: There are no subchannels defined for an ESCON base net so the network cannot be activated (pass self-test).

Cause: The virtual net handler(s) for this base net handler has (have) not been defined correctly.

Action: Define subchannels for the virtual net handler(s) on this ESCON adapter.

ESC.006

Level: UI-ERROR

Short Syntax: ESC.006 STOP: no IORB on nt *network*

Long Syntax: ESC.006 network *network* was unable to send a STOP command to the device driver because an IORB was not available

Description: The network was unable to complete deactivation because there was no IORB available with which to send the STOP command to the device driver.

ESC.007

Level: P-TRACE

Short Syntax: ESC.007 frm sent to lt *lantype* ln *lannumber* on nt *network*

Long Syntax: ESC.007 A frame was sent to LAN type *lantype*, LAN number *lannumber* on network *network*

Description: A frame was received on the channel and sent to a virtual net handler.

ESC.008

Level: P-TRACE

Short Syntax: ESC.008 data frm rcvd from nt *network*

Long Syntax: ESC.008 A data frame was received from network *network*

Description: A data frame was received from a virtual net handler to send to the channel.

ESC.009

Level: P-TRACE

Short Syntax: ESC.009 cmd *cmd_code* in frm rcvd from nt *network*

Long Syntax: ESC.009 command *cmd_code* in frame received from network *network*

Description: A command frame was received from a virtual net handler to send to the channel.

ESC.010

Level: P-TRACE

Short Syntax: ESC.010 notif *notif_code* rcvd on nt *network*

Long Syntax: ESC.010 notification *notif_code* received from device driver on network *network*

Description: A notification was received from the device driver.

ESC.011

Level: P-TRACE

Short Syntax: ESC.011 8232 cmd *cmd_code* rcvd on nt *network*

Long Syntax: ESC.011 8232 command *cmd_code* received on network *network*

Description: An 8232 command was received by the base net handler.

ESC.012

Level: C-TRACE

Short Syntax: ESC.012 nt *virtual_net_number* reg on nt *network*

Long Syntax: ESC.012 Network number *virtual_net_number* registering on base network *network*

Description: A virtual net handler is registering with an ESCON base net handler.

ESC.013

Level: P-TRACE

Short Syntax: ESC.013 Cmd *cmd_code* fail stat *cmd_status* on nt *network*

Long Syntax: ESC.013 Command *cmd_code* to device driver failed with status *cmd_status* on network *network*

Description: A command that the base net handler sent to the device driver has failed.

ESC.014

Level: P-TRACE

Short Syntax: ESC.014 Cmd *cmd_code* sent to DD on nt *network* (sub locaddr *locaddr* devaddr *devaddr* logpath *logpath*)

Long Syntax: ESC.014 Commands *cmd_code* was sent to the device driver on network *network* (subchannel local address *locaddr*, device address *devaddr*, logical path *logpath*)

Description: A command was sent to the device driver.

ESC.015

Level: P-TRACE

Short Syntax: ESC.015 Snd 8232 resp *cmd_code* (rc *retcode*) on nt *network* (sub locaddr *locaddr* devaddr *devaddr* logpath *logpath*)

Long Syntax: ESC.015 Sending 8232 response for command *cmd_code* with return code *retcode* on network *network* (subchannel local address *locaddr*, device address *devaddr*, logical path *logpath*)

Description: An 8232 response was sent to the host.

ESC.016

Level: P-TRACE

Short Syntax: ESC.016 Snd not *notification_id* to net *virt_net_number* on nt *network*

Long Syntax: ESC.016 Sending notification *notification_id* to net *virt_net_number* on network *network*

Description: A notification was sent to a virtual net handler from the base net handler.

ESC.017

Level: U-TRACE

Short Syntax: ESC.017 circdn for nt *net_num* on nt *network*

Long Syntax: ESC.017 circdown for net *net_num* called on network *network*

Description: The circuit down routine for a network has been called.

ESC.018

Level: U-TRACE

Short Syntax: ESC.018 circup for nt *net_num* on nt *network*

Long Syntax: ESC.018 circup for net *net_num* called on network *network*

Description: The circuit up routine for a network has been called.

ESC.019

Level: U-TRACE

Short Syntax: ESC.019 net up for nt *net_num* on nt *network*

Long Syntax: ESC.019 net up for net *net_num* called on network *network*

Description: The net up routine for a virtual network has been called.

ESC.020

Level: U-TRACE

Short Syntax: ESC.020 net dn for nt *net_num* on nt *network*

Long Syntax: ESC.020 net down for net *net_num* called on network *network*

Description: The net down routine for a virtual network has been called.

ESC.021

Level: UE_ERROR

Short Syntax: ESC.021 *file*(*line*): input *fsminput* curr stte *curr_state* new stte *new_state* actn *action* subchan *network*/)

Long Syntax: ESC.021 *file*(*line*): input *fsminput* current state *curr_state* new state *new_state* action *action* subchannel *network*/)

Description: Show the inputs to the FSM that is given in the message.

ESC.022

Level: UI-ERROR

Short Syntax: ESC.022 ESCON DD(slot *slot* adapter): Transmit List full threshold= *count*.

Long Syntax: ESC.022 ESCON DD(slot *slot* adapter): Transmit List full threshold= *count*.

Description: The device driver requires a transmit list entry in order to send data to the adapter. If the adapter cannot obtain an entry during initialization, the adapter will be restarted. If the adapter cannot obtain an entry at any other time, the internal software will attempt to recover. This error will only cause real problems for LSA.

Action: If the problem persists, contact Software Support.

ESC.028

Level: UI-ERROR

Short Syntax: ESC.028 ESCON DD asked to free private buffer 0x *iob* twice for adapter in slot *slot*.

Long Syntax: ESC.028 ESCON DD asked to free private buffer 0x *iob* twice for adapter in slot *slot*.

Description: The device driver has freed the same private buffer to the adapter twice. This could cause data corruption.

Action: If the problem persists, contact Software Support.

ESC.029

Level: UI_ERROR

Short Syntax: ESC.029 Unable to create dump file for ESCON adapter in slot *slot*. RC=0x *rc*

Long Syntax: ESC.029 Unable to create dump file for ESCON adapter in slot *slot*. RC=0x *rc*

Description: A fatal error was detected by the ESCON adapter microcode, but the device drive was unable to create a dump file. If this error did not occur as a result of removing the fiber, contact Software Support.

ESC.030

Level: UI_ERROR

Short Syntax: ESC.030 Dump file ' *filename*' created for the ESCON adapter in slot *slot*.

Long Syntax: ESC.030 Dump file ' *filename*' created for the ESCON adapter in slot *slot*.

Description: A fatal error was detected by the ESCON adapter microcode, and a dump file was created containing failure information. If this error did not occur as a result of removing the fiber, contact Software Support.

ESC.031

Level: C-INFO

Short Syntax: ESC.031 ESCON DD rcvd Logical Path Removed notif from slot *slot*,link addr=0x *link*, LPAR=0x *lpar*,cu-num=0x *cu_num*.

Long Syntax: ESC.031 ESCON DD received a Logical Path Removed notification from slot *slot* ESCON adapter, link addr=0x *link*, LPAR=0x *lpar*, cu-num=0x *cu_num*.

Description: The ESCON adapter has made removed a connection to the host via one of the configured subchannel paths.

ESC.032

Level: UE-ERROR

Short Syntax: ESC.032 Unknown ESCON adapter CCA error, nt *network*, subchan= *subchan*, cca0=0x *cca0*, cca1=0x *cca1*, rc=0x *rc*.

Long Syntax: ESC.032 Unknown ESCON adapter CCA error, network *network*, subchan= *subchan*, cca0=0x *cca0*, cca1=0x *cca1*, rc=0x *rc*.

Description: This is an unknown CCA error message being reported by the ESCON adapter.

Action: Typically, no action is required. If the problem

persists, contact Software Support. Refer to the documentation for further information.

ESC.033

Level: UE-ERROR

Short Syntax: ESC.033 ESCON adapter CCA error, nt *network*, *ccamsg*, subchan= *subchan*, cca0=0x *cca0*, cca1=0x *cca1*, rc=0x *rc*.

Long Syntax: ESC.033 ESCON adapter CCA error, network *network*, *ccamsg*, subchan= *subchan*, cca0=0x *cca0*, cca1=0x *cca1*, rc=0x *rc*.

Description: The ESCON adapter is reporting a CCA error to the ESCON device driver.

Action: Typically, no action is required. If the problem persists, contact Software Support. Refer to the documentation for further information.

ESC.034

Level: ALWAYS

Short Syntax: ESC.034 ESCON in slot *slot*. AIB FLASH mismatch: code at 0x *codelev*, adapter at 0x *adaplev*

Long Syntax: ESC.034 ESCON adapter in slot *slot*. AIB FLASH mismatch: code at 0x *codelev*, adapter at 0x *adaplev*

Description: The ESCON adapter has FLASH code that is different from the level available with the current load image.

Action: Contact Software Support to determine if the FLASH code on the adapter should be updated.

ESC.035

Level: C-INFO

Short Syntax: ESC.035 ESCON adapter in slot *slot* is operational.

Long Syntax: ESC.035 ESCON adapter in slot *slot* is operational.

Description: The ESCON adapter is operational. The adapter has not yet made a connection to the host.

ESC.036

Level: UI-ERROR

Short Syntax: ESC.036 ESCON adapter error, slot= *slot*, subchan= *subchan*, rc= *rc*, origcmd= *origcmd*, sev= *sev*, correl=0x *correl*.

Long Syntax: ESC.036 ESCON DD received an Error notif from slot *slot* ESCON adapter; subchan= *subchan*, rc = *rc*, origcmd= *origcmd*, severity= *sev*, correl=0x *correl*.

Description: The ESCON adapter is reporting an error to the ESCON device driver.

Action: Typically, no action is required. If the problem persists, contact Software Support. Refer to the documentation for further information.

ESC.037

Level: UI-ERROR

Short Syntax: ESC.037 ESCON adapter in slot= *slot* is offline to the host.

Long Syntax: ESC.037 ESCON adapter in slot= *slot* is offline to the host.

Description: The ESCON adapter is reporting that it is offline to the host. The adapter will be automatically restarted. Diagnostics will be invoked if the adapter fails to restart.

Action: If the adapter passes diagnostics but fails to start, contact Software Support.

ESC.038

Level: UI-ERROR

Short Syntax: ESC.038 ESCON DD received i960 Processor Fault notif from slot= *slot* ESCON adapter, Fault Type=0x *ft*.

Long Syntax: ESC.038 ESCON DD received an i960 Processor Fault notif from slot *slot* ESCON adapter with Fault Type=0x *ft*.

Description: The ESCON adapter is reporting that it had an i960 processor fault. The adapter will be automatically restarted. Diagnostics will be invoked if the adapter fails to start.

Action: Contact Software Support.

ESC.039

Level: UI-ERROR

Short Syntax: ESC.039 ESCON DD received SLC2 NMI Detected notif from slot= *slot* ESCON adapter.

Long Syntax: ESC.039 ESCON DD received an SLC2 NMI Detected notif from slot *slot* ESCON adapter.

Description: The ESCON adapter is reporting that it detected an SLC2 NMI error. The adapter will be automatically restarted. Diagnostics will be invoked if the adapter fails to restart.

Cause: This may be the result of disconnecting the ESCON fiber from the ESCON adapter and then reconnecting it.

Action: Contact Software Support.

ESC.040

Level: U-INFO

Short Syntax: ESC.040 ESCON adapter in slot *slot* had an unexpected interrupt.

Long Syntax: ESC.040 ESCON DD received an Unexpected Interrupt notification from slot *slot* ESCON adapter.

Description: ESCON adapter had an unexpected interrupt. If the problem persists, contact Software Support.

ESC.041

Level: UI-ERROR

Short Syntax: ESC.041 ESCON adapter in slot *slot* had a serial engine failure, dump is *log_stat*.

Long Syntax: ESC.041 ESCON DD received a Serial Engine Failure notification from slot *slot* ESCON adapter, dump is *log_stat*.

Description: The ESCON adapter is reporting that it had a serial engine failure. The adapter will be automatically restarted. Diagnostics will be invoked if the adapter fails to restart.

Cause: This may be the result of disconnecting the ESCON fiber from the ESCON adapter and then reconnecting it.

Action: If the adapter fails to restart, contact Software Support.

ESC.042

Level: UI-ERROR

Short Syntax: ESC.042 Slot *slot* ESCON adapter microcode aborted with rc=0x *rc*.

Long Syntax: ESC.042 ESCON DD received a Microcode Aborted notification from slot *slot* ESCON adapter, rc=0x *rc*.

Description: The ESCON adapter is reporting that the microcode aborted. The adapter will be automatically restarted. Diagnostics will be invoked if the adapter fails to restart.

Cause: This may be the result of disconnecting the ESCON fiber from the ESCON adapter and then reconnecting it.

Action: If the adapter fails to restart, contact Software Support.

ESC.043

Level: C-INFO

Short Syntax: ESC.043 ESCON DD rcvd Logical Path Estbl notif from slot *slot*,link addr=0x *link*, LPAR=0x *lpar*,cu-num=0x *cu_num*.

Long Syntax: ESC.043 ESCON DD received a Logical Path Established notification from slot *slot* ESCON adapter, link addr=0x *link*, LPAR=0x *lpar*, cu-num=0x *cu_num*.

Description: The ESCON adapter has made a connection to the host via one of the configured subchannel paths.

ESC.044

Level: UI-ERROR

Short Syntax: ESC.044 ESCON adapter in slot *slot* had a POST error, error = 0x *error*.

Long Syntax: ESC.044 ESCON adapter in slot *slot* has a POST error, error = 0x *error*.

Description: The ESCON adapter had a POST error. The adapter will be automatically restarted. Diagnostics will be invoked if the adapter fails to restart.

Action: If adapter fails to restart, contact Software Support.

ESC.045

Level: UI-ERROR

Short Syntax: ESC.045 ESCON adapter in slot *slot* had a POST error, CBSP value=0x *error*.

Long Syntax: ESC.045 ESCON adapter in slot *slot* had a POST error, CBSP value=0x *error*.

Description: The ESCON adapter had a POST error. The adapter will be automatically restarted. Diagnostics will be invoked if the adapter fails to restart.

Action: If adapter fails to restart, contact Software Support.

ESC.046

Level: UI-ERROR

Short Syntax: ESC.046 ESCON adapter in slot *slot* did not complete POST.

Long Syntax: ESC.046 ESCON adapter in slot *slot* did not complete POST.

Description: The ESCON adapter did not complete POST. The adapter will be automatically restarted. Diagnostics will be invoked if the adapter fails to restart.

Action: If the adapter passes diagnostics but fails to restart, contact Software Support.

ESC.047

Level: UI-ERROR

Short Syntax: ESC.047 ESCON adapter in slot *slot* had a PrePOST error = 0x *error*.

Long Syntax: ESC.047 ESCON adapter in slot *slot* had a PrePOST error = 0x *error*.

Description: The ESCON adapter had a PrePOST error. The adapter will be automatically restarted. Diagnostics will be invoked if the adapter fails to restart.

Action: If the adapter does not restart, contact Software Support.

ESC.048

Level: UI-ERROR

Short Syntax: ESC.048 Slot *slot* does not contain an ESCON card, identifier = *id*.

Long Syntax: ESC.048 Slot *slot* does not contain an ESCON card, identifier = *id*.

Description: The slot does not contain an ESCON card and the software has been configured for an ESCON adapter in that slot.

Action: Correct the configuration. If the problem occurs after reconfiguration, contact Software Support.

ESC.049

Level: UI-ERROR

Short Syntax: ESC.049 Slot *slot* ESCON Adapter timed-out during initialization, cmd=0x *cmd*.

Long Syntax: ESC.049 Slot *slot* ESCON Adapter timed-out during initialization, cmd=0x *cmd*.

Description: The adapter will be automatically restarted.

Action: If the adapter does not restart, contact Software Support.

ESC.050

Level: UI-ERROR

Short Syntax: ESC.050 Slot *slot* ESCON Control Unit table did not load correctly, rc=0x *rc*, tbl=0x *tbl_num*.

Long Syntax: ESC.050 Slot *slot* ESCON Control Unit table did not load correctly, rc=0x *rc*, tbl=0x *tbl_num*.

Description: The ESCON adapter cannot start properly without these tables. The adapter will be automatically restarted.

Action: If the adapter does not restart, contact Software Support.

ESC.051

Level: UI-ERROR

Short Syntax: ESC.051 ESCON DD could not obtain a Control Buffer from slot *slot* adapter.

Long Syntax: ESC.051 ESCON DD could not obtain a Control Buffer from adapter in slot *slot*.

Description: The device driver requires a buffer from the adapter. If the adapter cannot provide the buffer then the adapter is not functioning properly. The adapter will be restarted automatically.

Action: If the problem persists, contact Software Support.

ESC.052

Level: UI_ERROR

Short Syntax: ESC.052 *file*(*line*): ESCON DD encountered an internal error for slot *slot*. Identifier = *id*.

Long Syntax: ESC.052 *file*(*line*): ESCON DD encountered an internal error for slot *slot*. Identifier = *id*.

Description: The ESCON device driver has encountered a condition that it cannot handle properly.

Action: If the problem persists, contact Software Support.

ESC.053

Level: UI-ERROR

Short Syntax: ESC.053 ESCON DD detected a CRC error in CU Table *tbl_num* for slot *slot*.

Long Syntax: ESC.053 ESCON DD detected a CRC error in CU Table *tbl_num* for slot *slot*.

Description: The adapter will be restarted automatically.

Action: If the problem persists, contact Software Support.

ESC.054

Level: UI-ERROR

Short Syntax: ESC.054 ESCON DD could not obtain system memory; slot=0x *slot*, identifier= *id*.

Long Syntax: ESC.054 ESCON DD could not obtain system memory; slot=0x *slot*, identifier= *id*.

Description: If this error occurred during initialization, the adapter will be restarted.

Action: If the problem persists, contact Software Support.

ESC.055

Level: UI-ERROR

Short Syntax: ESC.055 ESCON DD could not open dump files on harddrive. Dumps not available for slot *slot* adapter.

Long Syntax: ESC.055 ESCON DD could not open the dump files on the harddrive. The dumps are not available for slot *slot* adapter

Description: The device driver attempted to open a file on the harddrive but was unsuccessful. The dump of the ESCON adapter is not available.

Action: If problems with the adapter persist, contact Software Support.

ESC.056

Level: UI-ERROR

Short Syntax: ESC.056 ESCON DD could not dump all slot *slot* ESCON adapter *data_type* data to the dump file.

Long Syntax: ESC.056 ESCON DD could not dump all of the slot *slot* ESCON adapter *data_type* data to the dump file on the harddrive.

Description: The device driver attempted to dump the ESCON adapter data to a file on the harddrive. The IRAM dump may be partially available in c:\ESCONIx.DMP, where x is the slot number. The DRAM dump may be partially available in c:\ESCONDx.DMP, where x is the slot number.

Action: Contact Software Support.

ESC.057

Level: C-INFO

Short Syntax: ESC.057 ESCON DD received a reset subchannel notif for subchannel 0x *sc*, slot= *slot*.

Long Syntax: ESC.057 ESCON DD received a reset subchannel notification for subchannel 0x *sc*, slot= *slot*.

Description: The device driver received a reset subchannel notification.

ESC.058

Level: C-INFO

Short Syntax: ESC.058 Incorrect subchannel configuration detected for slot *slot* ESCON adapter.

Long Syntax: ESC.058 Incorrect subchannel configuration detected for slot *slot* ESCON adapter.

Description: The device driver has detected that a subchannel configuration is incorrect. Correctly configured subchannels should not be affected by this problem.

Action: Correct the configuration.

ESC.059

Level: UI-ERROR

Short Syntax: ESC.059 ESCON DD(slot *slot* adapter): Command FIFO full threshold= *count*.

Long Syntax: ESC.059 ESCON DD(slot *slot* adapter): Command FIFO full threshold= *count*.

Description: The device driver requires a Command FIFO entry in order to communicate with the adapter. If the adapter cannot obtain an entry during initialization, the adapter will be restarted. If the adapter cannot obtain an entry at any other time, the internal software will attempt to recover. This error will only cause real problems for LSA.

Action: If the problem persists, contact Software Support.

ESC.060

Level: P-TRACE

Short Syntax: ESC.060 ESCON DD sending frame from slot= *slot*,, subchan= *subchan*,, LT= *lantype*,, LN= *lannum*, to base net.

Long Syntax: ESC.060 ESCON DD rcvd frame from slot *slot,* ESCON, subchan= *subchan*,, LanType= *lantype*,, and LanNum= *lannum*; sending it to base net.

Description: A frame was received by the channel and was sent to the ESCON base net handler.

ESC.061

Level: P-TRACE

Short Syntax: ESC.061 ESCON DD rcvd frame from net handler for slot= *slot*,, subchan= *subchan*,, LT= *lantype*,, LN= *lannum*,,PDU-hdr= *pdu_len*

Long Syntax: ESC.061 ESCON DD received a frame from a net handler destined for slot *slot,* ESCON adapter, subchan= *subchan*,, LanType= *lantype*,, and LanNum= *lannum*,, PDU-header len= *pdu_len*.

Description: An ESCON-related nethandler sent the ESCON DD a frame to transmit.

ESC.062

Level: P-TRACE

Short Syntax: ESC.062 ESCON DD rcvd *cmd,* cmd from net handler for slot *slot* ESCON.

Long Syntax: ESC.062 ESCON DD received *cmd,* command from net handler for slot *slot* ESCON adapter.

Description: An ESCON-related net handler sent the ESCON DD a command.

ESC.063

Level: P-TRACE

Short Syntax: ESC.063 ESCON DD rcvd *cmd,* cmd from nethandler for slot *slot,* ESCON, subchan= *subchan*.

Long Syntax: ESC.063 ESCON DD received *cmd,* command from a nethandler for slot *slot,* ESCON adapter, subchan= *subchan*.

Description: An ESCON-related net handler sent the ESCON DD a command.

ESC.064

Level: P-TRACE

Short Syntax: ESC.064 ESCON DD sent *notif,* notif for slot *slot,* ESCON, subchan= *subchan,*, LT= *lantype,*, LN= *lannum*, to nethandler.

Long Syntax: ESC.064 ESCON DD sent *notif,* notif for slot *slot,* ESCON adapter, subchan= *subchan,*, LT= *lantype,*, LN= *lannum*, to nethandler.

Description: The ESCON device driver sent a notification to an ESCON-related net handler

---

ESC.065

Level: U-INFO

Short Syntax: ESC.065 ESCON adapter ran out of rcv buffers, LCS frame discarded, slot= *slot*, local sc= *subchan*

Long Syntax: ESC.065 ESCON adapter ran out of receive buffers and discarded an LCS frame; slot= *slot* local subchan= *subchan*.

Description: The ESCON adapter is reporting that it discarded an LCS frame because it could not obtain a receive buffer.

Action: Typically, no action is required. If the problem persists, increase the number of receive buffers for this ESCON adapter.

ESC.066

Level: UI-ERROR

Short Syntax: ESC.066 ESCON adapter ran out of rcv buffers, LSA frame discarded, slot= *slot*, local sc= *subchan*

Long Syntax: ESC.066 ESCON adapter ran out of receive buffers and discarded an LSA frame; slot= *slot* local subchan= *subchan*.

Description: The ESCON adapter is reporting that it discarded an LSA frame because it could not obtain a receive buffer.

Action: Increase the number of receive buffers for this ESCON adapter.

ESC.067

Level: U-INFO

Short Syntax: ESC.067 ESCON adapter ran out of rcv buffers, MPC+ frame discarded, slot= *slot*, local sc= *subchan*

Long Syntax: ESC.067 ESCON adapter ran out of receive buffers and discarded an MPC+ frame; slot= *slot* local subchan= *subchan*.

Description: The ESCON adapter is reporting that it discarded an MPC+ frame because it could not obtain a receive buffer.

Action: Typically, no action is required. If the problem persists, increase the number of receive buffers for this ESCON adapter.

Panic escnomem

Short Syntax: escnomem: ESCON handler no memory

Description: An ESCON handler cannot allocate memory for control block(s).

Action: Contact customer service.

Panic escnsram

Short Syntax: escnsram: ESCON SRAM not found

Description: The SRAM record for an ESCON handler could not be found.

Action: Contact customer service.

Panic escbprt

Short Syntax: escbprt: bad prot init

Description: An unsupported Network Layer protocol tried to initialize an ESCON handler.

Action: Contact customer service.

Panic escdreg

Short Syntax: escdreg: virt net already reg

Description: An ESCON virtual net handler has already registered with the base.

Action: Contact customer service.

Panic escbreq

Short Syntax: escbreq: bad xmit rqst

Description: An unsupported protocol packet was given to the ESCON handler for transmission.

Action: Contact customer service.

Panic escnosub

Short Syntax: escnosub: subch not found

Description: The requested logical path and device address was not found in the ESCON base handler subchannel table.

Action: Contact customer service.

Panic escbcall

Short Syntax: escbcall: bad call to routine.

Description: An invalid call was made to a routine.

Action: Contact customer service.

Panic escbprd

Short Syntax: escbprt: bad prot down

Description: An unsupported Network Layer protocol tried to uninitialize an ESCON handler.

Action: Contact customer service.

Chapter 42. Ethernet Network Interface (ETH)

This chapter describes Ethernet Network Interface (ETH) messages. For information on message content and how to use the message, refer to the Introduction.

ETH.001

Level: P-TRACE

Short Syntax: ETH.001 brd rcv unkwn typ *packet_type source_Ethernet_address* -> *destination_Ethernet_address* nt *network*

Long Syntax: ETH.001 broadcast packet received with unknown Ethernet type *packet_type* from host *source_Ethernet_address* to *destination_Ethernet_address* network *network*

Description: A broadcast packet was received with an unknown or unsupported Ethernet type field.

ETH.002

Level: UE-ERROR

Short Syntax: ETH.002 rcv unkwn typ *packet_type source_Ethernet_address* -> *destination_Ethernet_address* nt *network*

Long Syntax: ETH.002 packet received with unknown Ethernet type field *packet_type* from *source_Ethernet_address* to *destination_Ethernet_address* network *network*

Description: A non-broadcast packet was received with an unknown or unsupported Ethernet type field.

ETH.003

Level: P-TRACE

Short Syntax: ETH.003 brd 802.3 bd ln *actual_length claimed_length source_Ethernet_address* -> *destination_Ethernet_address* nt *network*

Long Syntax: ETH.003 broadcast packet received with a bad 802.3 length field actual *actual_length* claimed *claimed_length* from *source_Ethernet_address* to *destination_Ethernet_address* network *network*

Description: A broadcast packet was received with a type field that indicated 802.3 but was shorter than data length claimed in the 802.3 header.

ETH.004

Level: UE-ERROR

Short Syntax: ETH.004 802.3 bd ln *actual_length claimed_length source_Ethernet_address* -> *destination_Ethernet_address* nt *network*

Long Syntax: ETH.004 packet received with a bad 802.3 length field actual *actual_length* claimed *claimed_length* from *source_Ethernet_address* to *destination_Ethernet_address* network *network*

Description: A non-broadcast packet was received with a type field that indicated 802.3 but was shorter than data length claimed in the 802.3 header.

ETH.005

Level: UE-ERROR

Short Syntax: ETH.005 DN bd ln *actual_length claimed_length source_Ethernet_address* -> *destination_Ethernet_address* nt *network*

Long Syntax: ETH.005 DECnet packet received with a bad length actual *actual_length* claimed *claimed_length* from *source_Ethernet_address* to *destination_Ethernet_address* network *network*

Description: A DECnet packet was received with a length field that was larger than the actual length of the packet.

ETH.006

Level: ALWAYS

Short Syntax: ETH.006 Eth *adapterror_condition adapdiag_code* nt *network*

Long Syntax: ETH.006 Ethernet adapter: *adapterror_condition*, diag *adapdiag_code* network *network*

Description: An error was encountered on the Fast Ethernet port. The causes are "CAUTION: DUPLEX MAY BE DIFFERENT ON SWITCH PORT", and "Fatal Error. Disable and run internal diagnostics for adapter".

Cause: The message "CAUTION: DUPLEX MAY BE DIFFERENT ON SWITCH PORT" is displayed to warn the user of a possible mis-match in duplex mode. It will always be displayed when connected to a non-negotiating device such as a hub. A mis-match in duplex mode can cause severe performance degradation.

Action: When the interface is conncected to a switch, ensure that the configuration for the switch port is identical to the configuration for the interface. Refer to the chapter on "Configuring and Monitoring the 10/100 Mbps Ethernet Network Interface" for more details.

Cause: The error "Fatal Error. Disable and run internal

diagnostics for adapter″ indicates that the software encountered an unexpected error when communicating with the adapter.

Action: This indicates a possible hardware problem on the adapter. Capture and save the information logged in Talk 2. Disable the interface and run diagnostics.

ETH.007

Level: UE-ERROR

Short Syntax: ETH.007 DIX V2 bd ln *actual_length claimed_length source_Ethernet_address -> destination_Ethernet_address* nt *network*

Long Syntax: ETH.007 packet received with a bad length field actual *actual_length* claimed *claimed_length* from *source_Ethernet_address* to *destination_Ethernet_address* network *network*

Description: A DIX V2 packet was received with a type field of 0x80D5 (IBM SNA or Netbios) which had a length field that was larger than the actual length of the packet.

ETH.010

Level: C-INFO

Short Syntax: ETH.010 LLC unk SAP *DSAP source_Ethernet_address -> destination_Ethernet_address* nt *network*

Long Syntax: ETH.010 802.2 LLC packet received with unknown DSAP *DSAP* from host *source_Ethernet_address* to *destination_Ethernet_address* network *network*

Description: An 802.2 LLC packet was received from the network with an inactive (unrecognized) DSAP.

ETH.011

Level: C-INFO

Short Syntax: ETH.011 LLC nt typ 1 *LLC_control_type* nt *network*

Long Syntax: ETH.011 802.2 LLC packet received, not Type 1 *LLC_control_type* network *network*

Description: A packet was received from the network that had an LLC but was not a Type 1 LLC.

ETH.012

Level: C-INFO

Short Syntax: ETH.012 LLC RSP *LLC_SSAP* nt *network*

Long Syntax: ETH.012 LLC RESPONSE packet received *LLC_SSAP* network *network*

Description: An LLC response was received from the network.

ETH.013

Level: C-INFO

Short Syntax: ETH.013 LLC XID *LLC_SSAP* nt *network*

Long Syntax: ETH.013 LLC XID packet received *LLC_SSAP* network *network*

Description: An LLC XID packet was received from the network.

ETH.014

Level: C-INFO

Short Syntax: ETH.014 LLC TEST *LLC_SSAP* nt *network*

Long Syntax: ETH.014 LLC TEST packet received *LLC_SSAP* network *network*

Description: An LLC TEST packet was received from the network.

ETH.015

Level: U-INFO

Short Syntax: ETH.015 unrec ctl *LLC_control_field* nt *network*

Long Syntax: ETH.015 packet received with unrecognized control field *LLC_control_field* network *network*

Description: A packet was received from the network that had an illegal control field or UI.

ETH.017

Level: P-TRACE

Short Syntax: ETH.017 LOOP rcv *source_Ethernet_address -> destination_Ethernet_address*, nt *network*

Long Syntax: ETH.017 Loopback Protocol frame received from *source_Ethernet_address* to *destination_Ethernet_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet was received.

ETH.018

Level: UE-ERROR

Short Syntax: ETH.018 LOOP odd skp *count*, *source_Ethernet_address -> destination_Ethernet_address*, nt *network*

Long Syntax: ETH.018 Loopback Protocol, odd skipCount *count* from *source_Ethernet_address* to *destination_Ethernet_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet had an odd

skipCount in the packet. It will be discarded.

Cause: Programming error on remote node.

ETH.019

Level: UE-ERROR

Short Syntax: ETH.019 LOOP func *function* not forw, *source_Ethernet_address* -> *destination_Ethernet_address*, nt *network*

Long Syntax: ETH.019 Loopback Protocol, function *function* not Forward Data from *source_Ethernet_address* to *destination_Ethernet_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet did not have a function code of forward (2). It will be discarded.

Cause: Function code was reply (1), because we were the ultimate destination of this packet.

Action: None.

Cause: Undefined function code, due to programming error in remote node.

ETH.020

Level: UE-ERROR

Short Syntax: ETH.020 LOOP mc fwd dst *forward_Ethernet_address*, *source_Ethernet_address* -> *destination_Ethernet_address*, nt *network*

Long Syntax: ETH.020 Loopback Protocol, multicast forward address *forward_Ethernet_address* from *source_Ethernet_address* to *destination_Ethernet_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet has a forward address that is a multicast. It will be discarded.

Cause: Programming error in remote node.

ETH.021

Level: P-TRACE

Short Syntax: ETH.021 LOOP fwd *source_Ethernet_address* -> *forward_Ethernet_address*, nt *network*

Long Syntax: ETH.021 Loopback Protocol, forwarding from *source_Ethernet_address* to *forward_Ethernet_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet is being forwarded to the specified next hop.

ETH.022

Level: UI-ERROR

Short Syntax: ETH.022 LOOP fwd to *forward_Ethernet_address* dsc, rsn *code*, nt *network*

Long Syntax: ETH.022 Loopback protocol, forward to *forward_Ethernet_address* discarded, for reason *code*, network *network*

Description: A Ethernet Loopback Protocol (Configuration Testing Protocol) packet could not be forwarded to the specified address, for the reason specified by code.

ETH.023

Level: UI-ERROR

Short Syntax: ETH.023 LLC RSP to *destination_Ethernet_address* dsc, rsn *code*, nt *network*

Long Syntax: ETH.023 LLC response to *destination_Ethernet_address* discarded, for reason *code*, network *network*

Description: An LLC response (XID or TEST) could not be transmitted to the specified address, for the reason specified by code.

ETH.024

Level: UE-ERROR

Short Syntax: ETH.024 MOP bd ln *actual_length claimed_length source_Ethernet_address* -> *destination_Ethernet_address* nt *network*

Long Syntax: ETH.024 DECnet MOP packet received with a bad length actual *actual_length* claimed *claimed_length* from *source_Ethernet_address* to *destination_Ethernet_address* network *network*

Description: A DECnet MOP packet was received with a length field that was larger than the actual length of the packet.

ETH.025

Level: UE-ERROR

Short Syntax: ETH.025 LOOP bd skp *count*, *source_Ethernet_address* -> *destination_Ethernet_address*, nt *network*

Long Syntax: ETH.025 Loopback Protocol, bad skipCount *count* from *source_Ethernet_address* to *destination_Ethernet_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet had a skipCount in the packet that points to beyond the end of the packet. It will be discarded.

Cause: Programming error on remote node.

ETH.042

Level: UI-ERROR

Short Syntax: ETH.042 Unable to get buf for ethernet packet.

Long Syntax: ETH.042 Unable to get buffer for ethernet packet.

Description: A buffer to set an Ethernet address, or to copy an Ethernet packet couldn't be gotten because of a buffer shortage.

ETH.043

Level: ALWAYS

Short Syntax: ETH.043 CMD596 Command Timeout. Interface *network* being restarted.

Long Syntax: ETH.043 CMD596 Command Timeout. Interface *network* being restarted.

Description: The 82596 chip on the interface card has failed to clear the command field for this interface. The interface will be re-initialized.

ETH.044

Level: ALWAYS

Short Syntax: ETH.044 I5IOCTL Bad Command *network* being restarted.

Long Syntax: ETH.044 I5IOCTL Bad Command *network* being restarted.

Description: An Incorrect command field has been sent to the driver. The interface will be re-initialized.

ETH.045

Level: UI-ERROR

Short Syntax: ETH.045 Eth self-test *selftest_phase* fld *error_condition* nt *network*

Long Syntax: ETH.045 Ethernet self-test phase *selftest_phase* failed: *error_condition*, network *network*

Description: The selftest for the Ethernet card has reported an error during selftest. The phases are ″Reset board″, ″Reset delay″, ″Check reset done″, ″Check reset delay″, ″Init SCB″, ″Init SCB delay″, ″Init SCB completion″, ″Read hardware address″, ″Set bus throttle timers″, ″Internal loopback″, ″Set hardware address″, ″Enable receive″, ″Internal loopback (output)″, ″Check internal loopback data″, ″External loopback″, ″External loopback delay″, ″External loopback (output)″, ″Check external loopback data″, ″Network loopback″, ″Network loopback delay″, ″Network loopback (output)″, ″Check network loopback data″, ″Clear loopback″, and ″Operational test″.

Cause: In the ″Reset board″ phase, the error ″Packetsize of < 1500 bytes″ indicates that the interface has been provided with buffers that are too small.

Action: Correct configuration of system that is artificially reducing packet size below Ethernet requirement of 1500 bytes.

Cause: In all phases, the error ″No buffers″ indicates that there is a severe packet buffer shortage in the router.

Action: Increase buffer memory size, decrease buffer size on configurable networks.

Cause: In phase ″Init SCB completion″, the error ″ISCP busy not 0″ indicates that the BUSY byte of the 82596 Intermediate System Configuration Pointer (ISCP) did not clear after the CA signal was sent.

Action: This indicates a probable hardware problem with the interface or router. Run diagnostics.

Cause: The error ″Unexpected receive pkt″ indicates that the interface recieved a packet in a self-test state where it did not expect to receive a packet.

Action: This indicates a possible hardware problem with the interface. Run diagnostics.

Cause: The error ″Loop back count error″, indicates that the recieved loopback packet was not of the same length as the transmitted one.

Action: This indicates a possible hardware problem with the interface. Run diagnostics.

Cause: The error ″Loop back stat error″ indicates that the receive of the loopback packet had an unsuccessful error status.

Action: This indicates a possible hardware problem with the interface. Run diagnostics.

Cause: The error ″Loop back data error″ indicates that there was a data mismatch in the loopback packet.

Action: This indicates a possible hardware problem with the interface. Run diagnostics.

Cause: In the ″Operational test″ phase, the error ″maintenance failure″ indicates that the interface could not perform a sucessful maintenance test. (The maintenance test sends one packet and checks for carrier sense.)

Action: Check the transceiver cabling and hardware.

Cause: In all phases, the error ″timeout″ indicates that the entire self-test did not complete within one-eighth of a second.

ETH.046

Level: UE-ERROR

Short Syntax: ETH.046 IPX pkt in *received_encapsulation* encap ign, using *configured_encapsulation* encaps, nt *network*

Long Syntax: ETH.046 IPX pkt in encapsulation

received_encapsulation ignored, using encapsulation *configured_encapsulation* on network *network*

Description: This message is generated when an IPX packet is received in a data-link encapsulation (frame) other than the one configured for IPX on this interface. The packet will be ignored. The received_encapsulation and configured_encapsulation are one of ″ETHERNET_802.3″, ″ETHERNET_II″, ″ETHERNET_802.2″, or ″ETHERNET_SNAP″. ETHERNET_802.3 is also known as ″Novell″, and ETHERNET_II is also known as ″Ethernet″.

Cause: If only one encapsulation is being used on this network, this node's encapsulation is not the same as all other IPX nodes on the network.

Action: Configure all nodes on network to use same encapsulation.

Cause: If multiple encapsulations are being used on this network, a packet has been received from a node using an encapsulation different from this node.

ETH.047

Level: UI-ERROR

Short Syntax: ETH.047 Eth self-test *selftest_phase* fld *error_condition* nt *network*

Long Syntax: ETH.047 Ethernet port self-test phase *selftest_phase* failed: *error_condition*, network *network*

Description: The self-test for the SCC Ethernet port has reported an error during self-test. The phases are ″Reset port″, ″Set media selection″, ″Set hardware address″, ″Network loopback″, ″Enable receive″, ″Network loopback (output)″, ″Check network loopback data″, ″Clear loopback″, ″Set multicast addresses″, and ″Operational test″.

Cause: In the ″Reset port″ phase, the error ″Packetsize of < 1500 bytes″ indicates that the interface has been provided with buffers that are too small.

Action: Correct the configuration of the system that is artificially reducing packet size below the Ethernet requirement of 1500 bytes.

Cause: In all phases, the error ″No buffers″ indicates that there is a severe packet buffer shortage in the router.

Action: Increase memory size, decrease size of routing tables, decrease buffer allocations to networks, decrease buffer size on configurable networks.

Cause: The error ″Loop back data error″ indicates that there was a data mismatch in the loopback packet.

Action: This indicates a possible hardware problem with the interface. Run diagnostics.

Cause: The error ″Loop back count error″ indicates that the received loopback packet was not of the same length as the transmitted one.

Action: This indicates a possible hardware problem with the interface. Run diagnostics.

Cause: The error ″Loop back status error″ indicates that the receive of the loopback packet had an unsuccessful error status.

Action: This indicates a possible hardware problem with the interface. Run diagnostics.

Cause: In all phases, the error ″Timeout″ indicates that the entire self-test did not complete within one-eighth of a second.

Action: This indicates a possible hardware problem with the interface. Run diagnostics.

Cause: In the ″Operational test″ phase, the error ″maintenance failure″ indicates that the interface could not perform a sucessful maintenance test. (The maintenance test sends one packet and checks for carrier sense.)

Action: Check the transceiver cabling and hardware. The router is probably not connected to the Ethernet correctly, or there is a hardware failure.

ETH.048

Level: UI-ERROR

Short Syntax: ETH.048 Eth Error *adapterror_condition adapdiag_code* nt *network*

Long Syntax: ETH.048 Ethernet adapter error: *adapterror_condition*, diag *adapdiag_code* network *network*

Description: An error was encountered on the Fast Ethernet port. The causes are ″Error status from TB″, ″Invalid counter from TB″, ″Timeout waiting for valid link status″, ″Timeout waiting for auto negotiation″, ″Link partner does not support auto negotiation″, ″Address parity error detected on bus″, ″Unable to set multicast address. State = ″, ″Burnt-in UAA is used due to illegitimate LAA″, ″Lost connection to link partner. phy reg01 = ″, and ″Incorrect ID read from PHY chip on adapter. ID = ″.

Cause: ″Error status from TB″ indicates that the transparent bridging logic on the interface signalled an error condition during initialization.

Action: Issue the test command for the interface. If the message re-appears, then power the system off and on. If the message re-appears, run diagnostics on the interface. If the diagnostics indicate a similar problem, the interface card may need to be replaced.

Cause: ″Invalid counter from TB″ indicates the 'frames filtered' counter test failed during initialization of the transparent bridge logic.

Action: Issue the test command for the interface. If the message re-appears, then power the system off and on. If the message re-appears, run diagnostics on the interface. If the diagnostics indicate a similar problem, the interface card may need to be replaced.

Cause: ″Timeout waiting for valid link status″ indicates that the interface is not receiving a valid link signal from its link partner. Interface will no longer wait for the link signal. Instead it will reset the interface and resume listening for a valid link signal. This condition can occur due to the following causes: 1) Bad or incorrect length of cable from interface to its link partner. 2) Disabled or malfunctioning port on the link partner. 3) The connector(s) on the cable is bad or not properly inserted into the port(s). 4) Speed or duplex mode configured for interface is not supported by the link partner.

Action: For 1), verify that there is no discontinuity in the cable. If connected thhrough patch panels, try a direct Cat. 5 cable no more than 100 meters in length. For 2), try a different port on the link partner. Ensure that it has not been disabled. For 3), Ensure both ends of the cable are inserted all the way into the port at both ends. Also ensure that it is not a cross-wired cable. For 4), Check the capabilities of the link partner and configure the interface accordingly. Refer to the chapter on ″Configuring and Monitoring the 10/100 Mbps Ethernet Network Interface″ for more details.

Cause: ″Timeout waiting for auto negotiation″ indicates that the interface is getting a valid link signal from the link partner, however the auto-negotiaion function is failing.

Action: Check the capabilities of the link partner and configure the interface accordingly. If auto-negotiation is still unsucessful, ensure that the cable length is not greater than 100 meters. If connected via patch panels, try using a direct cable, less than 100 meters in length. Refer to the chapter on ″Configuring and Monitoring the 10/100 Mbps Ethernet Network Interface″ for more details.

Cause: ″Link partner does not support auto negotiation″ indicates the interface has detected that the link partner is not capable of performing auto-negotiation.

Action: Check the capabilities of the link partner and configure the interface accordingly.

Cause: ″Address parity error detected on bus″ indicates that the interface has detected a parity error.

Action: If this message occurs more than once, issue the test command for the interface. If the message re-appears, then power the system off and on. If the message re-appears, run diagnostics on the interface. If the diagnostics indicate a similar problem, the interface card may need to be replaced.

Cause: ″Unable to set multicast address″ indicates that the interface was unable to set the filter which will permit receipt of frames destined to a certain multicast address. This can happen when the interface is unable to suspend the hardware to perform the operation.

Action: Issue the ″Test″ command for this interface, or disable and re-enable the interface. This will cause the hardware for this interface to be reset and allow storage of the multicast filter masks.

Cause: ″Burnt-in UAA is used due to illegitimate LAA″ indicates that the user specified locally administered MAC address is invalid. The universal address assigned to the interface will be used instead.

Action: If the locally adminstered address must be used on this interface, change the configuration providing a legal locally adminstered MAC address and restart the system.

Cause: ″Lost connection to link partner″ indicates that the interface has detected loss of a valid link signal from its link partner. This can occur under the following conditions: 1) The cable connector is removed or not properly inserted at either ends 2) The cable has been damaged. 3) The link partner is not sending valid link signals. (Note this can be a temporary condition from a link partner).

Action: For 1) Ensure that both the connectors are properly inserted. 2) Ensure that the cable has not been harmed. 3) Ensure that the link partner is functioning normally.

Cause: ″Incorrect ID read from PHY chip on adapter″ indicates that the PHY chip on the adapter for for the interface may be bad.

Action: Disable the interface and run diagnostics on the port associated with the interface for which this message was displayed. The adapter may need to be replaced.

ETH.049

Level: C-INFO

Short Syntax: ETH.049 Eth Info. *adaptinfo_condition adapinfo_data* nt *network*

Long Syntax: ETH.049 Ethernet adapter info: *adaptinfo_condition*, Data: *adapinfo_data* network *network*

Description: Information notification for the Fast Ethernet interface. The reasons are ″Performing Unicast frame filtering in software. State = ″, ″TB has been enabled. State = ″, ″Auto Negotiation is complete. RC = ″, ″Interface operating at speed (Mbps) = ″, ″Interface operating at half duplex. RC = ″, ″Interface operating at full duplex. RC = ″, ″Configured speed does not match neg speed. RC = ″, ″Configured duplex does not match neg duplex. RC = ″, ″Resetting the interface. State = ″, ″Issued -purge all entries- command to TB. State = ″, ″Issued -age- command to TB. Current Age = ″, ″Interface close command received. State = ″, ″Add multicast address command received. State = ″, ″Set LAA MAC address command received. State = ″, ″Setting new age for TB. New age = ″, and ″CAM is full. State = ″.

Cause: ″Performing Unicast frame filtering in software″ indicates that in addition to the adapter performing transparent bridging, the device driver will

also discard any unicast frames received, which do not have a destination MAC matching the local MAC address. Filtering is being performed by the device driver, because the system bridging function has placed this interface in blocking mode.

Action: None.

Cause: ″TB has been enabled″ indicates that transparent bridging support has been enabled on the interface. The interface will now filter incoming packets.

Action: None.

Cause: ″Auto Negotiation is complete″ indicates the interface has sucessfully completed auto-negotiating with its link partner.

Action: None.

Cause: ″Interface operating at speed (Mbps) ″ indicates the speed at which the interface is operating with its link partner.

Action: None.

Cause: ″Interface operating at half duplex″ indicates that the interface is operating in half-duplex mode with its link partner.

Action: None.

Cause: ″Interface operating at full duplex″ indicates that the interface is operating in full-duplex mode with its link partner.

Action: None.

Cause: ″Configured speed does not match hub speed″ indicates that the interface is operating at a speed different from the one configured in the system.

Action: If the operating speed is not desired, then change the value specified for speed in the configuration. Refer to the chapter on ″Configuring and Monitoring the 10/100 Mbps Ethernet Network Interface″ for more details.

Cause: ″Configured duplex does not match hub duplex″ indicates that the interface is operating in a mode different from the one configured in the system.

Action: If the operating mode is not desired, then change the value specified for mode in the configuration. Refer to the chapter on ″Configuring and Monitoring the 10/100 Mbps Ethernet Network Interface″ for more details.

Cause: ″Resetting the interface″ indicates that the interface is undergoing a reset operation. This is always done whenever the interface is enabled or is automatically trying to enable itself. The interface will automatically attempt to enable itself when it detects loss of connectivity on the link or when the system has asked it to perform a self test due to IO failure.

Action: None.

Cause: ″Issued -purge all entries- command to TB″ indicates that all learnt addresses in the interface's tables will be deleted. The interface will begin re-learning MAC address as it receives frames. This is a normal operation as the bridge learns its topology and updates its age value.

Action: None.

Cause: ″Issued -age- command to TB. Current Age = ″ indicates that the inteface will delete all MAC addresses from its tables from which it has has not heard since the previous age event. The age value corresponds to the value configured to the bridging protocol.

Action: None.

Cause: ″Interface close command received″ indicates the interface has been requested to cease all operations. Transmit and receive functions will be suspended until the interface is reactivated.

Action: None.

Cause: ″Add multicast address command received″ indicates that the interface has been given a multicast address to be used in filtering frames. Any frames with destination address equal to the specified multicast address will be filtered and discarded.

Action: None.

Cause: ″Set LAA MAC address command received″ indicates that the interface will overide the use of the assigned Universal MAC address with the locally adminstered address obtained from the configuration.

Action: None.

Cause: ″Setting new age for TB. New age = ″ indicates that the interface has been provided with a new value for aging out old MAC addresses from its tables. The new age indicated is a hexadecimal value.

Action: None.

Cause: ″CAM is full″ indicates that the address table on the interface is full, and there is no room to learn any new source addresses. The interface driver will automatically age out old entries when this condition occurs to make room for more new addresses.

Action: None.

Cause: ″Burnt-in UAA is used due to illegitimate LAA″ indicates that the user specified locally administered MAC address is invalid. The universal address assigned to the interface will be used instead.

Action: If the locally adminstered address must be used on this interface, change the configuration providing a legal locally adminstered MAC address and restart the system.

ETH.050

Level: UI-ERROR

Short Syntax: ETH.050 Eth Diag. *related_msg_index diag1_desc diag1_val diag2_desc diag2_val diag3_desc diag3_val* nt *network*

Long Syntax: ETH.050 Ethernet adapter dagnostics: *related_msg_index diag1_desc diag1_val diag2_desc diag2_val diag3_desc diag3_val* network *network*

Description: Diagnostics information for a previous Ethernet port message.

Action: None.

Panic ethbdtbl

Short Syntax: ethbdtbl: eth_llc tbl out of date

Description: The Ethernet LLC table is out of date.

Action: Contact customer service.

Panic ethintm

Short Syntax: ethintm: net intf mismtch

Description: The Ethernet data structure ″net″ is not Ethernet related.

Action: Contact customer service.

Panic ethbprt

Short Syntax: ethbprt: bad prot init

Description: An unsupported Network Layer protocol tried to initialize Ethernet handler.

Action: Contact customer service.

Panic ethbipx

Short Syntax: ethbipx: bad IPX rqst shd be 8137

Description: An unsupported IPX packet was given to the Ethernet handler for transmission.

Action: Contact customer service.

Panic ethbreq

Short Syntax: ethbreq: bad xmit rqst

Description: An unsupported protocol packet was given to the Ethernet handler for transmission.

Action: Contact customer service.

Panic ethtbig

Short Syntax: ethtbig: bad xmit rqst pkt too lg

Description: A packet was given to the Ethernet handler for transmission that was too large.

Action: Contact customer service.

Panic ethnbuf

Short Syntax: ethnbuf: no buf to set addr

Description: A buffer to set an Ethernet address could not be allocated.

Action: Contact customer service.

Panic ethsrtmcr

Short Syntax: ethsrtmcr: multicast address previously reserved

Description: One of the multicast addresses enabled on this interface is one of the multicast addresses in the range 01-80-C2-00-00-00 through 01-80-C2-00-00-0F.

Cause: Possibly one of these addresses that is being used by a protocol where the user can select the multicast address, such as the ES-IS and IS-IS protocols in ISO.

Action: Don’t use the reserved addresses.

Panic ethsrtnm

Short Syntax: ethsrtnm: no memory to register own MAC addr

Description: The learning database is so small that there are not enough free entries to learn the address of this interface.

Action: Increase the size of the learning database.

Panic ethsrtnmm

Short Syntax: ethsrtnmm: no memory to register mutlicast address

Description: The learning database is so small that there are not enough free entries to one of the multicast addresses of this interface.

Action: Increase the size of the learning database.

Fatal ethsrtun

Short Syntax: ethsrtun: unsupported command

Description: An unsupported command was given by the SRT protocol

Chapter 43. EventLog (EVL)

This chapter describes EventLog (EVL) messages. For information on message content and how to use the message, refer to the Introduction.

EVL.001

Level: ALWAYS

Short Syntax: EVL.001 EventLog() software error: type= *event type*, action= *action needed*, id= *event id*, sev= *severity*, rc= *return code*, filename= *file name*, lineno= *line number*, msg= *message*

Long Syntax: EVL.001 EventLog() software error: type= *event type*, action= *action needed*, id= *event id*, sev= *severity*, rc= *return code*, filename= *file name*, lineno= *line number*, msg= *message*

Description: Software has logged an error via EventLog()

Chapter 44. Fiber Distributed Data Interface (FDDI)

This chapter describes Fiber Distributed Data Interface (FDDI) messages. For information on message content and how to use the message, refer to the Introduction.

FDDI.001

Level: UI-ERROR

Short Syntax: FDDI.001 *setup_phase* fld - bff unav nt *network*

Long Syntax: FDDI.001 *setup_phase* failed, no buffer available net *network*

Description: There were no iorbs available for an ioctl-type function such as starting the self-test or updating statistics. The net may be marked down.

FDDI.002

Level: UI-ERROR

Short Syntax: FDDI.002 PLL error nt *network*

Long Syntax: FDDI.002 Elasticity buffer error detected net *network*

Description: There was an elasticity buffer overrun or underrun detected and the recovery sequence was started.

FDDI.003

Level: CI-ERROR

Short Syntax: FDDI.003 tx fld nt *network*

Long Syntax: FDDI.003 Transmit failed on network *network*

Description: This message is generated when a FDDI packet is added to the transmit queue and the transmission fails.

Cause: Normal when there is no network connection.

Action: Check the network connection.

FDDI.004

Level: CI-ERROR

Short Syntax: FDDI.004 rx fld nt *network*

Long Syntax: FDDI.004 Receive failed on network *network*

Description: This message is generated when a FDDI packet is received and it can can not be added to the receive queue.

FDDI.005

Level: C-INFO

Short Syntax: FDDI.005 *setup_phase*, nt *network*

Long Syntax: FDDI.005 *setup_phase*, network *network*

Description: FDDI adapter initialization in progress. Prior to executing phase.

FDDI.008

Level: P-TRACE

Short Syntax: FDDI.008 Trace FDDI frame

Long Syntax: FDDI.008 Trace FDDI frame

Description: FDDI packet tracing.

FDDI.009

Level: P-TRACE

Short Syntax: FDDI.009 Rcvd pkt *source_MAC* -> *destination_MAC* nt *network* wi RIF ln *RIF_length*

Long Syntax: FDDI.009 Received packet from *source_MAC* to *destination_MAC* network *network* with RIF length *RIF_length*

Description: This message is generated when a FDDI packet with source routing information is received.

FDDI.010

Level: P-TRACE

Short Syntax: FDDI.010 Txd pkt *source_MAC* -> *destination_MAC* nt *network* ln

Long Syntax: FDDI.010 Transmitted packet from *source_MAC* to *destination_MAC* network *network* length

Description: This message is generated when a FDDI packet is transmitted.

FDDI.011

Level: P-TRACE

Short Syntax: FDDI.011 Rxd pkt *source_MAC* -> *destination_MAC* nt *network* ln

Long Syntax: FDDI.011 Received packet from *source_MAC* to *destination_MAC* network *network* length

Description: This message is generated when a FDDI packet is received.

FDDI.012

Level: U-TRACE

Short Syntax: FDDI.012 unkn SNAP type *type_code source_MAC* -> *destination_MAC* nt *network*

Long Syntax: FDDI.012 Unknown SNAP type *type_code* from *source_MAC* to *destination_MAC* net *network*

Description: This message is generated when a frame with an unknown SNAP type (within organization code 000000) is received.

Cause: Host sending packets for unknown Ethernet type using SNAP.

FDDI.013

Level: U-TRACE

Short Syntax: FDDI.013 unkn SNAP mfr cd *number source_MAC* -> *destination_MAC* nt *network*

Long Syntax: FDDI.013 Unknown SNAP manufacturer code *number* from *source_MAC* to *destination_MAC* net *network*

Description: This message is generated when a frame with an unknown organization code in the SNAP header is received.

Cause: Host sending packets for unknown proprietary protocol using SNAP.

FDDI.014

Level: U-TRACE

Short Syntax: FDDI.014 unexp *type* frm *LLC_control* ssap *source_SAP* dsap *dest_SAP source_MAC* -> *destination_MAC* nt *network*

Long Syntax: FDDI.014 Unexpected *type* frame *LLC_control*, ssap *source_SAP*, dsap *dest_SAP*, from *source_MAC* to *destination_MAC* net *network*

Description: This message is generated when an unexpected 802.2 LLC frame type is received. Type may be I (information transfer) or S (supervisory).

Cause: Host attempting to make 802.2 type 2 connection to router.

FDDI.015

Level: U-TRACE

Short Syntax: FDDI.015 unexp U frm *LLC_control* ssap *source_SAP* dsap *dest_SAP source_MAC* -> *destination_MAC* nt *network*

Long Syntax: FDDI.015 Unexpected U frame *LLC_control*, ssap *source_SAP*, dsap *dest_SAP*, from *source_MAC* to *destination_MAC* net *network*

Description: This message is generated when an unexpected 802.2 LLC U (unnumbered) frame type is received. (Only UI, XID, and TEST are supported.)

FDDI.016

Level: U-TRACE

Short Syntax: FDDI.016 unkn SAP *sap_number source_MAC* -> *destination_MAC* nt *network*

Long Syntax: FDDI.016 Unknown SAP *sap_number* from *source_MAC* to *destination_MAC* net *network*

Description: This message is generated when a frame with an unknown destination SAP is received.

Cause: Host sending packets for unknown protocol identifier (SAP).

FDDI.017

Level: U-TRACE

Short Syntax: FDDI.017 xid pkt *source_MAC* src sap *source_sap* nt *network*

Long Syntax: FDDI.017 XID packet received from *source_MAC* source sap *source_sap* net *network*

Description: The handler received an xid message.

FDDI.018

Level: UI_ERROR

Short Syntax: FDDI.018 FC typ *frame_control* unex *source_MAC* -> *destination_MAC* nt *network*

Long Syntax: FDDI.018 Frame Control type *frame_control* unexpected from *source_MAC* to *destination_MAC* network *network*

Description: This message is generated when an unexpected FDDI FC (frame control) is received. (Only LLC is supported by the net handler).

FDDI.019

Level: U-TRACE

Short Syntax: FDDI.019 odd RIF len *source_MAC* -> *destination_MAC*; pkt drpd nt *network*

Long Syntax: FDDI.019 odd RIF length from *source_MAC* to *destination_MAC*; packet dropped on net *network*

Description: The length byte in the RIF header was odd, which is illegal. The packet was dropped.

FDDI.020

Level: U-TRACE

Short Syntax: FDDI.020 drop IPX pkt w/ *encap_seen* encaps - using *encap_used* encaps on int *intnum*

Long Syntax: FDDI.020 dropped IPX pkt with encaps *encap_seen* using *encap_used* on interface *intnum*

Description: This message is generated when an IPX packet is recieved with an encapsulation other than that which has been selected for this interface.

Cause: Normal for networks using multiple encapsulations on a single wire.

Action: None needed.

FDDI.021

Level: U-TRACE

Short Syntax: FDDI.021 DN bd ln *actual_length claimed_length source_MAC -> destination_MAC* nt *network*

Long Syntax: FDDI.021 DECnet packet received with a bad length actual *actual_length* claimed *claimed_length* from *source_MAC* to *destination_MAC* network *network*

Description: A DECnet packet was received with a length field that was larger than the actual length of the packet.

FDDI.022

Level: C-TRACE

Short Syntax: FDDI.022 test pkt *source_MAC* src sap *source_sap* nt *network*

Long Syntax: FDDI.022 Test packet from *source_MAC* source sap *source_sap* net *network*

Description: The handler received a test message.

FDDI.023

Level: C-TRACE

Short Syntax: FDDI.023 Rsp pkt *source_MAC* src sap *source_sap* nt *network*

Long Syntax: FDDI.023 RESPONSE packet received from *source_MAC* source sap *source_sap* net *network*

Description: The handler received a response message.

Panic fddialp

Short Syntax: fddialp: Can't allocate fddi pernet structure

Description: Cannot allocate the network specific FDDI structure.

Panic fddibprt

Short Syntax: fddibprt: bad prot init

Description: An unsupported Network Layer protocol tried to initialize the FDDI handler.

Action: Contact customer service.

Panic fddibreq

Short Syntax: fddibreq: bad xmit rqst

Description: An unsupported protocol packet was given to the FDDI handler for transmission.

Action: Contact customer service.

Chapter 45. Frame Relay Boundary Access Node (BAN)

This chapter describes Frame Relay Boundary Access Node (BAN) messages. For information on message content and how to use the message, refer to the Introduction.

BAN.001

Level: C-INFO

Short Syntax: BAN.001 T *direction*:I-FRM port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Long Syntax: BAN.001 T *direction*:I-FRAME port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Description: I-frame packet. Direction is "in" or "out" of the router. Shows the bridge port number. Shows the full RIF (routing information field) of the packet. Shows up to 20 bytes of the packet after the SSAP field. The length is the amount of data in the packet after the SSAP field in the packet.

BAN.002

Level: P-TRACE

Short Syntax: BAN.002 T *direction*:RR port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Long Syntax: BAN.002 T *direction*:RR port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Description: RR packet.

BAN.003

Level: UI-ERROR

Short Syntax: BAN.003 prt *bridge_port* not a bdg prt

Long Syntax: BAN.003 port *bridge_port* is not a bridge port

Description: The configured BAN bridge port is not a bridge port.

Cause: This is a configuration inconsistency.

Action: Correct configuration.

BAN.004

Level: CI-ERROR

Short Syntax: BAN.004 prt *bridge_port* initialized

Long Syntax: BAN.004 port *bridge_port* initialized

Description: The configured BAN bridge port has been initialized from a BAN point of view.

BAN.005

Level: UI-ERROR

Short Syntax: BAN.005 frm drp *source_mac-> dest_mac*, not BNI src addr *bni_mac*, prt *bridge_port*

Long Syntax: BAN.005 frame dropped *source_mac-> dest_mac*, not BNI source address *bni_mac*, port *bridge_port*

Description: Every frame sent by the NCP must match the configured Boundary Node Identifier (BNI) MAC address.

Cause: This is an NCP address configuration inconsistency between the NCP and the router.

Action: Correct configuration either on the NCP or the router.

Cause: This BAN bridge port is not connected to an NCP.

Action: Check bridge configuration. Check Frame Relay DLCI connections and configuration. Check cable connections.

Cause: BAN has mistakenly been configured on this port.

Action: Remove this port from the BAN configuration.

BAN.006

Level: UI-ERROR

Short Syntax: BAN.006 prt *bridge_port* not a FR bdg prt

Long Syntax: BAN.006 port *bridge_port* is not a Frame Relay bridge port

Description: The configured BAN bridge port is not a Frame Relay DLCI bridge port.

Cause: This is a configuration inconsistency. BAN ports can only be on Frame Relay DLCI bridge ports.

Action: Correct configuration.

BAN.008

Level: U-INFO

Short Syntax: BAN.008 frm flt, prt *bridge_port*, OUI br type *oui_type*

Long Syntax: BAN.008 frm flt, prt *bridge_port* ,OUI br type *oui_type*

Description: The outgoing frame was filtered by BAN because it is was NOT an RFC 1490 bridged Token-Ring frame, without preserved FCS, which is OUI type 9. This is the only type of frame expected by the NCP. Another bridge type frame is being sent: 1 and 7 are Ethernet, 2 and 8 are 802.4, 3 is Token-Ring with FCS, 4 and 10 are FDDI, 11 is 802.6, 14 is 802.1d Hello BPDU, 15 is SRB Hello BPDU.

Cause: This is not expected to happen because transparent behavior and the spanning tree are always forced off for a BAN port.

Action: None. This is harmless.

BAN.009

Level: C-TRACE

Short Syntax: BAN.009 frm flt *src_addr*-> *dest_addr*, prt *bridge_port*, da not BAN DCLI addr *ban_dlci_addr*

Long Syntax: BAN.009 frm flt *src_addr*-> *dest_addr*, prt *bridge_port*, da not BAN DCLI addr *ban_dlci_addr*

Description: The outgoing frame was filtered by BAN because the the frame's destination address did not equal the BAN DLCI address. This is done to protect the NCP.

Cause: The normal bridge logic will try to send to all ports.

Action: None. This is harmless.

BAN.010

Level: C-INFO

Short Syntax: BAN.010 prt *bridge_port* forcing: TB,STP off

Long Syntax: BAN.010 port *bridge_port* forcing: TB,STP off

Description: The BAN bridge port behavior is being dynamically forced to transparent bridging off, spanning tree off. This is required for BAN.

Cause: This is done dynamically as a configuration convenience.

BAN.011

Level: UI-ERROR

Short Syntax: BAN.011 prt *bridge_port* DLSw term but DLSw is not in the build

Long Syntax: BAN.011 port *bridge_port* DLSw term but DLSw is not in the build

Description: The BAN bridge port is configured for DLSw terminated. However, DLSw is not in this build.

Cause: This is a configuration inconsistency.

Action: Either correct configuration to do bridging instead of DLSw switching on the BAN port, or obtain a build with DLSw in it.

BAN.013

Level: C-TRACE

Short Syntax: BAN.013 frm flt *src_addr*-> *dest_addr*, prt *bridge_port*, DLSw snbn *dls_snbn* not in RIF *rif*

Long Syntax: BAN.013 frm flt *src_addr*-> *dest_addr*, prt *bridge_port*, DLSw snbn *dls_snbn* not in RIF *rif*

Description: The outgoing frame was filtered by BAN because the the BAN port is set to DLSw switching, but the frame was not a DLSw frame since the DLSw source-routing segment and bridge number were not in the RIF (routing information field) of the frame. This is done to protect the NCP.

Cause: This will occur at times since the standard DLSw logic tries to DLSw switch and bridge some of the SNA frames. For example, this will occur for a TEST command frame.

Action: None. This is harmless.

BAN.014

Level: P-TRACE

Short Syntax: BAN.014 T *direction*:RNR port= *bridge_port* len= *len src_mac*-> *dest_mac src_sap*-> *dest_sap rif data*

Long Syntax: BAN.014 T *direction*:RNR port= *bridge_port* len= *len src_mac*-> *dest_mac src_sap*-> *dest_sap rif data*

Description: RNR packet.

BAN.015

Level: P-TRACE

Short Syntax: BAN.015 T *direction*:REJ port= *bridge_port* len= *len src_mac*-> *dest_mac src_sap*-> *dest_sap rif data*

Long Syntax: BAN.015 T *direction*:REJ port= *bridge_port* len= *len src_mac*-> *dest_mac src_sap*-> *dest_sap rif data*

Description: REJ packet.

BAN.016

Level: CI-ERROR

Short Syntax: BAN.016 T *direction*:TST_C port= *bridge_port* len= *len src_mac*-> *dest_mac src_sap*-> *dest_sap rif data*

Long Syntax: BAN.016 T *direction*:TST_C port=

bridge_port len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Description: TEST_C (Test command) packet.

BAN.017

Level: CI-ERROR

Short Syntax: BAN.017 T *direction*:TST_R port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Long Syntax: BAN.017 T *direction*:TST_R port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Description: TEST_R (Test response) packet.

BAN.018

Level: CI-ERROR

Short Syntax: BAN.018 T *direction*:XID_C port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Long Syntax: BAN.018 T *direction*:XID_C port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Description: XID_C (XID command) packet.

BAN.019

Level: CI-ERROR

Short Syntax: BAN.019 T *direction*:XID_R port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Long Syntax: BAN.019 T *direction*:XID_R port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Description: XID_R (XID response) packet.

BAN.020

Level: CI-ERROR

Short Syntax: BAN.020 T *direction*:SABME port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Long Syntax: BAN.020 T *direction*:SABME port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Description: SABME packet.

BAN.021

Level: CI-ERROR

Short Syntax: BAN.021 T *direction*:UA port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Long Syntax: BAN.021 T *direction*:UA port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Description: UA packet.

BAN.022

Level: CI-ERROR

Short Syntax: BAN.022 T *direction*:DM port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Long Syntax: BAN.022 T *direction*:DM port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Description: DM packet.

BAN.023

Level: CI-ERROR

Short Syntax: BAN.023 T *direction*:DISC port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Long Syntax: BAN.023 T *direction*:DISC port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Description: DISC packet.

BAN.024

Level: CI-ERROR

Short Syntax: BAN.024 T *direction*:FRMR port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Long Syntax: BAN.024 T *direction*:FRMR port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Description: FRMR packet.

BAN.025

Level: CI-ERROR

Short Syntax: BAN.025 T *direction*:OTHER port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Long Syntax: BAN.025 T *direction*:OTHER port= *bridge_port* len= *len src_mac-> dest_mac src_sap-> dest_sap rif data*

Description: Other packet than those specifically called out above. Look at the data shown to determine exactly what type of frame it is.

BAN.026

Level: UI-ERROR

Short Syntax: BAN.026 frm drp, prt *bridge_port*, port is DLSw term, but DLSw not intialized

Long Syntax: BAN.026 frame dropped, port *bridge_port*, port is DLSw terminated, but DLSw not intialized

Description: All frames are being dropped. The port has been configured for DLSw termination, but DLSw is not running.

Cause: DLSw was not configured or only partially configured.

Action: Configure DLSw.

Cause: DLSw is not in your software load.

Action: Get a new software load with DLSw.

BAN.027

Level: UI-ERROR

Short Syntax: BAN.027 prt *bridge_port* not a source routing bdg prt

Long Syntax: BAN.027 port *bridge_port* is not a source routing bridge port

Description: The configured BAN bridge port is not a source routing bridge port.

Cause: This is a configuration inconsistency. BAN ports can only be on source routing Frame Relay DLCI bridge ports.

Action: Correct configuration.

BAN.028

Level: UI-ERROR

Short Syntax: BAN.028 frm drp, prt *bridge_port*, port is not intialized

Long Syntax: BAN.028 frame dropped, port *bridge_port*, port is not intialized

Description: All bridged frames, in and out, are being dropped. The BAN port did not initialize.

Cause: Some kind of configuration inconsistency.

Action: Turn on BAN ELS messages and restart the router to find out why BAN did not initialize on this port.

BAN.029

Level: UI-ERROR

Short Syntax: BAN.029 prt *bridge_port*, BAN DLCI addr *ban_dlci_addr* is a duplicate with SR-TB enabled.

Long Syntax: BAN.029 port *bridge_port*, BAN DLCI address *ban_dlci_addr* is a duplicate with SR-TB enabled.

Description: When SR-TB conversion is enabled on the bridge, the BAN DLCI MAC addresses of the bridging DLCI must be unique. This restriction does not apply if SR-TB is disabled.

Cause: The BAN DLCI MAC address for this BAN bridging DLCI is the same as the one used on another BAN DLCI and SR-TB is enabled.

Action: Possible alternative solutions are (1) Turn off SR-TB, if not needed. (2) Use DLSw terminated mode on the DLCI instead of the bridging mode. (3) Do not use multiple DLCIs, if not needed. (4) Use unique BAN DLCI MAC addresses on the DLCIs.

Chapter 46. Frame Relay Network Interface (FRL)

This chapter describes Frame Relay Network Interface (FRL) messages. For information on message content and how to use the message, refer to the Introduction.

FR.001

Level: C-INFO

Short Syntax: FR.001 Frame rcvd, circuit *name* DLCI *circuit* prtcl *protocol* nt *network ID*

Long Syntax: FR.001 Frame received, circuit *name* DLCI = *circuit* protocol type = *protocol*, on network *network ID*

Description: A LAPD frame had been received on the FR interface.

FR.002

Level: C-INFO

Short Syntax: FR.002 Frame xmitted circuit *name* DLCI *circuit* prtcl *protocol* nt *network ID*

Long Syntax: FR.002 Frame transmitted circuit *name* DLCI = *circuit* protocol type = *protocol*, on network *network ID*

Description: A LAPD frame had been transmitted on the FR interface.

FR.003

Level: UI-ERROR

Short Syntax: FR.003 Voice fwd disabled circuit *name* DLCI *dlci*, *reason*, nt *network ID*

Long Syntax: FR.003 Error detected in the voice forwarding configuration. Voice forwarding disabled for circuit *name* DLCI *dlci*, *reason* on network *network ID*

Description: An error was detected in the voice forwarding circuit configuration.

FR.004

Level: C-INFO

Short Syntax: FR.004 Circuit outbound congest circuit *name* DLCI *circuit* nt *network ID*

Long Syntax: FR.004 Circuit outbound congestion circuit *name* DLCI = *circuit*, on network *network ID*

Description: The circuit is now experiencing congestion in the outbound direction.

FR.005

Level: C-INFO

Short Syntax: FR.005 Circuit outbound uncongest circuit *name* DLCI *circuit* nt *network ID*

Long Syntax: FR.005 Circuit outbound uncongested circuit *name* DLCI = *circuit* on network *network ID*

Description: The circuit is now not experiencing congestion in the outbound direction.

FR.006

Level: C-INFO

Short Syntax: FR.006 Circuit active circuit *name* DLCI *circuit* nt *network ID*

Long Syntax: FR.006 Circuit enters active state circuit *name* DLCI = *circuit*, on network *network ID*

Description: The circuit enters the active state.

FR.007

Level: C-INFO

Short Syntax: FR.007 Orphan circuit joins net DLCI *circuit* nt *network ID*

Long Syntax: FR.007 An orphan circuit not statically configured has joined the network DLCI = *circuit*, on network *network ID*

Description: An orphan circuit was created for this interface.

FR.008

Level: C-INFO

Short Syntax: FR.008 Circuit becomes inactive circuit *name* DLCI *circuit* nt *network ID*

Long Syntax: FR.008 Circuit enters inactive state circuit *name* DLCI = *circuit*, on network *network ID*

Description: The circuit enters the inactive state.

Cause: The remote end-point on the circuit either is down or is disabled.

FR.009

Level: C-INFO

Short Syntax: FR.009 Circuit becomes unavailable circuit *name* DLCI *circuit* nt *network ID*

Long Syntax: FR.009 Circuit is unavailable circuit *name* DLCI = *circuit*, on network *network ID*

Description: The circuit is no longer available on the network.

Cause: In a LMI message, the Frame Relay switch indicated that the circuit is no longer configured on the network.

FR.010

Level: C-INFO

Short Syntax: FR.010 Circuit becomes available circuit *name* DLCI *circuit* nt *network ID*

Long Syntax: FR.010 Circuit is available circuit *name* DLCI = *circuit*, on network *network ID*

Description: The circuit is now available on the network.

FR.011

Level: C-INFO

Short Syntax: FR.011 LMI seq exchange req rcv seq = *rcvseq* xmt seq = *xmtseq* nt *network ID*

Long Syntax: FR.011 LMI sequence number exchange requested, last received sequence = *rcvseq* current transmit sequence = *xmtseq*, on network *network ID*

Description: A LMI sequence number exchange has been requested.

FR.012

Level: C-INFO

Short Syntax: FR.012 LMI Status Enquiry req rcv seq = *rcvseq* xmt seq = *xmtseq* nt *network ID*

Long Syntax: FR.012 LMI Status Enquiry requested, last received sequence = *rcvseq* current transmit sequence = *xmtseq*, on network *network ID*

Description: A LMI full Status Enquiry has been requested.

FR.013

Level: C-INFO

Short Syntax: FR.013 LMI solicited Status Enquiry response received nt *network ID*

Long Syntax: FR.013 LMI solicited Status Enquiry response had been received on network *network ID*

Description: A solicited LMI Status Enquiry response has been received.

FR.014

Level: C-INFO

Short Syntax: FR.014 LMI *rcvseq* response xmitted, rcv seq = *xmtseq* xmt seq = *network ID*

Long Syntax: FR.014 LMI *rcvseq* number response transmitted, last received sequence = *xmtseq* current transmit sequence = *network ID*

Description: A LMI sequence number exchange has been requested.

FR.015

Level: C-INFO

Short Syntax: FR.015 Modem status change, DCD = *dcd* CTS = *cts* nt *network ID*

Long Syntax: FR.015 Modem status changed DCD = *dcd* CTS = *cts* on network *network ID*

Description: A modem status change has occurred. The present state is described.

FR.016

Level: C-INFO

Short Syntax: FR.016 Multicast frm xmitted circuit *name* DLCI *circuit* prtcl = *protocol* nt *network ID*

Long Syntax: FR.016 Multicast frame transmitted circuit *name* DLCI = *circuit* protocol type = *protocol*, on network *network ID*

Description: A LAPD frame had been transmitted on the FR interface.

FR.017

Level: C-INFO

Short Syntax: FR.017 Circuit remains outbound congest circuit *name* DLCI *circuit* nt *network ID*

Long Syntax: FR.017 Circuit remains congested in the outbound direction circuit *name* DLCI = *circuit*, on network *network ID*

Description: The circuit is remaining in the outbound congested state toward the network.

FR.018

Level: C-INFO

Short Syntax: FR.018 LMI seq *rcvseq* enquiry received, rcv seq = *xmtseq* xmt seq = *network ID*

Long Syntax: FR.018 LMI *rcvseq* number enquiry received, last received sequence = *xmtseq* current

transmit sequence = *network ID*

Description: A LMI sequence number exchange has been requested.

FR.019

Level: C-INFO

Short Syntax: FR.019 Orphan circuit ignored DLCI *circuit* nt *network ID*

Long Syntax: FR.019 An disallowed orphan circuit not statically configured has been ignored DLCI = *circuit*, on network *network ID*

Description: The LMI signalled present and active a circuit which had not been statically configured or allowed.

FR.020

Level: C-INFO

Short Syntax: FR.020 Circuits exceeded, orphan circuit discarded DLCI *circuit* nt *network ID*

Long Syntax: FR.020 The total circuits allowed has been exceeded, an orphan circuit has been ignored DLCI = *circuit*, on network *network ID*

Description: The LMI signalled present and active a circuit which cannot join the interface, maximum circuits have been exceeded.

FR.021

Level: C-INFO

Short Syntax: FR.021 No memory for orphan, circuit discarded DLCI *circuit* nt *network ID*

Long Syntax: FR.021 No available memory for orphan circuit, the circuit has been ignored DLCI = *circuit*, on network *network ID*

Description: In a LMI message, the Frame Relay switch signalled that the circuit is present and available. However, the circuit cannot join the the interface because there is not enough memory to support it.

FR.022

Level: UE-ERROR

Short Syntax: FR.022 Unsupported LMI IE, type = 0x *type* on nt *network ID*

Long Syntax: FR.022 Unsupported LMI information element, type = 0x *type* on network *network ID*

Description: An unsupported management information element has been encountered.

Cause: Software out of date, contact customer service.

FR.023

Level: UE-ERROR

Short Syntax: FR.023 Unsupported LMI *msg_type* type = 0x *type_val* nt *network ID*

Long Syntax: FR.023 Unsupported LMI *msg_type* type encountered = 0x *type_val*, on network *network ID*

Description: An unsupported management LMI message type or report type has been encountered.

Cause: Software out of date, contact customer service.

FR.024

Level: C-INFO

Short Syntax: FR.024 LMI asynch status *stat* reported for circuit *dlci* DLCI *network ID*/

Long Syntax: FR.024 A LMI asynchronous status message indicating a status of *stat* was transmitted for *dlci* DLCI *network ID*/

Description: Asynchronous status sent

FR.025

Level: UE-ERROR

Short Syntax: FR.025 No LIV received, nt *network ID*

Long Syntax: FR.025 LIV not received on network *network ID*

Description: LIV status not received.

FR.026

Level: UE-ERROR

Short Syntax: FR.026 Unsupported NLPID, type = 0x *type*, circuit *name* DLCI = *circuit* on nt *network ID*

Long Syntax: FR.026 Unsupported Network Layer Protocol ID, type = 0x *type* from circuit *name* DLCI = *circuit* on network *network ID*

Description: An unsupported network layer protocol NLPID has been encountered.

Cause: Software out of date or incompatible, contact customer service.

FR.027

Level: UE-ERROR

Short Syntax: FR.027 Unsupported ethertype = 0x *etype* for NLPID = 0x *nlpid*, circuit *name* DLCI = *circuit* on nt *network ID*

Long Syntax: FR.027 Unsupported ethernet type = 0x *etype* for NLPID = 0x *nlpid* from circuit *name* DLCI = *circuit* on network *network ID*

Description: An unsupported ethernet type has been encountered.

Cause: Software out of date or incompatible, contact customer service.

FR.028

Level: UE-ERROR

Short Syntax: FR.028 Unsupported OUI = 0x *oui* with NLPID = 0x80, circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.028 Unsupported organization unique identifier (OUI) = 0x *oui* with NLPID = 0x80 from circuit *name* DLCI = *circuit* on network *network ID*

Description: An unsupported organization unique identifier (OUI) has been encountered in a frame encapsulated using the SNAP NLPID (i.e. 0x80).

Cause: Software out of date or incompatible, contact customer service.

FR.029

Level: UE-ERROR

Short Syntax: FR.029 Received data on invalid circuit, DLCI *circuit* on nt *network ID*

Long Syntax: FR.029 Data received on invalid or nonconfigured circuit, DLCI = *circuit* on network *network ID*

Description: Data has been received on a circuit not configured or learned dynamically but not yet active on network.

Cause: Network mis-configuration or mis-timing.

FR.030

Level: C-INFO

Short Syntax: FR.030 LMI seq exchange received rcv seq = *rcvseq* xmt seq = *xmtseq* nt *network ID*

Long Syntax: FR.030 LMI sequence number exchange received, last received sequence = *rcvseq*current transmit sequence = *xmtseq*, on network *network ID*

Description: A LMI sequence number exchange has been received.

FR.031

Level: C-INFO

Short Syntax: FR.031 LMI unsolicited PVC Status Update received nt *network ID*

Long Syntax: FR.031 LMI unsolicited single Status Update had been received on network *network ID*

Description: An LMI unsolicited single status update message has been received.

FR.032

Level: UE-ERROR

Short Syntax: FR.032 Circuit address length too short nt *network ID*

Long Syntax: FR.032 Circuit address length less than the 2 octet minimum received on network *network ID*

Description: The router encountered a frame on a Frame Relay interface containing an address filed shorter than 2 octets. The router only supports a 2 octet address field on a Frame Relay interface.

FR.033

Level: UE-ERROR

Short Syntax: FR.033 Circuit address length too large nt *network ID*

Long Syntax: FR.033 Circuit address length greater than the 2 octet maximum received on network *network ID*

Description: The router encountered a frame on a Frame Relay interface containing an address field longer than 2 octets. The router only supports a 2 octet address field on a Frame Relay interface.

FR.034

Level: UE-ERROR

Short Syntax: FR.034 Circuit status message using reserved address, DLCI *circuit* nt *network ID*

Long Syntax: FR.034 Circuit status update message contained a reserved management channel address, DLCI = *circuit*, on network *network ID*

Description: The LMI status message contained a reserved management channel address.

FR.035

Level: UE-ERROR

Short Syntax: FR.035 Unsupported control frame, type = 0x *type*, circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.035 Unsupported Link Layer control frame encountered, type = 0x *type* from circuit *name* DLCI = *circuit* on network *network ID*

Description: An unsupported link layer control frame encountered.

Cause: Software out of date or incompatible, contact customer service.

FR.036

Level: UE-ERROR

Short Syntax: FR.036 Unsupported management protocol descriptor, type = 0x *type* on nt *network ID*

Long Syntax: FR.036 Unsupported layer management protocol descriptor encountered, type = 0x *type* on network *network ID*

Description: An unsupported network layer protocol descriptor has been encountered.

Cause: Software out of date or incompatible, contact customer service.

FR.037

Level: UE-ERROR

Short Syntax: FR.037 Unsupported management call reference encountered on nt *network ID*

Long Syntax: FR.037 Unsupported layer management call reference encountered on network *network ID*

Description: An unsupported network layer call reference field has been encountered.

Cause: Software out of date or incompatible, contact customer service.

FR.038

Level: UE-ERROR

Short Syntax: FR.038 No lock shift encountered in ANSI LMI message on nt *network ID*

Long Syntax: FR.038 No lock shift encountered in received ANSI LMI message on network *network ID*

Description: The received ANSI management frame did not include required locking shift information element.

Cause: Error in network switch management frame, contact site administrator.

FR.039

Level: UE-ERROR

Short Syntax: FR.039 Incorrect formatted information element encountered on nt *network ID*

Long Syntax: FR.039 Incorrectly formatted information element encountered on network *network ID*

Description: The received management frame information element was incorrectly formatted.

Cause: Error in network switch management frame, contact site administrator.

FR.040

Level: UE-ERROR

Short Syntax: FR.040 LMI rcv seq number in error seq = *rcvseq* expected seq = *xmtseq* nt *network ID*

Long Syntax: FR.040 LMI receive sequence number in error, receive sequence = *rcvseq* expected sequence = *xmtseq*, on network *network ID*

Description: An incorrect LMI receive sequence number has been received.

FR.041

Level: C-INFO

Short Syntax: FR.041 Circuit leaves net circuit *name* DLCI *circuit* nt *network ID*

Long Syntax: FR.041 A circuit has been removed from the network circuit *name* DLCI = *circuit*, on network *network ID*

Description: The Frame Relay switch did not include the circuit in the last LMI full status message. The circuit is assumed to be removed from the network.

FR.042

Level: C-INFO

Short Syntax: FR.042 Circuit inbound congest circuit *name* DLCI *circuit* nt *network ID*

Long Syntax: FR.042 Circuit experiencing inbound congestion circuit *name* DLCI = *circuit*, on network *network ID*

Description: The circuit is now experiencing congestion in the inbound direction.

FR.043

Level: UE-ERROR

Short Syntax: FR.043 Incorrect formatted addr hdr for LMI packet encountered on nt *network ID*

Long Syntax: FR.043 Incorrect formatted address header for LMI packet encountered on network *network ID*

Description: The address header on received management frame had BECN, FECN, DE or CR bits set.

Cause: Error in network switch management frame, contact site administrator.

FR.044

Level: UE-ERROR

Short Syntax: FR.044 Unsolicited LMI LIV received rcv seq = *xseq* xmt seq = *rseq* nt *network ID*

Long Syntax: FR.044 Unsolicited LMI Link Integrity Verification received receive seq number = *xseq* transmit seq number = *rseq* on network *network ID*

Description: LMI Link Integrity Verification message was received from the network without the router polling for it.

Cause: Duplicate packet may have been sent. Monitor LMI link and contact site administrator.

FR.045

Level: UE-ERROR

Short Syntax: FR.045 Unsolicited LMI FULL STATUS received rcv seq = *xseq* xmt seq = *rseq* nt *network ID*

Long Syntax: FR.045 Unsolicited LMI FULL STATUS response received receive seq number = *xseq* transmit seq number = *rseq* on network *network ID*

Description: LMI Full Status message was received from the network without the router polling for it.

Cause: Duplicate packet may have been sent. Monitor LMI link and contact site administrator.

FR.046

Level: UE-ERROR

Short Syntax: FR.046 DROP: Bridging not enabled on circuit *name* DLCI *circuit*, nt *network ID*

Long Syntax: FR.046 DROP: Bridging not enabled on circuit *name* DLCI = *circuit*, network *network ID*

Description: A frame was received of a bridge type defined in RFC 1490. However, since bridging has not been enabled on this circuit, frame is being discarded.

Cause: In a point-to-point WAN connection, this indicates that bridging is enabled on one end point router and disabled on another. This is an invalid configuration.

Action: Either enable proper bridging behavior on both ends of the circuit or disable bridging on the bridge ports connected to this circuit. In other words, you must enable or disable bridging at both ends of the circuit.

FR.047

Level: C-INFO

Short Syntax: FR.047 DROP: Bridge port not fwding on circuit *name* DLCI *circuit*, nt *network ID*

Long Syntax: FR.047 DROP: Bridge port not forwarding on circuit *name* DLCI = *circuit*, network *network ID*

Description: A bridge frame is being discarded as a bridge port is not in forwarding state.

Cause: It could be that port has just come up and is progressing from blocking to listening to learning to forwarding state, or that Spanning Tree Protocol has determined that this port should stay in blocked state as a backup port.

FR.048

Level: UE-ERROR

Short Syntax: FR.048 DROP: *source_mac* to *dest_mac*, Frame to bdg port behav mismatch on circuit *name* DLCI *circuit*, nt *network ID*

Long Syntax: FR.048 DROP: *source_mac* to *dest_mac*, Frame to bridge port behavior mismatch on circuit *name* DLCI = *circuit*u, network *network ID*

Description: A bridged frame has been received and is being discarded due to mismatch in the frame type versus the bridge port behavior.

Cause: Either a source routed frame was received on a bridge port where source routing is disabled, or a transparent frame was received on a bridge port where transparent bridging is disabled.

Action: Enable proper bridging behavior on both ends of the circuit, or disable bridging on the bridge ports connected to this circuit.

FR.049

Level: UE-ERROR

Short Syntax: FR.049 Unsupported bdg frame type = 0x *type*, circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.049 Unsupported bridge frame type = 0x *type* from circuit *name* DLCI = *circuit* on network *network ID*

Description: An unsupported bridge frame type has been encountered and the frame has been discarded.

Cause: Either a 802.4 bridge frame, a 802.6 bridge frame, or a bridge frame with a bridge protocol ID that is not supported by RFC 1490 has been received.

Action: Ensure compatible bridging behavior is configured on both ends of the circuit and contact customer service if the problem still occurs.

FR.050

Level: UI-ERROR

Short Syntax: FR.050 Unrecgnz outgoing bdg frame type = *type* on circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.050 Unrecognized outgoing bridge frame type = *type* on circuit *name* DLCI = *circuit* on network *network ID*

Description: An unrecognized outgoing bridge frame type. Bridge has asked the frame relay interface to send out a frame whose type cannot be translated into the

encapsulation defined in RFC 1490.

Cause: Software problem

Action: Contact customer service

FR.051

Level: C-INFO

Short Syntax: FR.051 Xmit frame rej: rsn = *reason*, circuit *name* DLCI *circuit*, prtcl = *protocol*, nt *network ID*

Long Syntax: FR.051 Transmit frame rejected with reason = *reason* for circuit *name* DLCI = *circuit* protocol type = *protocol* on network *network ID*

Description: A protocol frame has been rejected because it could not be queued for transmission.

Cause: There is a buffer shortage, the Bandwidth Reservation queue has reached its maximum length, or the interface has gone down.

FR.052

Level: UE-ERROR

Short Syntax: FR.052 LMI rcv seq = 0, prev rcv = *prevrcv_seq*, xmt seq = *xmt_seq* nt *network ID*

Long Syntax: FR.052 LMI receive sequence number = 0, previous receive sequence number = *prevrcv_seq*, current transmit sequence number = *xmt_seq* on network *network ID*

Description: An LMI send sequence number of 0 has been received.

FR.053

Level: UE-ERROR

Short Syntax: FR.053 DN bd ln *actual_length claimed_length*, circuit *name* DLCI *circuit* nt *network id*

Long Syntax: FR.053 DECnet packet received with a bad length actual *actual_length* claimed *claimed_length* on circuit *name* DLCI = *circuit*, network *network id*

Description: A DECnet packet was received with a length field that was larger than the actual length of the packet.

FR.054

Level: UE-ERROR

Short Syntax: FR.054 Rqd PVC *required_pvc* unavail nt *network id*; continue testing

Long Syntax: FR.054 Required PVC *required_pvc* unavailable on network *network id*; continue testing interface

Description: A successful exchange of LMI messages has occurred between the router and the Frame Relay switch. However, the switch has not notified the router that a required PVC is active so the router will continue to test the interface until all required PVCs are active.

Cause: At least one required PVC is not active.

FR.055

Level: UE-ERROR

Short Syntax: FR.055 Rqd PVC *required_pvc* removed nt *network id*; start testing

Long Syntax: FR.055 Required PVC *required_pvc* removed from network *network id*; start testing interface

Description: The router received a LMI message from the Frame Relay switch indicating that a required PVC is no longer active. The router has taken the interface down until that PVC becomes active again.

Cause: A required PVC has become inactive.

FR.056

Level: UE-ERROR

Short Syntax: FR.056 No PVCs present nt *network id*; testing

Long Syntax: FR.056 No PVCs present on network *network id*; testing interface

Description: The router has successfully exchanged LMI messages with the FR switch but the LMI messages indicate no circuits are active. Since the NO-PVC configuration option is enabled on the interface, the router will test the interface until one or more circuits become active.

Cause: No circuits on the interface are active.

FR.057

Level: UE-ERROR

Short Syntax: FR.057 No rpt type in LMI msg nt *network ID*

Long Syntax: FR.057 No report type encountered in received LMI message on network *network ID*

Description: The received LMI did not include the required report type information element.

Cause: Error in FR network switch management frame, contact site administrator.

FR.058

Level: C-INFO

Short Syntax: FR.058 Protocol address cannot be configured for FH PVC *circuit* on nt *network ID*

Long Syntax: FR.058 Protocol address cannot be configured for FH PVC *circuit* on network *network ID*

Description: Protocol addresses cannot be configured for frame handler circuits.

FR.059

Level: UE-ERROR

Short Syntax: FR.059 Frame dropped: APPN or DLSw not enabled on circuit *name* DLCI *circuit*, nt *network ID*

Long Syntax: FR.059 Frame dropped: APPN or DLSw not enabled on circuit *name* DLCI = *circuit*, network *network ID*

Description: A frame was received which used the APPN or SNA encapsulation defined in RFC 1490. However, since neither APPN nor DLSw has been enabled on this circuit, the frame is being discarded.

Cause: This indicates that APPN and SNA traffic is enabled on one end of the circuit and disabled on the other end. This is an invalid configuration.

Action: Either enable or disable APPN or SNA traffic on both ends of the circuit.

FR.060

Level: UE-ERROR

Short Syntax: FR.060 Unsupported L2/L3 PIDs = 0x *l2pid*/0x *l3pid*, circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.060 Unsupported L2 and/or L3 protocol ids = 0x *l2pid*/0x *l3pid* when NLPID = 0x08, circuit *name* DLCI = *circuit* on network *network ID*

Description: A frame was received with a NLPID value of 0x08 but the layer 2 and/or layer 3 protocol ids are not supported for APPN and SNA traffic.

Cause: Software out of date or incompatible, contact customer service.

FR.061

Level: C-TRACE

Short Syntax: FR.061 Info rate changed from *cur_vir* to *new_vir*, circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.061 Information rate changed from *cur_vir* to *new_vir* for circuit *name* DLCI = *circuit* on network *network ID*

Description: The information rate is being changed because either congestion is occurring (a frame was received with BECN set) or congestion is ending (a frame was received without BECN set or no frames have been received for awhile)

FR.062

Level: UE-ERROR

Short Syntax: FR.062 Frame len of *length* too short for circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.062 Frame length of *length* is too short for frame received on circuit *name* DLCI *circuit* on network *network ID*

Description: A frame whose length is shorter than the length of the address field, control field, plus the RFC 1490 encapsulation header was received.

Cause: Software out of date or incompatible, contact customer service.

FR.063

Level: UE-ERROR

Short Syntax: FR.063 FH circuit *name* DLCI *dlci* inactived, *reason*, nt *network ID*

Long Syntax: FR.063 Error detected in the frame handler circuit configuration. Circuit *name* DLCI *dlci* inactived, *reason* on network *network ID*

Description: An error was detected in the voice forwarding circuit configuration.

FR.064

Level: UE-ERROR

Short Syntax: FR.064 Config info missing for required group *groupname* on nt *network ID*

Long Syntax: FR.064 Configuration information missing for required PVC group *groupname* on network *network ID*

Description: A required PVC has been defined as belonging to a required PVC group. The SRAM group information record cannot be located.

Cause: Software (record not written) or hardware, contact customer service.

FR.065

Level: UE-ERROR

Short Syntax: FR.065 All PVCs in rqd group *groupname* unavail nt *network id*; continue testing

Long Syntax: FR.065 All PVCs in required PVC group *groupname* unavailable on network *network id*; continue testing interface

Description: A successful exchange of LMI messages has occurred between the router and the Frame Relay switch. However, the switch has not notified the router that any PVCs in the required PVC group are active, so the router will continue to test the interface until at least one PVC in the group is active.

Cause: All of the circuits in a required PVC group are inactive.

FR.066

Level: UE-ERROR

Short Syntax: FR.066 All PVCs in rqd group *groupname* removed nt *network id*; start testing

Long Syntax: FR.066 All PVCs in required PVC group *groupname* removed from network *network id*; start testing interface

Description: The router received an LMI message from the Frame Relay switch indicating that the last active PVC in a required PVC group is no longer active. The router has taken down the interface until at least one PVC in the group becomes active again.

Cause: All of the circuits in a required PVC group have become inactive.

FR.067

Level: UE-ERROR

Short Syntax: FR.067 Net down due to *type* of *n2evnc N2* LMI errors nt *network id*; start testing

Long Syntax: FR.067 Frame relay LMI detected *type* errors out of *n2evnc N2* consecutive events on network *network id*; start testing interface

Description: The interface has been marked down due to excessive frame relay LMI errors.

Cause: Excessive frame relay LMI errors.

FR.068

Level: UE-ERROR

Short Syntax: FR.068 Asynchronous status message with LIV IE received on nt *network ID*

Long Syntax: FR.068 Asynchronous status message with LIV IE received on network *network ID*

Description: Asynchronous status message with LIV IE received from the network

Action: Contact customer service.

FR.069

Level: C-INFO

Short Syntax: FR.069 CLLM cause *cv* rcvd for PVCs *elsstring* on nt *network ID*

Long Syntax: FR.069 A CLLM message was received with cause value *cv* for PVCs *elsstring* on network *network ID*

Description: A valid CLLM message was received and processed.

FR.070

Level: UE-ERROR

Short Syntax: FR.070 Compress frame disc (bad header) circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.070 Circuit *name* DLCI *circuit* discarded a compression frame (bad header) on network *network ID*

Description: FR compression frame discarded by receive side because of bad header

---

FR.071

Level: UE-ERROR

Short Syntax: FR.071 Compress frame disc (not oper) circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.071 Circuit *name* DLCI *circuit* discarded a compression frame (not oper) - network *network ID*

Description: FR compression frame discarded by receive side - not operational

---

FR.072

Level: UE-ERROR

Short Syntax: FR.072 Frame discarded (decompress err) circuit *name* DLCI *circuit* rc= *returncode* on nt *network ID*

Long Syntax: FR.072 Circuit *name* DLCI *circuit* had a decompression error (rc = *returncode*) on network *network ID*

Description: FR compression frame discarded because of a decompression error

Cause: Decompression error.

FR.073

Level: C-INFO

Short Syntax: FR.073 DCP retry limit exhausted for circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.073 DCP retries exhausted for circuit *name* DLCI *circuit* on network *network ID*

Description: FR compression negotiation retry limit exhausted

---

FR.074

Level: UE-ERROR

Short Syntax: FR.074 Circuit *name* DLCI *circuit* rcvd DCP control PDU out of seq on nt *network ID*

Long Syntax: FR.074 Circuit *name* DLCI *circuit* received DCP ctl PDU out of sequence on network *network ID*

Description: FR compression control frame received out of sequencee

---

FR.075

Level: C-INFO

Short Syntax: FR.075 DCP R-R mode started for circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.075 DCP R-R mode started for circuit *name* DLCI *circuit* on network *network ID*

Description: FR compression reset request (R-R) mode started

FR.076

Level: UE-ERROR

Short Syntax: FR.076 Compress not done (no resources) for circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.076 Compression not done (no resources) for circuit *name* DLCI *circuit* on network *network ID*

Description: Compression not performed - resources not available

Cause: Buffers are not available for the compression function to use.

FR.077

Level: UE-ERROR

Short Syntax: FR.077 Compress frame disc by non-comp circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.077 Compression frame discarded by non-comp circuit *name* DLCI *circuit* on network *network ID*

Description: Compression frame discarded by non-compression circuit

---

FR.078

Level: UE-ERROR

Short Syntax: FR.078 Compress failed for circuit *name* DLCI *circuit* (rc = *returncode*) on nt *network ID*

Long Syntax: FR.078 Compression failed for circuit *name* DLCI *circuit* with rc = *returncode* on network *network ID*

Description: Compression failed

Cause: The data compression algorithm returned a negative return code.

FR.079

Level: UE-ERROR

Short Syntax: FR.079 Compress frame disc (R-R mode) circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.079 Circuit *name* DLCI *circuit* discarded a compression frame (in R-R mode) on network *network ID*

Description: FR compression frame discarded by receive side (in R-R mode)

Cause: Data frame received during compression resynchronization (R-R) mode.

FR.080

Level: UE-ERROR

Short Syntax: FR.080 Compress frame disc (seq err) circuit *name* DLCI *circuit* on nt *network ID - seq_numbers*

Long Syntax: FR.080 Compression frame discarded (seq num err) by circuit *name* DLCI *circuit* on network *network ID - seq_numbers*

Description: Compression frame discarded because of a sequence number error

Cause: Compression data frame received with bad sequence number.

FR.081

Level: C-INFO

Short Syntax: FR.081 Compress frame disc (LCB err) by circuit *circuit* DLCI *network ID/ LCBs -*

Long Syntax: FR.081 Compression frame discarded (LCB err) by circuit *circuit* DLCI *network ID/ LCBs -*

Description: Compression frame discarded because of an LCB error

---

FR.082

Level: C-INFO

Short Syntax: FR.082 DCP R-R mode ended for circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.082 DCP R-R mode ended for circuit *name* DLCI *circuit* on network *network ID*

Description: FR compression reset request (R-R) mode ended

FR.083

Level: C-INFO

Short Syntax: FR.083 Data compress oper circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.083 Data compression operational for

circuit *name* DLCI *circuit* on network *network ID*

Description: FR data compression operational

FR.084

Level: UE-ERROR

Short Syntax: FR.084 Data compress stop circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.084 Data compression stopped for circuit *name* DLCI *circuit* on network *network ID*

Description: FR data compression stopped

Cause: The network is down or compression negotiation has been suspended.

FR.085

Level: CE-ERROR

Short Syntax: FR.085 Circuit *name* DLCI *circuit* on nt *network ID* wait for a compress context

Long Syntax: FR.085 Circuit *name* DLCI *circuit* on network *network ID* is waiting for a compression context

Description: The circuit is waiting for a data compression context

Cause: Compression context not available.

FR.086

Level: CE-ERROR

Short Syntax: FR.086 Circuit *name* DLCI *circuit* on nt *network ID* wait for compress limit to reduce

Long Syntax: FR.086 Circuit *name* DLCI *circuit* on network *network ID* waiting for circuit compression limit to reduce

Description: The circuit is waiting for the circuit compression limit to reduce

Cause: The interface compression PVC limit has been reached.

FR.087

Level: C-INFO

Short Syntax: FR.087 Compress neg suspended circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.087 Compression negotiation suspended for circuit *name* DLCI *circuit* on network *network ID*

Description: Data compression negotiation suspended

FR.088

Level: CE-ERROR

Short Syntax: FR.088 Compression internally disabled for dynamic actv'ed nt *network ID*

Long Syntax: FR.088 Compression internally disabled for dynamically activated network *network ID*

Description: Data compression internally disabled for dynamically activated interface

Cause: Buffer header or trailer size not big enough for compression.

FR.089

Level: UE-ERROR

Short Syntax: FR.089 CLLM msg *elsstring* fmt error offset = *erroffset* on nt *network ID*

Long Syntax: FR.089 A format error in the header of CLLM message *elsstring* was detected at offset *erroffset* (starting from 1) on network *network ID*

Description: An invalid CLLM message was received and discarded.

Cause: Software out of date or incompatible, contact customer service.

FR.090

Level: UE-ERROR

Short Syntax: FR.090 Xmit frame rej: prtcl *protocol* rsn *reason* data *data* nt *network ID*

Long Syntax: FR.090 Transmit frame rejected for protocol *protocol*, reason code = *reason*, associated data = *data* network *network ID*

Description: Frame could not be transmitted due to internal or routing error

Cause: Invalid control block or DLCI pointer.

FR.091

Level: UE-ERROR

Short Syntax: FR.091 Xmit frame rej: inactive or removed circuit DLCI *pvc* prtcl *protocol* nt *network ID*

Long Syntax: FR.091 Transmit frame rejected for inactive or removed circuit DLCI *pvc* for protocol *protocol* network *network ID*

Description: Frame could not be transmitted since the associated circuit was either inactive or removed. If this was a multicast packet, the circuit number will be zero.

Cause: Transmission attempted on an inactive or removed circuit.

FR.092

Level: UE-ERROR

Short Syntax: FR.092 Encrypt frame disc circ *name* DLCI *circuit* on nt *network ID* - multi-encryption not allowed

Long Syntax: FR.092 Encrypt frame disc circ *name* DLCI *circuit* on nt *network ID* - multi-encryption not allowed

Description: Encryption frame discarded because of a multi-encryption attempt.

Cause: Encryption attempted twice.

Action: User needs to reconfigure IP/SEC or PPP/FR so that encryption is not multiply attempted.

FR.093

Level: C-INFO

Short Syntax: FR.093 DCP frm sent circuit *name* DLCI *circuit* (len *length*, seq *seqnum*, lcb *lcb*) - nt *network ID*

Long Syntax: FR.093 DCP frame sent for circuit *name* DLCI *circuit* (len = *length*, seq = *seqnum*, lcb = *lcb*) - network *network ID*

Description: DCP frame with compressed data transmitted

FR.094

Level: C-INFO

Short Syntax: FR.094 DCP frm w/uncomp data sent circuit *name* DLCI *circuit* (len *length*, seq *seqnum*) - nt *network ID*

Long Syntax: FR.094 DCP frame with uncompressed data sent for circuit *name* DLCI *circuit* (len = *length*, seq = *seqnum*) - network *network ID*

Description: DCP frame with uncompressed data transmitted

FR.095

Level: C-INFO

Short Syntax: FR.095 DCP frm rcv'd circuit *name* DLCI *circuit* (len *length*, seq *seqnum*, lcb *lcb*) - nt *network ID*

Long Syntax: FR.095 DCP frame received for circuit *name* DLCI *circuit* (len = *length*, seq = *seqnum*, lcb = *lcb*) - network *network ID*

Description: DCP frame with compressed data received

FR.096

Level: C-INFO

Short Syntax: FR.096 DCP frm w/uncomp data rcv'd circuit *name* DLCI *circuit* (len *length*, seq *seqnum*) - nt *network ID*

Long Syntax: FR.096 DCP frame with uncompressed data received for circuit *name* DLCI *circuit* (len = *length*, seq = *seqnum*) - network *network ID*

Description: DCP frame with uncompressed data received

FR.097

Level: CE-ERROR

Short Syntax: FR.097 MTU = *mtu* too small for P1 = *p1*, MTU = *b* req on nt *network ID*

Long Syntax: FR.097 The MTU defined for the interface of *mtu* is too small to hold a LMI full status message containing P1 *p1* PVCs - MTU of *b* bytes required - network *network ID*

Description: LMI errors will occur if P1 PVCs are included in LMI frames since they can't be received

Cause: MTU size is too small to hold an LMI full status message with P1 PVCs.

Action: Increase MTU or decrease the P1 parameter

FR.098

Level: UE-ERROR

Short Syntax: FR.098 Encrypt frame disc circuit *name* DLCI *circuit* on nt *network ID* - *discard_reason*

Long Syntax: FR.098 Encryption frame discarded by circuit *name* DLCI *circuit* on network *network ID* - *discard_reason*

Description: Encryption frame discarded

FR.099

Level: C-INFO

Short Syntax: FR.099 DEP retry limit exhausted circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.099 DEP retries exhausted for circuit *name* DLCI *circuit* on network *network ID*

Description: FR encryption negotiation retry limit exhausted

FR.100

Level: UE-ERROR

Short Syntax: FR.100 Data encrypt stop circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.100 Data encryption stopped for circuit *name* DLCI *circuit* on network *network ID*

Description: FR data encryption stopped

Cause: The network is down or data encryption negotiation has been suspended.

FR.101

Level: C-INFO

Short Syntax: FR.101 Encrypt neg suspended circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.101 Encryption negotiation suspended for circuit *name* DLCI *circuit* on network *network ID*

Description: Data encryption negotiation suspended

FR.102

Level: C-INFO

Short Syntax: FR.102 Data encrypt oper circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.102 Data encryption operational for circuit *name* DLCI *circuit* on network *network ID*

Description: FR data encryption operational

FR.103

Level: UE-ERROR

Short Syntax: FR.103 Circuit *name* DLCI *circuit* rcvd DEP control PDU out of seq nt *network ID*

Long Syntax: FR.103 Circuit *name* DLCI *circuit* received DEP control PDU out of sequence on network *network ID*

Description: FR encryption control frame received out of sequencee

---

FR.104

Level: UE-ERROR

Short Syntax: FR.104 Frame discarded, *reason*, circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.104 Frame discarded, *reason*, by circuit *name* DLCI *circuit* on network *network ID*

Description: Non-encryption frame discarded by encryption circuit.

FR.105

Level: C-INFO

Short Syntax: FR.105 DEP frm rcvd circuit *name* DLCI *circuit* (len *length*, seq *seqnum*, lcb *lcb*) - nt *network ID*

Long Syntax: FR.105 DEP frame received for circuit *name* DLCI *circuit* (len = *length*, seq = *seqnum*, lcb = *lcb*) - network *network ID*

Description: DEP frame received

FR.106

Level: C-INFO

Short Syntax: FR.106 DEP R-R mode started circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.106 DEP R-R mode started for circuit *name* DLCI *circuit* on network *network ID*

Description: FR encryption reset request (R-R) mode started

FR.107

Level: UE-ERROR

Short Syntax: FR.107 Encrypt not done (no resources) circuit *name* DLCI *circuit* on nt *network ID*

Long Syntax: FR.107 Encryption not done (no resources) for circuit *name* DLCI *circuit* on network *network ID*

Description: Encryption not performed - resources not available

Cause: Buffers are not available for the data encryption function to use.

FR.108

Level: C-INFO

Short Syntax: FR.108 DEP frm sent circuit *name* DLCI *circuit* (len *length*, seq *seqnum*, lcb *lcb*) - nt *network ID*

Long Syntax: FR.108 DEP frame sent for circuit *name* DLCI *circuit* (len = *length*, seq = *seqnum*, lcb = *lcb*) - network *network ID*

Description: DEP frame transmitted

FR.109

Level: UE-ERROR

Short Syntax: FR.109 Xmit frame rej: encrypt not oper circuit *name* DLCI *circuit* prtcl *protocol* nt *network ID*

Long Syntax: FR.109 Protocol xmit request rejected: secure connection not operational yet - circuit *name* DLCI *circuit* protocol *protocol* network *network ID*

Description: Protocol transmission request rejected - secure connection not operational yet

Cause: Transmssion attempted on a secure circuit before encryption operational.

FR.110

Level: UE-ERROR

Short Syntax: FR.110 No memory for circuit during reset int, disc circuit *circuit* nt *network ID*

Long Syntax: FR.110 No available memory for a circuit during a DR reset interface, the circuit has been ignored circuit *circuit*, on network *network ID*

Description: Control block memory allocation failed during a dynamic reconfiguration reset interface operation. A circuit was not created.

Cause: Control block core allocation failed during DR reset interface.

FR.111

Level: UE-ERROR

Short Syntax: FR.111 Buffer not available on nt *network ID* for *transmission_reason*

Long Syntax: FR.111 Buffer not available on network *network ID* for *transmission_reason*

Description: Transmission failed or delayed because buffer not available.

FR.112

Level: C-INFO

Short Syntax: FR.112 SVC call-in ignored, *reason* nt *network ID*

Long Syntax: FR.112 An incoming call for a switched virtual circuit was ignored, *reason* on network *network ID*

Description: A call-in was not accepted.

FR.113

Level: C-INFO

Short Syntax: FR.113 SVC call-in accept, SVC *name*, DLCI *num* nt *network ID*

Long Syntax: FR.113 An incoming call for a switched virtual circuit was accepted, SVC = *name*, DLCI = *num* on network *network ID*

Description: A call-in was accepted.

FR.114

Level: C-INFO

Short Syntax: FR.114 SVC call-out ignored, *reason* nt *network ID*

Long Syntax: FR.114 An outgoing call for a switched virtual circuit was ignored, *reason* on network *network ID*

Description: The call-out was not placed.

FR.115

Level: C-INFO

Short Syntax: FR.115 SVC call-out accept, SVC *name* call state = *state* nt *network ID*

Long Syntax: FR.115 An outgoing call for a switched virtual circuit was accepted, SVC = *name* call state = *state* on network *network ID*

Description: The call-out was accepted.

FR.116

Level: C-INFO

Short Syntax: FR.116 SVC call-out complete, SVC *name*, DLCI *num* nt *network ID*

Long Syntax: FR.116 An outgoing call for a switched virtual circuit was completed, SVC = *name*, DLCI = *num* on network *network ID*

Description: The call-out was completed.

FR.117

Level: C-INFO

Short Syntax: FR.117 SVC disconnect, *reason*, SVC *name* nt *network ID*

Long Syntax: FR.117 A switched virtual circuit was disconnected, *reason*, SVC = *name* on network *network ID*

Description: The call will be cleared.

FR.118

Level: C-INFO

Short Syntax: FR.118 Establish sent to q922 layer, nt *network ID*

Long Syntax: FR.118 Establish sent to q922 layer,network *network ID*

Description: The call will be cleared.

FR.119

Level: UE-ERROR

Short Syntax: FR.119 SVC call-out failed, *reason*, SVC *name* nt *network ID*

Long Syntax: FR.119 A switched virtual circuit call-out attempt failed, *reason*, SVC = *name* on network *network ID*

Description: The call will be cleared.

FR.120

Level: C-INFO

Short Syntax: FR.120 SVC *name* connected number *num*, connected subaddress *sub*, nt *network ID*

Long Syntax: FR.120 An outgoing call for a switched virtual circuit was completed, SVC = *name*, with a different connected number *num* and/or subaddress *sub* than was called on network *network ID*

Description: The call-out was accepted.

FR.121

Level: UE-ERROR

Short Syntax: FR.121 Net down due to failure of Q.922 on nt *network id*; start testing

Long Syntax: FR.121 Frame relay detected a failure of the Q.922 data link layer on network *network id*; start testing interface

Description: The interface has been marked down due to failure of Q.922.

FR.122

Level: UE-ERROR

Short Syntax: FR.122 Invalid Q.922 message. Unknown data received on nt *network id*

Long Syntax: FR.122 An invalid Q.922 message was received and could not be processes on network *network id*

Description: The message is ignored.

FR.123

Level: C-INFO

Short Syntax: FR.123 Establish confirm received on nt *network id*

Long Syntax: FR.123 Establish confirm received byt the q933 layer for network *network id*

Description: Q922 layer has come up.

FR.124

Level: UE-ERROR

Short Syntax: FR.124 Out of sequence numbers Vr *Vr,*, Vs *Vs,*, Nr *Nr,*, Ns *Ns,Va,*, Va *network id/*

Long Syntax: FR.124 Out of sequence numbers Vr *Vr,*,Vs *Vs,*,Nr *Nr,*,Ns *Ns,Va,*,Va *network id/*

Description: Out of sequence numbers in data frames, Q922 session will reset.

FR.125

Level: UE-ERROR

Short Syntax: FR.125 Invalid Q.933 message, *reason*, nt *network ID*

Long Syntax: FR.125 An invalid Q.933 message was received and ignored, *reason*, on network *network ID*

Description: The message is ignored.

FR.126

Level: CI-ERROR

Short Syntax: FR.126 Frame rcvd, circuit *name* DLCI *circuit* not ready for prtcl *protocol* nt *network ID*

Long Syntax: FR.126 Frame received, circuit *name* DLCI = *circuit* not ready for protocol type = *protocol*, on network *network ID*

Description: IPX frames are being recieved but the corresponding circuit is either marked as not ready to receive or is non-existent.

Cause: Frames are being received and the IPX circuit is disabled

Action: Enable the IPX circuit

Cause: Frames are being received and IPX is not configured on the interface

Action: Configure an IPX circuit on the interface

FR.127

Level: P_TRACE

Short Syntax: FR.127 LMI frame transmitted, nt *network ID*

Long Syntax: FR.127 LMI frame transmitted on network *network ID*

Description: LMI packet trace information.

FR.128

Level: P_TRACE

Short Syntax: FR.128 LMI frame received, nt *network ID*

Long Syntax: FR.128 LMI frame received on network *network ID*

Description: LMI packet trace information.

FR.129

Level: P_TRACE

Short Syntax: FR.129 Q.922 frame transmitted, nt *network ID*

Long Syntax: FR.129 Q.922 frame transmitted on network *network ID*

Description: Q.922 packet trace information.

FR.130

Level: P_TRACE

Short Syntax: FR.130 Q.922 frame received, nt *network ID*

Long Syntax: FR.130 Q.922 frame received on network *network ID*

Description: Q.922 packet trace information.

FR.131

Level: P_TRACE

Short Syntax: FR.131 Q.933 frame forwarded, nt *network ID*

Long Syntax: FR.131 Q.933 frame forwarded on network *network ID*

Description: Q.933 packet trace information.

FR.132

Level: P_TRACE

Short Syntax: FR.132 Q.933 frame received, nt *network ID*

Long Syntax: FR.132 Q.933 frame received on network *network ID*

Description: Q.933 packet trace information.

FR.133

Level: P_TRACE

Short Syntax: FR.133 DATA frame received, nt *network ID*

Long Syntax: FR.133 DATA frame received on network *network ID*

Description: DATA packet trace information.

FR.134

Level: P_TRACE

Short Syntax: FR.134 CLLM frame received, nt *network ID*

Long Syntax: FR.134 CLLM frame received on network *network ID*

Description: CLLM packet trace information.

FR.135

Level: UE-ERROR

Short Syntax: FR.135 SVC call collision, *reason*, SVC *name* nt *network ID*

Long Syntax: FR.135 A switched virtual circuit setup was received for an SVC that had a call setup in progress, *reason*, SVC = *name* on network *network ID*

Description: The SVC with the lesser local party number will cancel it's call-out and accept the call-in.

FR.136

Level: UE-ERROR

Short Syntax: FR.136 Invalid fragmentation header received on nt *network id*

Long Syntax: FR.136 An invalid fragmentation header was received in an incoming fragmented frame *network id*

Description: The frame is discarded.

FR.137

Level: UE-ERROR

Short Syntax: FR.137 Fragmentation problem, reason *reason*, dlci *dlci*, nt *network ID*

Long Syntax: FR.137 A problem has been encountered by the fragmentation code, reason *reason*, dlci *dlci* on network *network ID*

Description: Fragmentation of the incoming or outgoing buffer will not be done.

FR.138

Level: UE-ERROR

Short Syntax: FR.138 Encryption not started on network *network ID - reason*

Long Syntax: FR.138 Encryption not started on network *network ID - reason*

Description: Encryption not started because of a lack of resources

Cause: Encryption not started because of a lack of resources.

FR.139

Level: UE-ERROR

Short Syntax: FR.139 Compression not started on network *network ID - reason*

Long Syntax: FR.139 Compression not started on network *network ID - reason*

Description: Compression not started because of a lack of resources

Cause: Compression not started because of a lack of resources.

FR.140

Level: P_TRACE

Short Syntax: FR.140 FR fragment forwarded, nt *network ID*

Long Syntax: FR.140 FR fragment forwarded on network *network ID*

Description: FR fragment packet trace information.

FR.141

Level: P_TRACE

Short Syntax: FR.141 FR fragment received, nt *network ID*

Long Syntax: FR.141 FR fragment received on network *network ID*

Description: FR fragment packet trace information.

Panic frimem

Short Syntax: Frame Relay interface initialization failed - no memory

Description: The Frame Relay interface failed to allocate sufficient memory to complete initialization.

Action: Contact customer service.

Panic friprt

Short Syntax: FR: unsupported protocol during initialization

Description: The Frame Relay network handler detected an unsupported protocol during initialization.

Action: Contact customer service.

Panic frfprt

Short Syntax: FR: unsupported protocol during frame forward

Description: The Frame Relay network handler detected an unsupported protocol during the protocol frame forward phase.

Action: Contact customer service.

Panic frcompmem

Short Syntax: Frame Relay interface compression initialization failed, no memory.

Description: The Frame Relay interface failed to allocate sufficient memory to complete compression initialization.

Action: Contact customer service.

Panic frencmem

Short Syntax: Frame Relay interface encryption initialization failed, no memory.

Description: The Frame Relay interface failed to allocate sufficient memory to complete encryption initialization.

Action: Contact customer service.

Chapter 47. Gateway (GW)

This chapter describes Gateway (GW) messages. For information on message content and how to use the message, refer to the Introduction.

GW.001

Level: ALWAYS

Short Syntax: GW.001 Copyright 1984 Massachusetts Institute of Technology, Copyright 1989 The Regents of the University of California

Long Syntax: GW.001 Copyright 1984 Massachusetts Institute of Technology, Copyright 1989 The Regents of the University of California

Description: Portions of the original code on which this system was based bear the following copyright notice: Permission to use, copy, modify, and distribute this program for any purpose and without fee is hereby granted, provided that this copyright and permission notice appear on all copies and supporting documentation, the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the program without specific prior permission, and notice be given in supporting documentation that copying and distribution is by permission of M.I.T. M.I.T. makes no representations about the suitability of this software for any purpose. It is provided ″as is″ without express or implied warranty. Copyright (c) 1989 The Regents of the University of California. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. All advertising materials mentioning features or use of this software must display the following acknowledgement: This product includes software developed by the University of California, Berkeley and its contributors. 4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ″AS IS″ AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GW.002

Level: ALWAYS

Short Syntax: GW.002 Portable CGW *router name* Rel *release level* strtd

Long Syntax: GW.002 Portable C Gateway *router name* Release *release level* started

Description: Prints the name of the router (as indicated in the router), and the release level of the software load which has just started in the router.

GW.003

Level: ALWAYS

Short Syntax: GW.003 Unus pkt len *unused_length* nt *network ID*

Long Syntax: GW.003 Unused packet length *unused_length* net *network ID*

Description: The router will not be able to send or receive the last [unused length] bytes of maximum size packets.

Cause: The configuration for the router has dictated a maximum packet size that the software will handle, which is smaller than the Maximum Transmission Unit (MTU) of the network.

Action: If the the buffer size setting on the router has been manually set, modify or remove the buffer size setting in the router. If the message persists, contact customer service.

GW.004

Level: ALWAYS

Short Syntax: GW.004 Sys *queue type* q adv alloc *advisable queue length* excd *actual queue length*

Long Syntax: GW.004 System *queue type* queue

advisory allocation of *advisable queue length* exceeded *actual queue length*

Description: The system has detected that there are probably an insufficent number of buffers for optimal operation. On startup, the maximum number of buffers allocated to either the permanent device input queue or the transient device output queue had exceeded an advisable allocation of the entire buffer pool.

Cause: The router has been configured with overly large routing tables for some protocol.

Action: Ensure that the routing tables for each protocol are of a reasonable size for the network configuration. Memory allocated to routing tables cannot be used for packet buffers.

Cause: The router in question has too many network interfaces for the amount of buffer memory available.

Action: Reduce the number of network interfaces on the router. If there are only a reasonable number of interfaces on the router, or if a (buffer) memory upgrade is available, consider expanding the amount of memory on the router. If the message persists, contact customer service.

Cause: The number of buffers has been manually set to a low number.

Action: Modify or remove the number of buffers setting in the router. If the message persists, contact customer service.

GW.005

Level: ALWAYS

Short Syntax: GW.005 Bffrs: *total created* avail *initially free* idle fair *fair share amount* low *high water mark*

Long Syntax: GW.005 Buffers: *total created* available *initially free* idle fair share *fair share amount* low water *high water mark*

Description: The message gives information about the number of buffers created by the initialization procedure, as well as some information on parameters used by the buffer allocation system. As long as the number of buffers currently free in the router is above the low water mark, any user can allocate buffers. Below that point, any user can allocate buffers, as long as the number is less than the 'fair share'.

GW.006

Level: C-INFO

Short Syntax: GW.006 Pkt frm nt *network ID* for uninit prt, disc

Long Syntax: GW.006 Packet from net *network ID* for uninitialized protocol discarded

Description: An incoming packet was in a protocol which, although recognized, did not have a handler loaded and enabled.

GW.007

Level: C-INFO

Short Syntax: GW.007 Ip err *error_code* nt *network ID*

Long Syntax: GW.007 Input error *error_code* net *network ID*

Description: A device input operation returned an error, along with a device specific error code. The input error counter for that network was incrememented, and any packet associated with that error was probably discarded.

Action: Refer to the Router Hardware Manual under the appropriate interface to see what the specific error codes for this type of interface mean.

GW.009

Level: UI-ERROR

Level: METER

Short Syntax: GW.009 Nt dwn ip rstrt nt *network ID*

Long Syntax: GW.009 Net down for input restart net *network ID*

Description: When the router attempted to queue additional input operations for the network, the network had been disabled for input.

Cause: This is caused by timing windows in the internal operation of the router; an input restart operation was requested, but when the time came to do it, input on the interface had been internally disabled. The condition is detected, and is harmless.

Action: If the message persists, contact customer service.

GW.010

Level: UI-ERROR

Level: METER

Short Syntax: GW.010 Ip q len *queue_length* no ip buf nt *network ID*

Long Syntax: GW.010 Input queue length *queue_length* no input buffer net *network ID*

Description: When the router attempted to queue additional input operations for the network, not enough free buffers were available to refill the input queue to the level desired; the actual level attained is listed.

Cause: If the message occurs on an occasional basis, a traffic peak is causing the router to run short of buffers.

Action: No action is necessary.

Cause: The router is short of buffers for some reason. This may be because there is not enough memory on the router to provide enough buffers.

Action: If there is a shortage of memory for buffers, either add memory to the router or reduce the number of network interfaces on the router. If the message persists, contact customer service.

Cause: The number of buffers may have been manually set low.

Action: Modify or remove the number of buffers setting in the router. If the message persists, contact customer service.

GW.014

Level: UI-ERROR

Level: METER

Short Syntax: GW.014 Nt dwn op rstrt nt *network ID*

Long Syntax: GW.014 Net down for output restart net *network ID*

Description: When the router attempted to queue additional output operations for the network, the network had been disabled for output.

Cause: This is caused by timing windows in the internal operation of the router. An output restart operation was requested, but when the time came to do it, output on the interface had been internally disabled. The condition is detected and is harmless.

Action: If the message persists, contact customer service.

GW.017

Level: UE-ERROR

Short Syntax: GW.017 Intfc hdw mssng nt *network ID*

Long Syntax: GW.017 Interface hardware missing net *network ID*

Description: When the router software went to initialize the network interface for the first time, it discovered that the interface's adapter is not plugged in.

Cause: The adapter is not plugged in.

Action: Follow the procedures to insert the appropriate adapter.

Cause: The interface's adapter is broken.

Action: Contact customer service.

GW.018

Level: U-TRACE

Short Syntax: GW.018 Strt nt slf tst nt *network ID*

Long Syntax: GW.018 Start network self test network *network ID*

Description: A network self-test (see Software Operator's Manual for more information on self-test) has been requested.

GW.019

Level: C-INFO

Short Syntax: GW.019 Slf tst nt *network ID*

Long Syntax: GW.019 Self test network *network ID*

Description: A network self-test (see Software Operator's Manual for more information on self-test) has been started.

GW.020

Level: U-TRACE

Short Syntax: GW.020 Nt pss slf tst nt *network ID*

Long Syntax: GW.020 Network passed self test network *network ID*

Description: A network undergoing self-test (see Software Operator's Manual for more information on self-test) has passed the self-test.

GW.021

Level: C-INFO

Short Syntax: GW.021 Nt up nt *network ID*

Long Syntax: GW.021 Network up network *network ID*

Description: After passing self-test (see Software Operator's Manual for more information on self-test), a network that was previously down has come up.

GW.022

Level: U-TRACE

Short Syntax: GW.022 Nt fld slf tst nt *network ID*

Long Syntax: GW.022 Network failed self test network *network ID*

Description: A network undergoing self-test (see Software Operator's Manual for more information on self-test) has failed the self-test.

Cause: The exact cause is network dependant. Use any trap messages printed by the network handler, along with network specific information as revealed by the CGWCON 'Interface' command, to isolate the problem.

GW.023

Level: UE-ERROR

Short Syntax: GW.023 Nt dwn nt *network ID*

Long Syntax: GW.023 Network down network *network ID*

Description: After failing self-test (see Software Operator's Manual for more information on self-test), a network that was previously up has gone down.

GW.024

Level: U-TRACE

Short Syntax: GW.024 Processing pending disable for nt *network ID*

Long Syntax: GW.024 Processing pending disable for network *network ID*

Description: The user requested that a network be disabled (e.g. with the GWCON disable command) but the network was in the middle of self-test. Since the self-test has just completed, the pending disable will now be performed.

GW.025

Level: UE-ERROR

Short Syntax: GW.025 Nt fld mnt nt *network ID*

Long Syntax: GW.025 Network failed maintainence network *network ID*

Description: The number of maintainence checks (see Software Operator's Manual for more information on maintainence) failed in a given interval has exceeded the allowed limit (see the appropriate Router Hardware Manual for more detail on what the exact numbers are for each interface). A self test (see Software Operator's Manual for more information on maintainence) will be started on the interface; if it fails, the interface will be marked down.

Cause: The exact cause is network dependant. Use any trap messages printed by the network handler, along with network specific information as revealed by the CGWCON 'Interface' command, to isolate the problem. A self-test of the network may reveal additional information.

GW.026

Level: C-TRACE

Short Syntax: GW.026 Mnt nt *network ID*

Long Syntax: GW.026 Maintainence network *network ID*

Description: A maintainence check (see Software Operator's Manual for more information on maintainence) has been started for the indicated interface.

GW.027

Level: CI-ERROR

Short Syntax: GW.027 No pkt fr mnt nt *network ID*

Long Syntax: GW.027 No packet for maintainence network *network ID*

Description: A buffer could not be allocated when needed by network maintaince.

Cause: This may be caused by temporary traffic loads. Many other causes are possible.

Action: If the message persists, contact customer service.

GW.028

Level: U-INFO

Short Syntax: GW.028 Snk dsc pkt prt *protocol* to *next_hop_host*

Long Syntax: GW.028 Sink network discarding packet protocol *protocol* to host *next_hop_host*

Description: A buffer was sent to the sink network, which discarded it with no indication of error to the forwarder.

GW.029

Level: U-INFO

Short Syntax: GW.029 Int dis nt *network ID*

Long Syntax: GW.029 Interface disabled in configuration net *network ID*

Description: The inteface in question was disabled in the configuration and will not come up; it can be started at any time by testing it.

GW.030

Level: U-INFO

Short Syntax: GW.030 *heap_bytes* bytes reserved by *subsystem*

Long Syntax: GW.030 *heap_bytes* bytes of heap reserved by subsystem *subsystem*

Description: At start-up time, one of the router's subsystems has reserved so many bytes of heap memory. This will be subtracted from the router's free memory before the remainder is carved into packet buffers.

GW.031

Level: ALWAYS

Short Syntax: GW.031 IP q alloc fl nt *network ID* avl *number of buffers*

Long Syntax: GW.031 Input queue allocation failed net *network ID* available *number of buffers*

Description: The system has detected that there are probably an insufficent number of buffers for optimal operation. On startup, each of the fast devices are allocated a fixed number of buffers. If these buffers are not available, the particular interface may not perform well.

GW.033

Level: U-INFO

Short Syntax: GW.033 Slf tst req rejected for nt *network ID*

Long Syntax: GW.033 Self-test request rejected for net *network ID*

Description: A self-test has been started for the interface but the interface is in an unusable state so the self-test cannot be performed. Use the GWCON configuration command to determine the interface state that is preventing the self-test from occurring.

Cause: If the interface's state is "Not Present" the adapter is not plugged in.

Action: Follow the procedures to insert the appropriate adapter.

Cause: If the interface's state is "HW Mismatch" then there is a hardware mismatch. A hardware mismatch occurs when the configured adapter type does not match the adapter type that is actually present in the slot.

Action: Follow the procedures to insert the appropriate adapter or to re-configure the interface.

Cause: If the interface's state is "HW Failure" then the interface's adapter is broken.

Action: Contact customer service.

Cause: If the interface's state is "Diagnostics" then the interface's adapter is undergoing diagnostics.

Action: Wait until the diagnostics are complete and then start another self-test.

GW.034

Level: U-TRACE

Short Syntax: GW.034 Nt disabled: nt *network ID*

Long Syntax: GW.034 Network disabled: net *network ID*

Description: The network is being disabled.

GW.035

Level: UI-ERROR

Level: METER

Short Syntax: GW.035 Nt dwn to hst *next_hop_host* nt *network ID*

Long Syntax: GW.035 Net down transmitting to host *next_hop_host* net *network ID*

Description: When the router went to send a packet to a given host, the network interface it had been told to send the packet over to was not up. The output discard counter for that network was incremented, and the packet was returned to the protocol forwarder for attention. Usually, next_hop_host will be the MAC layer address of the next hop router or host that this packet is being sent to. The format of this MAC address depends on the network type. If next_hop_host starts with an @ (at-sign), then that network does not provide a formatted display of MAC addresses, and the following number is the hex address in router memory that the next hop host address is stored at.

Cause: This is caused by timing windows in the internal operation of the router; a packet was queued for output, but when the time came to send it, the interface was down. The condition is detected, and is harmless.

Action: If the message persists, contact customer service.

GW.036

Level: U-INFO

Short Syntax: GW.036 Op ovfl to hst *next_hop_host* nt *network ID*

Long Syntax: GW.036 Output overflow when transmitting to host *next_hop_host* net *network ID*

Description: When the router went to send a packet to a given host, the network output queue was too full, and the packet had to be discarded. The output overflow counter for that network was incremented, and the packet was returned to the protocol forwarder for attention. Usually, next_hop_host will be the MAC layer address of the next hop router or host that this packet is being sent to. The format of this MAC address depends on the network type. If next_hop_host starts with an @ (at-sign), then that network does not provide a formatted display of MAC addresses, and the following number is the hex address in router memory that the next hop host address is stored at.

Cause: This is caused by the offered load in the network being higher than the bandwidth available in the output network. Since the router itself is keeping up with the traffic, there is little it can do; the hosts generating the traffic are simply sending more data than the output network can accommodate.

Action: Increase the speed of the network in question (particularly if it is a slow speed leased line), or take measures to restrict the offered load.

GW.037

Level: C-INFO

Short Syntax: GW.037 Nt dwn, disc pkt to hst *next_hop_host* nt *network ID*

Long Syntax: GW.037 Network down, discarding packet to host *next_hop_host* network *network ID*

Description: Packets waiting for transmission on the network in question were discarded when the network went down. The discard counter for the network in question is incremented. Usually, next_hop_host will be the MAC layer address of the next hop router or host that this packet is being sent to. The format of this MAC address depends on the network type. If next_hop_host starts with an @ (at-sign), then that network does not provide a formatted display of MAC addresses, and the following number is the hex address in router memory that the next hop host address is stored at.

GW.038

Level: C-INFO

Short Syntax: GW.038 User *default name* has logged on

Long Syntax: GW.038 User *default name* has logged on

Description: A new user has logged on to the system.

GW.039

Level: CE-ERROR

Short Syntax: GW.039 Failed logon: ID = *default name*

Long Syntax: GW.039 A logon attempt has failed: user ID = *default name*

Description: Someone attempted to log onto the system but did not supply a correct user-name and password.

GW.048

Level: U-INFO

Short Syntax: GW.048 Int rsvd for rst nt *network ID*

Long Syntax: GW.048 Interface reserved for WAN restoral in configuration net *network ID*

Description: The interface in question has been reserved for WAN restoral in the configuration and will not come up until needed by the WAN restoral process.

GW.049

Level: U-INFO

Short Syntax: GW.049 Patched *variable_name* to *new_value*

Long Syntax: GW.049 Variable *variable_name* has been patched to value *new_value*

Description: The user has patched the value of one of the router's data items accordingly.

GW.050

Level: U-INFO

Short Syntax: GW.050 Patch to *variable_name* failed

Long Syntax: GW.050 Attempt to patch variable *variable_name* has failed

Description: The user has attempted to patch the value of one of the router's data items. The patch failed.

GW.051

Level: UI_ERROR

Short Syntax: GW.051 Wrt SRAM failed blk *block_num,* typ *rec_type_num*

Long Syntax: GW.051 Attempt to write block *block_num,*, record type *rec_type_num* to SRAM has failed

Description: Some code which cannot put out a console message (typically during startup) tried to write SRAM and failed.

GW.052

Level: UI_ERROR

Short Syntax: GW.052 No UDP port avail to sync time

Long Syntax: GW.052 No UDP port available to send time sync request

Description: udp_notify returned 0. Probably, IP is not configured.

GW.053

Level: UI_ERROR

Short Syntax: GW.053 No UDP port avail to srvc time req

Long Syntax: GW.053 No UDP port available to receive time sync requests

Description: udp_notify returned 0. Probably, IP is not configured, or else software error.

GW.054

Level: U-INFO

Short Syntax: GW.054 Ip ovfl nt *network ID*, *count* pkts disc

Long Syntax: GW.054 Input overflow net *network ID*, *count* packets discarded

Description: Packets are arriving on the stated interface too quickly for the router's forwarders to process them; they are discarded before being examined by the router software because of the overload. The count of packets is the number of packets this has happened to since the last time it was attempted to log this message. The input overflow counter for this network ID is incremented.

Cause: This may sometimes be caused by ''broadcast storms'', which are network events caused by combinations of buggy and/or out-of-date software running on network hosts which spread in a chain reaction, typically causing the network to be consumed with back to back packets (often broadcast) for a period of seconds, or occasionally, a minute or two.

Action: If a broadcast storm is happening, fix or disable the responsible hosts.

Cause: It may be simply caused by very heavy load.

Action: If heavy load is the cause, and this message happens frequently, you may be using one of the slower routers in the product line. If there is a faster CPU option available for the router you are using, consider upgrading.

GW.055

Level: UI-ERROR

Short Syntax: GW.055 Nt dwn trans on nt *network ID*

Long Syntax: GW.055 Net down transmitting on net *network ID*

Description: When the router went to send a packet, the network interface it had been told to send the packet over to was not up. The output discard counter for that network was incremented, and the packet was returned to the protocol forwarder for attention.

Cause: This is caused by timing windows in the internal operation of the router; a packet was queued for output, but when the time came to send it, the interface was down. The condition is detected, and is harmless.

Action: If the message persists, contact customer service.

GW.056

Level: UI-ERROR

Short Syntax: GW.056 Nt out dis trans on nt *network ID*

Long Syntax: GW.056 Net output disabled, transmitting on net *network ID*

Description: When the router went to send a packet, the network interface it had been told to send the packet over had packet transmission disabled. The output discard counter for that network was incremented.

Cause: This is caused by timing windows in the internal operation of the router; a packet was queued for output, but when the time came to send it, output on the interface was disabled. The condition is detected, and is harmless.

Action: If the message persists, contact customer service.

GW.057

Level: U-INFO

Short Syntax: GW.057 Op ovfl nt *network ID*

Long Syntax: GW.057 Output overflow when transmitting on net *network ID*

Description: When the router went to send a packet, the network output queue was too full, and the packet had to be discarded. The output overflow counter for that network was incremented, and the packet was returned to the protocol forwarder for attention.

Cause: This is caused by the offered load in the network being higher than the bandwidth available in the output network. Since the router itself is keeping up with the traffic, there is little it can do; the hosts generating the traffic are simply sending more data than the output network can accommodate.

Action: Increase the speed of the network in question (particularly if it is a slow speed leased line), or take measures to restrict the offered load.

GW.058

Level: U-INFO

Short Syntax: GW.058 Op err hst *next_hop_host* nt *network ID*

Long Syntax: GW.058 Output error transmitting to host *next_hop_host* net *network ID*

Description: A packet has not been sucessfully retransmitted. The output error counter for that network is incremented, and the packet is discarded. Usually, next_hop_host will be the MAC layer address of the next hop router or host that this packet is being sent to. The format of this MAC address depends on

the network type. If next_hop_host starts with an @ (at-sign), then that network does not provide formatted display of MAC addresses, and the following number is the hex address in router memory that the next hop host address is stored at.

Cause: If this message occurs more than very rarely, it probably indicates hardware transmission problems on the network in question.

Action: Utilize appropriate level 2 network management tools such as Tokenview (for rings) or a Time Domain Reflectometer (for Ethernet) to isolate and fix the problem.

GW.059

Level: C-INFO

Short Syntax: GW.059 Alloc buff with min *global_buffers* global, *private_buffers* per net

Long Syntax: GW.059 Allocating buffers with mininum of *global_buffers* global buffers, and *private_buffers* buffers per fast input network

Description: The router is going to do the buffer allocation with the specified constraints.

Cause: This is normal on router startup.

GW.060

Level: C-INFO

Short Syntax: GW.060 Buffs alloc with reduction *reduction*

Long Syntax: GW.060 Buffers allocated with reduction by *reduction* of private buffers

Description: The router has completed the buffer allocations. If the input networks could not get all the buffers that were requested, the reduction will be non-zero.

Cause: This message always happens on startup of the router. However, a non-zero reduction indicates that the router is close to being short on buffer memory. The higher the reduction, the more severe the buffer memory shortage. However, the shortage is not so severe that the router will not operate, but performance may be impaired.

Action: Upgrade size of buffer memory. Choose smaller buffer size on those devices (Token-Ring, Serial Line) where that is configurable.

GW.061

Level: UI-ERROR

Short Syntax: GW.061 Priv buff alloc failed, nt *network ID*

Long Syntax: GW.061 Private buffer allocation failed, network *network ID*

Description: The buffer allocation for a private buffer for the specified network failed. This network will have one less buffer than was intended. This message is severe only of it happens many times.

Cause: Shortage of buffer memory. (Particularly if preceeded by ELS message GW.064.)

Action: Upgrade size of buffer memory.

Action: Choose smaller buffer size on those devices (Token-Ring, Serial Line) where that is configurable.

Cause: Shortage of heap memory. (Particularly if preceeded by ELS message GW.063.)

Action: Reduce routing table sizes. Increase size of data memory.

Cause: Buffer allocation routine did not accurately predict how many buffers could be allocated.

Action: On some configurations, some portions of the buffer memory are unuseable. The pre-allocator does not take this into account, so a few buffer allocations may fail.

GW.062

Level: UI-ERROR

Short Syntax: GW.062 Global buff alloc failed after *count*

Long Syntax: GW.062 Global buffer allocation failed after *count* allocated

Description: The buffer allocation for a global buffer failed. The router will have one less global buffer than was intended. This message is severe only of it happens many times, starting at low values of count.

Cause: Shortage of buffer memory. (Particularly if preceeded by ELS message GW.064.)

Action: Upgrade size of buffer memory.

Action: Choose smaller buffer size on those devices (Token-Ring, Serial Line) where that is configurable.

Cause: Shortage of heap memory. (Particularly if preceeded by ELS message GW.063.)

Action: Reduce routing table sizes. Increase size of data memory.

Cause: Buffer allocation routine did not accurately predict how many buffers could be allocated.

Action: On some configurations, some portions of the buffer memory are unuseable. The pre-allocator does not take this into account, so a few buffer allocations may fail.

GW.063

Level: UI-ERROR

Short Syntax: GW.063 Alloc of iorb failed

Long Syntax: GW.063 Allocation of I/O request block failed

Description: Some code in the router was allocating an I/O request block and buffer. The allocation of the I/O request block failed.

Cause: Shortage of heap memory.

Action: Reduce routing table sizes. Increase size of data memory.

GW.064

Level: UI-ERROR

Short Syntax: GW.064 Alloc of buffer failed

Long Syntax: GW.064 Allocation of buffer failed

Description: Some code in the router was allocating an I/O request block and buffer. The allocation of the buffer failed.

Cause: Shortage of buffer memory.

Action: Upgrade size of buffer memory.

Action: Choose smaller buffer size on those devices (Token-Ring, Serial Line) where that is configurable.

GW.065

Level: U-INFO

Short Syntax: GW.065 *heap_bytes* bytes buff reserved by *subsystem*

Long Syntax: GW.065 *heap_bytes* bytes of buffer memory reserved by subsystem *subsystem*

Description: At start-up time, one of the router's subsystems has reserved so many bytes of buffer memory. This will be subtracted from the router's free buffer memory before the remainder is carved into packet buffers.

GW.073

Level: UI_ERROR

Short Syntax: GW.073 Rcv buffs increased to *configured_buffers*, exceeds max of *maximum_buffers*, nt *network ID*

Long Syntax: GW.073 Receive buffers increased to *configured_buffers*, exceeds maximum of *maximum_buffers*, net *network ID*

Description: The user-configured number of receive buffers exceeds the maximum allowed for this interface type. The number of buffers will be reduced to the maximum.

Cause: Excessive number of buffers in Config>SET RECEIVE-BUFFERS command.

Action: Configure for acceptable number of buffers.

GW.074

Level: C_INFO

Short Syntax: GW.074 Rcv buffs increased from *configured_buffers* to *default_buffers*, nt *network ID*

Long Syntax: GW.074 Receive buffers increased from *configured_buffers* to *default_buffers*, net *network ID*

Description: The user configuration is increasing the number of receive buffers on this interface from the default to the configured value.

GW.075

Level: U_INFO

Short Syntax: GW.075 Rcv buffs decreased from *default_buffers* to *configured_buffers*, nt *network ID*

Long Syntax: GW.075 Receive buffers decreased from *default_buffers* to *configured_buffers*, net *network ID*

Description: The user configuration is decreasing the number of receive buffers on this interface from the default to the configured value. This may reduce performance on this interface.

GW.078

Level: U_INFO

Short Syntax: GW.078 Rcv low water changed from *default_low_water* to *configured_low_water*, nt *network ID*

Long Syntax: GW.078 Receive low water level changed from *default_low_water* to *configured_low_water*, net *network ID*

Description: The user configuration is changing the receive low water level on this interface from the default to the configured value. This will change the behavior of flow control for packets received on this interface.

GW.079

Level: UI_ERROR

Short Syntax: GW.079 Int hw err nt *network ID*

Long Syntax: GW.079 Hardware failure detected for net *network ID*

Description: The router detected a hardware failure for the interface in question. The interface will not come up.

Action: Contact customer service.

GW.080

Level: C_INFO

Short Syntax: GW.080 Ext Slt: *String supplied by external device*

Long Syntax: GW.080 External Slot device is: *String supplied by external device*

Description: This serves to identfy the status of the external slot.

GW.083

Level: C-TRACE

Short Syntax: GW.083 LID st *old_state,-> new_state,* nt *network ID*

Long Syntax: GW.083 Line ID state *old_state,* changed to *new_state,*, net *network ID*

Description: FSM trace event.

GW.089

Level: C-TRACE

Short Syntax: GW.089 Match dial addr [*dial_address*] to nt *switched network ID*

Long Syntax: GW.089 Matched inbound destination dial address [*dial_address*] to network *switched network ID*

Description: An inbound call arrived and the specified network is configured to match it. Match the dial_address address string in hex. Empty string is a wildcard and will match a network with any_inbound setting.

GW.096

Level: CI-ERROR

Short Syntax: GW.096 DialRec: bad addr rec: smaller than hdr (name= *parent_name*)

Long Syntax: GW.096 DialRec: bad address record: smaller than header (name= *parent_name*)

Description: The router read a SR_DCADDR (dial circuit address) record shorter than the address record header from SRAM under the displayed destination name. Report this error to customer service.

GW.099

Level: C-TRACE

Short Syntax: GW.099 Dropped lnk due to encap slftst errs nt *switched network ID*

Long Syntax: GW.099 Dropped link due to encapsulator self-test errors network *switched network ID*

Description: The amount of time during which consecutive encapsulator self-test errors occurred exceeded the SET IDLE nnn interval set by the user, so CML dropped the link.

GW.100

Level: UI-ERROR

Short Syntax: GW.100 Bad MP config nt *network ID*

Long Syntax: GW.100 Bad MP config for net *network ID*

Description: The MP net configured is invalid or BRS is on the link.

Cause: Configuration error.

Action: Configure a valid MP net or turn off BRS on the link.

GW.101

Level: UE-ERROR

Short Syntax: GW.101 Intfc hdw mismtch nt *network ID*

Long Syntax: GW.101 Interface hardware mismatch net *network ID*

Description: When the router software went to initialize the network interface for the first time, it discovered a hardware mismatch. A hardware mismatch occurs when the interface's configured adapter type does not match the adapter type that is actually present in the slot.

Cause: There is a hardware mismatch.

Action: Either follow the procedures to insert the configured adapter type or refer to the appropriate manuals to check and correct the interface's configuration.

Cause: The interface's adapter is broken.

Action: Contact customer service.

GW.102

Level: DEBUG

Short Syntax: GW.102 bufget() failed. No more global buffers.

Long Syntax: GW.102 bufget() failed. No more global buffers.

Description: The router has run out of global buffers. When this message is enabled, the box will bughlt if it runs out of buffers.

Panic gwbadhd

Short Syntax: GW: Bd cnf inf nt hdr lngths

Description: Bad configuration information in the load was detected.

Cause: Hand-configured maximum header and trailer sizes are smaller than the actual lengths of at least one network in the router.

Action: Contact customer service.

Panic gwbdntv

Short Syntax: GW: incompatible net table vers

Description: A load with incompatible versions of binary modules has been detected.

Cause: The version number on the network configuration table does not match the version number of the compiled code.

Action: Contact customer service.

Panic gwbdpm

Short Syntax: GW: incompatible P_MAX

Description: A load with incompatible versions of binary modules has been detected.

Cause: The maximum number of protocols in the configuration information does not match the maximum number of protocols in the compiled code.

Action: Contact customer service.

Panic gwbdtm

Short Syntax: GW: incompatible T_MAX

Description: A load with incompatible versions of binary modules has been detected.

Cause: The maximum number of network types in the configuration information does not match the maximum number of network types in the compiler code.

Action: Contact customer service.

Panic gwbdim

Short Syntax: GW: incompatible I_MAX

Description: A load with incompatible versions of binary modules has been detected.

Cause: The maximum number of interface types in the configuration information does not match the maximum number of interface types in the conpiled code.

Action: Contact customer service.

Panic gwnmp

Short Syntax: GW: no mem for prot tbl

Description: No memory was available for a critical system table.

Cause: Insufficient memory was available to allocate either the installed or complete protocol table, or the per network protocol upcalls, early in initialization.

Action: Contact customer service.

Panic gwfrfr

Short Syntax: GW: freeing free buffer

Description: The buffer free routine detected software in the system attempting to free a buffer that has already been freed.

Cause: Software problem that frees the same buffer twice. This is a grave error.

Action: Take a dump of this failure, and send it to customer service.

Panic gwgtgt

Short Syntax: GW: alloc busy buffer

Description: The buffer free routine detected software in the system attempting to allocate a buffer that is already busy.

Cause: Software problem.

Action: Take a dump of this failure, and send it to customer service.

Panic gwifdrv

Short Syntax: GW: net with multiple i_fdrv requests

Description: The buffer allocation routine encountered a network that wanted more than one type of memory per buffer.

Cause: Software problem.

Action: Take a dump of this failure, and send it to customer service.

Panic gwlgwc

Short Syntax: GW: leading buffer guard word corrupted

Description: The code that monitors the packet buffers

detected that the guard word in front of a buffer has been corrupted.

Cause: Software problem.

Cause: Hardware failure.

Action: Take a dump of this failure, and send it to customer service.

Panic gwtgwc

Short Syntax: GW: trailing buffer guard word corrupted

Description: The code that monitors the packet buffers detected that the guard word after the end of a buffer has been corrupted.

Cause: Software problem.

Cause: Hardware failure.

Action: Take a dump of this failure, and send it to customer service.

Panic gwnhifdrv

Short Syntax: GW: no heap mem for i_fdrv

Description: No heap memory available for buffer cache data block.

Cause: Shortage of heap memory.

Action: Reduce routing table sizes. Increase size of data memory.

Action: Take a dump of this failure, and send it to customer service.

Panic gwnbifdrv

Short Syntax: GW: no buff mem for i_fdrv

Description: No buffer memory available for buffer cache data block.

Cause: Shortage of buffer memory.

Action: Upgrade size of buffer memory.

Action: Choose smaller buffer size on those devices (Token-Ring, Serial Line) where that is configureable.

Action: Take a dump of this failure, and send it to customer service.

Fatal gwtfb

Short Syntax: GW: too little buffer memory

Description: The buffer allocation code simply cannot allocate enough input buffers to each network while still leaving a reasonable number of free buffers for the routing protocols. (These constraints are given by ELS message GW.059.)

Cause: Too many devices, or too large a buffer size, for the available amount of buffer memory.

Action: De-configure (or remove) some devices. Upgrade size of buffer memory. Choose smaller buffer size on those devices (Token-Ring, Serial Line) where that is configureable.

Chapter 48. Generic Packet Filter (FLT)

This chapter describes Generic Packet Filter (FLT) messages. For information on message content and how to use the message, refer to the Introduction.

FLT.001

Level: UI-ERROR

Short Syntax: FLT.001 no free mem to create *structure_type*

Long Syntax: FLT.001 No free memory to create a *structure_type*

Description: This message is generated when the filtering subsystem cannot allocate the memory to hold a data structure to hold filtering information. This results in a filter not being built.

FLT.002

Level: U-TRACE

Short Syntax: FLT.002 cant apply fltr (offset *filter_offset*), pkt too shrt (ln *packet_offset*)

Long Syntax: FLT.002 Cannot apply filter (offset *filter_offset*), to packet of length *packet_offset*

Description: This message is generated when the maximum offset in a filter is larger than the length of a packet. The filter is not applied to the packet.

FLT.003

Level: U-TRACE

Short Syntax: FLT.003 no mem to cache pkt (max *cache_entries_allocated*)

Long Syntax: FLT.003 No memory to cache packet (maximum *cache_entries_allocated*)

Description: This message is generated if a filter is attempting to create a cache entry but cannot do so because there is no available memory on the heap. Instead, an existing entry is reused from the filter.

FLT.004

Level: C-INFO

Short Syntax: FLT.004 crtng flt, sys *system_name*

Long Syntax: FLT.004 Creating filter for system *system_name*

Description: A filter is being created for the router system identified by system_name

FLT.005

Level: C-INFO

Short Syntax: FLT.005 flt che hit, sys *system_name*

Long Syntax: FLT.005 Filter cache hit, system *system_name*

Description: A filter produced a cache hit. System_name is the system name of a filter that was previously created.

FLT.006

Level: C-INFO

Short Syntax: FLT.006 flt match, sys *system_name*

Long Syntax: FLT.006 Filter match, system *system_name*

Description: A filter produced a match, but with no cache hit. System_name is the system name of a filter that was previously created.

FLT.007

Level: C-INFO

Short Syntax: FLT.007 flt miss, sys *system_name*

Long Syntax: FLT.007 Filter miss, system *system_name*

Description: A filter was applied to a block a data, but not match was found. System_name is the system name of a filter that was previously created.

Chapter 49. IBM LAN Emulation Client Functions (ILEC)

This chapter describes IBM LAN Emulation Client Functions (ILEC) messages. For information on message content and how to use the message, refer to the Introduction.

ILEC.001

Level: C-INFO

Short Syntax: ILEC.001 ILEC function entry/exit tracing

Long Syntax: ILEC.001 ILEC function entry/exit tracing

Description: The user can enable/disable the function entry and exit tracing of the ILEC by simply turning on/off the display of this message.

ILEC.002

Level: C-INFO

Short Syntax: ILEC.002 nt *network entry_exit log_point*

Long Syntax: ILEC.002 network *network*: ilec trace log: *entry_exit log_point*

Description: ILEC generic function entry/exit

ILEC.003

Level: C-INFO

Short Syntax: ILEC.003 nt *network entry_exit log_point*, D1= *arg1*

Long Syntax: ILEC.003 network *network*: ilec trace log: *entry_exit log_point*, D1= *arg1*

Description: ILEC generic function entry/exit with one arg

ILEC.004

Level: C-INFO

Short Syntax: ILEC.004 nt *network entry_exit log_point*, D1= *arg1*, D2= *arg2*

Long Syntax: ILEC.004 network *network*: ilec trace log: *entry_exit log_point*, D1= *arg1*, D2= *arg2*

Description: ILEC generic function entry/exit with two args

ILEC.005

Level: C-INFO

Short Syntax: ILEC.005 nt *network entry_exit log_point*, D1= *arg1*, D2= *arg2*, D3= *arg3*

Long Syntax: ILEC.005 network *network*: ilec trace log: *entry_exit log_point*, D1= *arg1*, D2= *arg2*, D3= *arg3*

Description: ILEC generic function entry/exit with three args

ILEC.007

Level: UI-ERROR

Short Syntax: ILEC.007 Inbound call rejected, on nt *network ID*, rsn= *reason*, hndl= *conn_handle*, d1= *debug1*, d2= *debug2*

Long Syntax: ILEC.007 Inbound call rejected, on network *network ID*, reason = *reason*, conn handle = *conn_handle*, debug1 = *debug1*, debug2 = *debug2*

Description: Inbound call rejected

ILEC.011

Level: P_TRACE

Short Syntax: ILEC.011 Trace ILEC data packet

Long Syntax: ILEC.011 Trace ILEC data packet

Description: Trace ILEC data packet

ILEC.012

Level: P_TRACE

Short Syntax: ILEC.012 Trace ILEC control packet

Long Syntax: ILEC.012 Trace ILEC control packet

Description: Trace ILEC control packet

ILEC.020

Level: UE-ERROR

Short Syntax: ILEC.020 nt *network error_lvl log_point*

Long Syntax: ILEC.020 network *network*: ilec error log: *error_lvl log_point*

Description: ilec generic error

ILEC.021

Level: UE-ERROR

Short Syntax: ILEC.021 nt *network error_lvl log_point*, D1= *arg1*

Long Syntax: ILEC.021 network *network*: ilec error log: *error_lvl log_point*, D1= *arg1*

Description: ilec generic error with one arg

ILEC.022

Level: UE-ERROR

Short Syntax: ILEC.022 nt *network error_lvl log_point*, D1= *arg1*, D2= *arg2*

Long Syntax: ILEC.022 network *network*: ilec error log: *error_lvl log_point*, D1= *arg1*, D2= *arg2*

Description: ilec generic error with two args

ILEC.023

Level: UE-ERROR

Short Syntax: ILEC.023 nt *network error_lvl log_point*, D1= *arg1*, D2= *arg2*, D3= *arg3*

Long Syntax: ILEC.023 network *network*: ilec error log: *error_lvl log_point*, D1= *arg1*, D2= *arg2*, D3= *arg3*

Description: ilec generic error with three args

ILEC.024

Level: UI-ERROR

Short Syntax: ILEC.024 open frame SAP failed on nt *n_net*, rc= *retcd*

Long Syntax: ILEC.024 open frame SAP failed on network *n_net*, rc = *retcd*

Description: open frame SAP failed

ILEC.025

Level: UI-ERROR

Short Syntax: ILEC.025 open call SAP failed on nt *n_net*, rc= *retcd*

Long Syntax: ILEC.025 open call SAP failed on network *n_net*, rc = *retcd*

Description: open call SAP failed

ILEC.026

Level: UI-ERROR

Short Syntax: ILEC.026 open data path failed for outgoing call, on nt *n_net*, rc= *retcd*

Long Syntax: ILEC.026 open data path failed for outgoing call, on network *n_net*, rc = *retcd*

Description: open data path failed for outgoing call

ILEC.027

Level: UI-ERROR

Short Syntax: ILEC.027 open data path failed for incoming call, on nt *n_net*, rc= *retcd*

Long Syntax: ILEC.027 open data path failed for incoming call, on network *n_net*, rc = *retcd*

Description: open data path failed for incoming call

ILEC.028

Level: C-INFO

Short Syntax: ILEC.028 Function *function_name* called, nt *network ID*

Long Syntax: ILEC.028 Function *function_name* called, on network *network ID*

Description: ATM ILEC function called

ILEC.029

Level: UI-ERROR

Short Syntax: ILEC.029 Start failed, on nt *network ID*, rc= *retcd*

Long Syntax: ILEC.029 Start failed, on network *network ID*, rc = *retcd*

Description: Start failed for ILEC object

ILEC.030

Level: UI-ERROR

Short Syntax: ILEC.030 create ILEC object failed, on nt *network ID*, rc= *retcd*

Long Syntax: ILEC.030 create ILEC object failed, on network *network ID*, rc = *retcd*

Description: Could not create ILEC object

ILEC.031

Level: UI-ERROR

Short Syntax: ILEC.031 usr reg failed, on nt *network ID*, rc= *retcd*

Long Syntax: ILEC.031 user registration failed, on network *network ID*, rc = *retcd*

Description: ILEC could not register

ILEC.032

Level: UI-ERROR

Short Syntax: ILEC.032 nt *network ID*, ATM nt *network ID* nt nbld

Long Syntax: ILEC.032 on network *network ID*, ATM

network *network ID* not enabled

Description: ATM interface not enabled

ILEC.033

Level: UI-ERROR

Short Syntax: ILEC.033 ILEC activate failed, on nt *network ID*, rc= *retcd*

Long Syntax: ILEC.033 ILEC activate failed, on network *network ID*, rc = *retcd*

Description: ILEC activate failed

ILEC.034

Level: UI-ERROR

Short Syntax: ILEC.034 ILEC activate complete, on nt *network ID*, rc= *retcd*

Long Syntax: ILEC.034 ILEC activate complete, on network *network ID*, rc = *retcd*

Description: ILEC activate failed.

ILEC.035

Level: UI-ERROR

Short Syntax: ILEC.035 Outbound frame freed, on nt *network ID*

Long Syntax: ILEC.035 Outbound frame freed, on network *network ID*

Description: Outbound frame freed

ILEC.036

Level: UI-ERROR

Short Syntax: ILEC.036 Outbound frame queued, on nt *network ID*

Long Syntax: ILEC.036 Outbound frame queued, on network *network ID*

Description: Outbound frame queued

ILEC.037

Level: UI-ERROR

Short Syntax: ILEC.037 Transmit failed, on nt *network ID*, rc= *retcd*

Long Syntax: ILEC.037 Transmit failed, on network *network ID*, rc = *retcd*

Description: Transmit failed

ILEC.042

Level: UI-ERROR

Short Syntax: ILEC.042 SRAM nt fnd on dsbl, on nt *network ID*

Long Syntax: ILEC.042 SRAM not found after disable, on network *network ID*

Description: Couldn't find the matching SRAM block after user disabled the ILEC interface.

ILEC.044

Level: UI-ERROR

Short Syntax: ILEC.044 Outbnd frm dscrd, on nt *net_no*,frm sz (*frame_size*) xcds cnfgd frm sz (*config_frame_size*)

Long Syntax: ILEC.044 Outbound frame discarded, on network *net_no*, frame size (*frame_size*) exceeds configured frame size (*config_frame_size*)

Description: An outbound frame was discarded, because the frame's size was larger than the configured frame size.

ILEC.045

Level: UI-ERROR

Short Syntax: ILEC.045 Inbnd frm dscrd, on nt *net_no*,frm sz (*frame_size*) xcds cnfgd frm sz (*config_frame_size*)

Long Syntax: ILEC.045 Inbound frame discarded, on network *net_no*, frame size (*frame_size*) exceeds configured frame size (*config_frame_size*)

Description: An inbound frame was discarded, because the frame's size was larger than the configured frame size.

ILEC.046

Level: UE-ERROR

Short Syntax: ILEC.046 ILEC inbnd fr dscrd, bad FC, on nt *network ID*, *word1 word2 word3 word4 word5*

Long Syntax: ILEC.046 ILEC inbnd fr dscrd, bad FC, on network *network ID*, *word1 word2 word3 word4*x *word5*

Description: ILEC inbound data frame was discarded - bad FC byte

---

ILEC.047

Level: UI-ERROR

Short Syntax: ILEC.047 nt *network ID*:ILEC inbnd fr dscrd, bad frame type (*frame_type*)

Long Syntax: ILEC.047 nt *network ID*:ILEC inbnd fr

dscrd, bad frame type (*frame_type*)

Description: LEC inbound data frame was discarded - wrong frame type

---

ILEC.048

Level: UI-ERROR

Short Syntax: ILEC.048 nt *network ID*:ILEC inbnd fr dscrd, bad frame type (*frame_type*)

Long Syntax: ILEC.048 nt *network ID*:ILEC inbnd fr dscrd, bad frame type (*frame_type*)

Description: LEC inbound data frame was discarded - wrong frame type

---

ILEC.049

Level: UI-ERROR

Short Syntax: ILEC.049 nt *network ID*:ILEC Inbnd frm dscrd, dst = *dest_addr* src = *src_addr*, rsn = *reason*

Long Syntax: ILEC.049 nt *network ID*:ILEC Inbound frame discarded, dest = *dest_addr* source = *src_addr*, reason = *reason*

Description: Inbound frame discarded

Chapter 50. Integrated Services Digital Network (ISDN)

This chapter describes Integrated Services Digital Network (ISDN) messages. For information on message content and how to use the message, refer to the Introduction.

ISDN.001

Level: CE-ERROR

Short Syntax: ISDN.001 I_ERR (0x *status*) len(*msglen*) on rcv nt *network ID*

Long Syntax: ISDN.001 Packet received with I_ERR set (status = 0x *status*) or bad length(*msglen*), on network *network ID*

Description: YDC ISDN: isdny_rx() received a buffer from the driver with the error flag set or with a length less than the minimum.

Action: Report this event to customer service.

ISDN.002

Level: UE-ERROR

Short Syntax: ISDN.002 RX bad type (0x *type*) on nt *network ID*

Long Syntax: ISDN.002 Received an unrecognized packet type (0x *type*), on network *network ID*

Description: YDC ISDN: isdny_rx() received a packet with an unrecognized type.

Action: Report this event to customer service.

ISDN.003

Level: C-INFO

Short Syntax: ISDN.003 ConnID 0x *ConnID* Status msg cause (0x *cause0*:0x *cause1*) *message* on nt *network ID*

Long Syntax: ISDN.003 ConnID (0x *ConnID*) Received a status message from the ISDNcard: Cause field 0x *cause0*:0x *cause1* (*message*) on network *network ID*

Description: isdny_rx() received a status message from the ISDN card.

ISDN.004

Level: C-INFO

Short Syntax: ISDN.004 ConnID 0x *ConnID message displaystring* (cause 0x *cause0*:0x *cause1*) on nt *network ID*

Long Syntax: ISDN.004 ConnID (0x *ConnID*) received a *message* (*displaystring*) from the ISDNcard: Cause field 0x *cause0*:0x *cause1* on network *network ID*

Description: isdny_rx() received an NLS Display Information status message from the ISDN card. This may reflect error conditions at the network interface.

Action: If the network interface will not come up, contact customer service, and/or your local service provider.

ISDN.005

Level: UE-ERROR

Short Syntax: ISDN.005 ConnID 0x *ConnID* Bad msg (0x *message*) in stt *state*, sts 0x *status*, len *length*, cse(0x *cause1*:0x *cause2*) nt *network ID*

Long Syntax: ISDN.005 ConnID 0x *ConnID* received an unexpected message (0x *message*) in state *state*, status 0x *status*, length *length*, cause (0x *cause1*:0x *cause2*) on network *network ID*

Description: isdny_rx() received an unexpected packet in its current state.

Action: Report this event to customer service.

ISDN.006

Level: UE-ERROR

Short Syntax: ISDN.006 Bad Config nt *network ID*

Long Syntax: ISDN.006 The ISDN network interface configuration for network *network ID* is bad.

Description: The configuration of the ISDN network for this port is incomplete, missing, or inconsistent.

Action: Verify that the ISDN configuration for this interface includes at least the Local Address.

ISDN.007

Level: UE-ERROR

Short Syntax: ISDN.007 Download failed (0x *dlstat*), PUD status(0x *pudstat*) nt *network ID*

Long Syntax: ISDN.007 Download of the ISDN network interface card failed with status 0x *dlstat*, Power-Up Diagnostics code 0x *pudstat* for network *network ID*.

Description: Either power-up diagnostics results inhibit download, or the download image itself was corrupted.

Action: Report this event to customer service.

ISDN.008

Level: C-INFO

Short Syntax: ISDN.008 Download OK, PUD status (0x *pudstat*) nt *network ID*

Long Syntax: ISDN.008 Download of the ISDN network interface card succeeded, Power-Up Diagnostics returned 0x *pudstat* for network *network ID*.

Description: Download of the ISDN smart card completed normally.

ISDN.009

Level: UE-ERROR

Short Syntax: ISDN.009 Config bad st (0x *cfgstat*) nt *network ID*

Long Syntax: ISDN.009 The ISDN network interface card rejected configuration with the status 0x *cfgstat* for network *network ID*.

Description: Part of the ISDN smart card configuration is inconsistent or missing.

Action: Report this event to customer service.

ISDN.010

Level: C-INFO

Short Syntax: ISDN.010 Config ok nt *network ID*

Long Syntax: ISDN.010 Configuration of the ISDN network interface card succeeded for network *network ID*.

Description: Configuration of the ISDN smart card completed normally.

ISDN.011

Level: UE-ERROR

Short Syntax: ISDN.011 Board Down DCT flags in (0x *idctst*) out (0x *odctst*) nt *network ID*

Long Syntax: ISDN.011 INIDEV of the ISDN network interface card failed, DCT flags for input and output are 0x *idctst* and 0x *odctst* respectively for network *network ID*.

Description: The ISDN card isn't responding to driver initialization attempts.

Action: Test the network interface. If this does not correct the problem, restart the router. Report this error to customer service.

ISDN.012

Level: UE-ERROR

Short Syntax: ISDN.012 Dead Board nt *network ID*

Long Syntax: ISDN.012 The ISDN network interface card for network *network ID* is dead.

Description: The ISDN interface card is not responding at all. A router restart is required (at minimum).

Action: Verify that the correct slot was specified in the device configuration, and restart the card. If it still fails, reseat the card in the router. Lastly, contact customer service and report a hardware failure.

ISDN.013

Level: U-INFO

Short Syntax: ISDN.013 Board reset complete nt *network ID*

Long Syntax: ISDN.013 The ISDN network interface card for network *network ID* has been reset. Attempting download.

Description: The board crashed. As the first step in recovering, we reset it.

Action: Report this event to customer service.

ISDN.014

Level: UE-ERROR

Short Syntax: ISDN.014 Unexpected state (*state1*) instead of *state2* nt *network ID*

Long Syntax: ISDN.014 ISDN handler state (*state1*) is different from that expected (*state2*) for internal event on network *network ID*.

Description: An event occurred in a state that is inconsistent with the design of the FSM.

Action: Report this event to customer service.

ISDN.015

Level: C-INFO

Short Syntax: ISDN.015 Chn *channel* FSM st *state1* ev *event -> state2* nt *network ID*

Long Syntax: ISDN.015 Channel *channel* FSM transition occurred: old state *state1*, event *event*, new state *state2* on network *network ID*.

Description: An FSM transition occurred.

ISDN.016

Level: U-INFO

Short Syntax: ISDN.016 Chn *channel* ConnID 0x *ConnID* FSM odd stt *state1* ev *event* -> *state2* nt *network ID*

Long Syntax: ISDN.016 Channel *channel* ConnID 0x *ConnID* unusual FSM state transition occurred: old state *state1*, event *event*, new state *state2* on network *network ID*.

Description: A transition occurred in the ISDN handler's channel FSM contrary to the normal path, because of resource shortages, or synchronization problem between the interface card and the router.

Action: Report this event to customer service.

ISDN.017

Level: UE-ERROR

Short Syntax: ISDN.017 Chn *channel* N-CONN-RQ bad iostat 0x *status* nt *network ID*

Long Syntax: ISDN.017 An N-CONN-RQ I/O request for channel *channel* completed with status 0x *status* network *network ID*.

Description: The ISDN handler sent an N-CONN-RQ to the interface card, but the transfer did not complete successfully.

Action: Report this event to customer service.

ISDN.018

Level: UE-ERROR

Short Syntax: ISDN.018 No Hchn *channel* A-DISC-RQ nt *network ID*

Long Syntax: ISDN.018 A client issued a disconnect request for a connection (*channel*) unrecognized by the handler on network *network ID*.

Description: A client of the ISDN handler issued a disconnect request (isdny_client_DR) for a connection unknown to the handler. This indicates a serious synchronization problem between the handler and its client.

Action: Report this event to customer service.

ISDN.019

Level: UE-ERROR

Short Syntax: ISDN.019 Bd stats cmp sts 0x *status* nt *network ID*

Long Syntax: ISDN.019 A statistics request to the interface card was returned with a bad status (0x *status*) for network *network ID*.

Description: The handler for the CNX YDC ISDN card regularly issues statistics requests to the ISDN card, and the status on this request was bad. This may indicate a slight congestion problem on the control queue between the router and the card, or, if it persists, it may indicate a problem with the card.

Action: If this problem persists, test the network interface. If it is a persistent problem, report the event to customer service.

ISDN.020

Level: U-TRACE

Short Syntax: ISDN.020 Chn *channel* ConnID 0x *ConnID* Rxd Dt Pkt ln *msglen* bd stt *state* nt *network ID*

Long Syntax: ISDN.020 Channel *channel* ConnID 0x *ConnID* : received a Data Packet of length (*msglen*) in wrong state (*state*) from network *network ID*.

Description: The handler for the CNX YDC ISDN card received a data packet for the indicated channel, but the channel was not in Data Transfer state. This may delay the establishment of the Serial Link over the connection for several seconds. This usually indicates a misordering in the receipt of signalling and data packets from the network interface.

Action: Report this event to customer service.

ISDN.021

Level: P-TRACE

Short Syntax: ISDN.021 Chn *channel* ConnID 0x *ConnID* RxD Pkt ln *msglen* nt *network ID*

Long Syntax: ISDN.021 Channel *channel* ConnID 0x *ConnID* received a Data Packet of length (*msglen*) from network *network ID*.

Description: The handler for the CNX YDC ISDN card received a data packet for the indicated channel in Data Transfer state.

ISDN.022

Level: U-TRACE

Short Syntax: ISDN.022 ConnID 0x *ConnID* Rxd *msgtype* Pkt ln *msglen* bd stt *state* nt *network ID*

Long Syntax: ISDN.022 ConnID 0x *ConnID* received a *msgtype* Packet of length (*msglen*) in wrong state (*state*) from network *network ID*.

Description: The handler for the CNX YDC ISDN card received a control packet for the indicated channel, but the channel was not in the appropriate state.

Action: Report this event to customer service.

ISDN.023

Level: C-TRACE

Short Syntax: ISDN.023 ConnID 0x *ConnID* Rxd N_STAT_IN ln *msglen* cause 0x *cause1*:0x *cause2* nt *network ID*

Long Syntax: ISDN.023 ConnID 0x *ConnID* received a N_STAT_IN message of length (*msglen*) cause 0x *cause1*:0x *cause2* from network *network ID*.

Description: The handler for the CNX YDC ISDN card received a Status Indication for the indicated connection.

ISDN.024

Level: UE-ERROR

Short Syntax: ISDN.024 Start Rq bd st (0x *startstatus*) nt *network ID*

Long Syntax: ISDN.024 The ISDN network interface card rejected a N_START_RQ with the status 0x *startstatus* for network *network ID*.

Description: The ISDN interface card is not in a consistent state with the handler.

Action: Report this event to customer service.

ISDN.025

Level: C-INFO

Short Syntax: ISDN.025 Start ok nt *network ID*

Long Syntax: ISDN.025 Start of the ISDN network interface card succeeded for network *network ID*.

Description: Start of the ISDN smart card completed normally.

ISDN.026

Level: C-INFO

Short Syntax: ISDN.026 Hndlr inidev() st *state* nt *network ID*

Long Syntax: ISDN.026 Handler inidev() from state *state* for network *network ID*.

Description: Initialization of the device interface by the device handler.

ISDN.027

Level: C-INFO

Short Syntax: ISDN.027 Hndlr N_START_RQ nt *network ID*

Long Syntax: ISDN.027 Handler sent N_START_RQ for network *network ID*.

Description: N_START_RQ sent to device interface.

ISDN.028

Level: C-INFO

Short Syntax: ISDN.028 Can't N_START_RQ DCT i/o flg (0x *istatus*:0x *ostatus*) nt *network ID*

Long Syntax: ISDN.028 Either the device status (0x *istatus*:0x *ostatus*) or the lack of a buffer prevented an N_START_RQ to the ISDN CNX YDC port for network *network ID*.

Description: The handler has to send an N_START_RQ to initiate transfers, but can't.

ISDN.029

Level: UE-ERROR

Short Syntax: ISDN.029 Brd Crsh -- rstrng: nt *network ID*

Long Syntax: ISDN.029 Interface software crash, attempting restart nt *network ID*

Description: The ISDN CNX YDC board software has crashed (LOG_EXIT).

Action: Report this event to customer service.

ISDN.030

Level: UE-ERROR

Short Syntax: ISDN.030 Tx Frm too long (*frame* > *configsize*) nt *network ID*

Long Syntax: ISDN.030 The size of the frame (*frame*) passed to the ISDN handler for transmission exceeded the maximum size configured (*configsize* -- less one) net *network ID*

Description: The ISDN CNX YDC card restricts the transmit size to be one less than the maximum configured, and discards any frames that exceed this length. Check the encapsulator packet size. It should be smaller than the ISDN frame size less one and less any encapsulator headers.

ISDN.031

Level: U-INFO

Short Syntax: ISDN.031 Cll rfsd frm *FromAddress*: *FromSubAddress* to *ToAddress*: *ToSubAddress* on nt *network ID*

Long Syntax: ISDN.031 Incoming Call Refused from *FromAddress*: *FromSubAddress* to *ToAddress*: *ToSubAddress* on net *network ID*

Description: An N_CONN_IND was received from the ISDN network, but no registered client chose to accept it.

Action: Check the reported address against those configured. It may be that the remote router's

configuration is in error, or that some device on the ISDN network is calling the wrong number.

ISDN.032

Level: C-INFO

Short Syntax: ISDN.032 Chn *Channel* ConnID 0x *ConnID* Cll Txcmp on nt *network ID*

Long Syntax: ISDN.032 Channel *Channel* ConnID 0x *ConnID*: transfer of N_CONN_RQ to ISDN smart card completed on net *network ID*

Description: A connection has been successfully initiated.

ISDN.033

Level: C-INFO

Short Syntax: ISDN.033 Chn *Channel* ConnID 0x *ConnID* FSM st *state1* ev *event* -> *state2* nt *network ID*

Long Syntax: ISDN.033 Channel *Channel* ConnID 0x *ConnID* FSM transition: old state *state1*, event *event*, new state *state2* on network *network ID*.

Description: An FSM transition occurred.

ISDN.034

Level: U-INFO

Short Syntax: ISDN.034 Chn UN ConnID UNAS callout rfsd (no chnl/destrsp) nt *network ID*

Long Syntax: ISDN.034 Channel (unassigned) ConnID (unassigned) call out refused (no channel available, or destination not responding) on network *network ID*.

Description: Connection setup failed, either because no spare channel was available, or the destination has refused (retry-count) previous calls within the timeout period. In the latter case, a subsequent attempt will proceed once the timeout has expired.

Action: Verify that the address configured for the dial circuits is correct, and that at least one of the two channels (locally and at the destination) is currently unassigned.

ISDN.035

Level: U-INFO

Short Syntax: ISDN.035 Inv Chn (0x *Channel*) ConnID 0x *ConnID* ev *message* nt *network ID*

Long Syntax: ISDN.035 Invalid Channel (0x *Channel*) ConnID 0x *ConnID* in message *message* on network *network ID*.

Description: The channel type in a message received from the interface card was invalid. The message was ignored or rejected.

Action: This may indicate that the ISDN switch to which the interface card is connected is trying to initialize connections on channels that the software cannot recognize. One instance of note may be the unassigned value (reported as 0xFF, but actually 0x0), which, if it persists, may prevent any connections. Contact customer service.

ISDN.036

Level: ALWAYS

Short Syntax: ISDN.036 Bad drct Tx prot *Protocol*, remap to dial circuit on nt *network ID*

Long Syntax: ISDN.036 Some forwarder (*Protocol*) has attempted to transmit directly over the ISDN network *network ID*

Description: Transmits over the ISDN network are only supposed to be done via an associated dial circuit, which will do an appropriate encapsulation. This event was caused by a mistake in the configuration of the forwarders. No forwarder should be configured to use the ISDN network. To bound the number of these messages, they will be logged only a fraction of the actual events.

Cause: A forwarder (IP, IPX, etc) address was assigned to the ISDN interface.

Action: Delete the address, and (probably) re-assign it to a dial circuit (which is itself mapped to the ISDN network).

Cause: The bridge or other forwarder has been configured to use the ISDN interface.

Action: Remove the ISDN interface as a port used by the bridge or forwarder.

ISDN.037

Level: UE-ERROR

Short Syntax: ISDN.037 Stat Rq bd st (0x *startstatus*) nt *network ID*

Long Syntax: ISDN.037 The ISDN network interface card rejected a N_STAT_RQ (parameter download) with the status 0x *startstatus* for network *network ID*.

Description: The ISDN interface card failed to accept the configuration parameters sent down by the router and initialize properly.

Action: Report this event to customer service.

ISDN.038

Level: C-INFO

Short Syntax: ISDN.038 Parameter download ok nt *network ID*

Long Syntax: ISDN.038 Parameter download for the

ISDN network interface card succeeded for network *network ID*.

Description: The ISDN card accepted and initialized correctly with the configuration parameters passed down from the router.

ISDN.039

Level: C-INFO

Short Syntax: ISDN.039 Hndlr N_STAT_RQ nt *network ID*

Long Syntax: ISDN.039 Handler sent N_STAT_RQ for network *network ID*.

Description: N_STAT_RQ sent to device interface.

ISDN.040

Level: C-INFO

Short Syntax: ISDN.040 Can't N_STAT_RQ DCT i/o flg (0x *istatus*:0x *ostatus*) nt *network ID*

Long Syntax: ISDN.040 Either the device status (0x *istatus*:0x *ostatus*) or the lack of a buffer prevented an N_STAT_RQ to the ISDN CNX port for network *network ID*.

Description: The handler has to send an N_STAT_RQ for configuration parameter download, but can't.

ISDN.041

Level: U-INFO

Short Syntax: ISDN.041 Too many non-resp, will try later nt *network ID*

Long Syntax: ISDN.041 Too many non-responses, will try later on network *network ID*.

Description: The destination has refused (retry-count) previous calls within the timeout period. A subsequent attempt will proceed once the timeout has expired.

Action: Verify that the address configured for the dial circuits is correct, and that at least one of the two channels (locally and at the destination) is currently unassigned.

ISDN.042

Level: C-TRACE

Short Syntax: ISDN.042 *event* nt *network ID*

Long Syntax: ISDN.042 *event* on network *network ID*.

Description: Trace of Physical layer events.

ISDN.043

Level: C-TRACE

Short Syntax: ISDN.043 *packet*

Long Syntax: ISDN.043 *packet*.

Description: ISDN packet trace.

Panic isdnym

Short Syntax: YDC ISDN: mem alloc fld

Description: The YDC ISDN network handler failed to allocate sufficient memory during the initialization phase.

Action: Contact customer service.

Chapter 51. Intermediate System-Intermediate System Protocol (ISIS)

This chapter describes Intermediate System-Intermediate System Protocol (ISIS) messages. For information on message content and how to use the message, refer to the Introduction.

ISIS.001

Level: UE-ERROR

Short Syntax: ISIS.001 OSI protocol does not run over *nettype/ n_int*

Long Syntax: ISIS.001 OSI protocol does not run over *nettype/ n_int*

Description: OSI was configured to run over a type of network which currently doesn't support OSI.

ISIS.002

Level: UE-ERROR

Short Syntax: ISIS.002 received incomplete isis pdu

Long Syntax: ISIS.002 received incomplete isis packet

Description: A packet fragment recognized as an ISIS packet was received.

ISIS.003

Level: UE-ERROR

Short Syntax: ISIS.003 received isis pdu with a bad version # = *version_number*

Long Syntax: ISIS.003 received packet with a bad version number, vers = *version_number*

Description: An ESIS packet was received but had a bad or unsupported version number.

ISIS.004

Level: UE-ERROR

Short Syntax: ISIS.004 received isis pdu with a bad id length = *id_length*

Long Syntax: ISIS.004 received packet with a bad ID length = *id_length*

Description: An ISIS packet was dropped because it had a bad ID length.

ISIS.005

Level: P_TRACE

Short Syntax: ISIS.005 *pdu_type* rcvd on int *interface* source id *source_id*

Long Syntax: ISIS.005 *pdu_type* received on interface *interface* source id *source_id*

Description: An ISIS packet was received.

ISIS.006

Level: UE-ERROR

Short Syntax: ISIS.006 received isis pdu *pdu_type* with bad header length = *hdr_length*

Long Syntax: ISIS.006 received packet, type= *pdu_type*, with a bad header length = *hdr_length* bytes

Description: An ISIS packet with a bad header length has been dropped.

ISIS.007

Level: UE-ERROR

Short Syntax: ISIS.007 received pdu *pdu_type* with out of range area address, length = *add_length*

Long Syntax: ISIS.007 received packet, type= *pdu_type*, with an out of range area address length = *add_length*

Description: An IS-IS packet with an out of range area address has been dropped.

ISIS.008

Level: UE-ERROR

Short Syntax: ISIS.008 isis pdu *pdu_type* received with a bad option *opt_code* length = *opt_length*

Long Syntax: ISIS.008 received packet, type= *pdu_type*, with a bad option, code= *opt_code*, length = *opt_length*

Description: An ISIS packet with an unknown PDU type has been dropped.

ISIS.009

Level: UE-ERROR

Short Syntax: ISIS.009 received isis pdu *pdu_type* with invalid option *opt_code*

Long Syntax: ISIS.009 received packet *pdu_type* with an invalid option = *opt_code*

Description: An ISIS packet with an invalid option has been dropped.

ISIS.010

Level: UE-ERROR

Short Syntax: ISIS.010 received isis pdu *pdu_type* with multiple authentication fields

Long Syntax: ISIS.010 received packet, type= *pdu_type*, with multiple authentication fields

Description: An ISIS packet with multiple authentication fields has been dropped.

ISIS.011

Level: UE-ERROR

Short Syntax: ISIS.011 isis pdu *pdu_type* dropped - unsupported password type = *pwd_type*

Long Syntax: ISIS.011 received packet, type= *pdu_type*, with unsupported password type = *pwd_type*

Description: An ISIS packet with an unsupported password type has been dropped.

ISIS.012

Level: UE-ERROR

Short Syntax: ISIS.012 isis pdu *pdu_type* dropped - authentication failure

Long Syntax: ISIS.012 received packet, type= *pdu_type* - authentication failure

Description: An ISIS packet failed authentication, packet dropped.

ISIS.013

Level: UE-ERROR

Short Syntax: ISIS.013 isis pdu *pdu_type* dropped - bad pdu length = *pdu_length*

Long Syntax: ISIS.013 received packet, type= *pdu_type*, with a bad pdu length = *pdu_length* bytes

Description: An ISIS packet with a bad header length has been dropped.

ISIS.014

Level: UE-ERROR

Short Syntax: ISIS.014 isis pdu *pdu_type* dropped - out of order options

Long Syntax: ISIS.014 received packet, type= *pdu_type*, has out of order options

Description: An ISIS packet with out of order options has been dropped.

ISIS.015

Level: UE-ERROR

Short Syntax: ISIS.015 isis pdu *pdu_type* dropped - out of range prefix address, length = *add_length*

Long Syntax: ISIS.015 received packet, type= *pdu_type*, with an out of range prefix address length = *add_length*

Description: An IS-IS packet with an out of range prefix address has been dropped.

ISIS.016

Level: UE_ERROR

Short Syntax: ISIS.016 mismatch between subnet type and net type on *nettype/ n_int*

Long Syntax: ISIS.016 mismatch between subnet type and net type on *nettype/ n_int*

Description: While bringing up a network, an inconsistency between the ISIS subnet type and the network type was discovered.

ISIS.017

Level: UE_ERROR

Short Syntax: ISIS.017 invalid subnet type on *nettype/ n_net*

Long Syntax: ISIS.017 invalid subnet type on *nettype/ n_net*

Description: Couldn't bring up the ISIS subnet due to an invalid subnet type.

ISIS.018

Level: UE_ERROR

Short Syntax: ISIS.018 isis turned off on lan - not started on *nettype/ n_int*

Long Syntax: ISIS.018 ISIS turned off on lan, ISIS not started on *nettype/ n_int*

Description: Couldn't start ISIS on the LAN because ISIS is configured to be off.

ISIS.019

Level: UE_ERROR

Short Syntax: ISIS.019 adjacency not established - no common area

Long Syntax: ISIS.019 Adjacency rejected because it doesn't have a matching area address

Description: The adjacency is rejected because it doesn't have an area address that matches one in the router's set of area addresses.

ISIS.020

Level: UE_ERROR

Short Syntax: ISIS.020 no free IS adjacencies

Long Syntax: ISIS.020 No free IS adjacency structures

Description: Unable to get an IS adjacency structure from the free list.

ISIS.021

Level: UE_ERROR

Short Syntax: ISIS.021 adjacency not established - system type mismatch

Long Syntax: ISIS.021 Adjacency rejected due to a system type mismatch

Description: Adjacency rejected due to a mismatch between the remote system and the router IS type.

ISIS.022

Level: UE_ERROR

Short Syntax: ISIS.022 send of isis pkt failed on *nettype/ n_int*

Long Syntax: ISIS.022 Send of an ISIS packet on *nettype/ n_int* failed

Description: An attempt to send an ISIS packet on the specified interface failed.

ISIS.024

Level: P_TRACE

Short Syntax: ISIS.024 iipph pdu sent on *nettype/ n_int*

Long Syntax: ISIS.024 ISIS point-to-point hello packet sent on *nettype/ n_int*

Description: An ISIS point-to-point packet was successfully transmitted on the specified interface.

ISIS.025

Level: UE_ERROR

Short Syntax: ISIS.025 no memory for lsu

Long Syntax: ISIS.025 No memory available for the link state update

Description: No memory available for the link state update - entering the wait state.

ISIS.026

Level: UE_ERROR

Short Syntax: ISIS.026 isis pdu not prcssd - sbnt not cnfg'd on *nettype/ n_int*

Long Syntax: ISIS.026 ISIS pkt not processed - subnet not configured on *nettype/ n_int*

Description: An ISIS packet was not processed because the subnet was nonexistent or inactive on the interface.

ISIS.027

Level: UE_ERROR

Short Syntax: ISIS.027 isis pdu not processed - pvc not configured

Long Syntax: ISIS.027 ISIS pkt not processed over X25 interface - PVC not configured

Description: ISIS pkt not processed over the specified X25 interface - couldn't find the PVC.

ISIS.028

Level: UE_ERROR

Short Syntax: ISIS.028 isis pdu not processed - isis turned off on *nettype/ n_int*

Long Syntax: ISIS.028 ISIS packet not processed - ISIS turned off on *nettype/ n_int*

Description: An ISIS packet was not processed because ISIS was configured to be off on the specified interface.

ISIS.029

Level: UE_ERROR

Short Syntax: ISIS.029 isis pdu not processed - external domain on *nettype/ n_int*

Long Syntax: ISIS.029 ISIS packet not processed - external domain defined on *nettype/ n_int*

Description: An ISIS packet was not processed because ISIS was configured to be an external domain.

ISIS.030

Level: UE_ERROR

Short Syntax: ISIS.030 L2 PDU dropped (type = *pdu_type*) - IS type is L1 only

Long Syntax: ISIS.030 Level 2 PDU dropped (type = *pdu_type*), IS type is level 1 only

Description: A level 2 ISIS PDU was dropped because this router is configured with an IS type of level 1 only.

ISIS.032

Level: P_TRACE

Short Syntax: ISIS.032 *pdu_type* sent on int *interface* source id *source_id*

Long Syntax: ISIS.032 *pdu_type* sent on interface *interface* source id *source_id*

Description: An ISIS packet was sent.

ISIS.033

Level: UE-ERROR

Short Syntax: ISIS.033 no iob avail to send ISIS packet

Long Syntax: ISIS.033 no i/o buffer available to send isis packet

Description: An attempt to send an ISIS packet failed because of a lack of system i/o buffers.

ISIS.034

Level: P_TRACE

Short Syntax: ISIS.034 LSU queued on circuit *circuit* type *type*

Long Syntax: ISIS.034 A link state update was queued on LAN circuit *circuit* type *type*

Description: A link state update was queued on a LAN circuit do to maximum number of transmission constraints.

ISIS.035

Level: UE_ERROR

Short Syntax: ISIS.035 Transmission failed

Long Syntax: ISIS.035 Transmission failed

Description: The handler returned an error on an attemted transmission.

ISIS.036

Level: UE_ERROR

Short Syntax: ISIS.036 Link State database *type* entering wait state

Long Syntax: ISIS.036 Link State database *type* entering wait state

Description: One of the link state databases entered the waiting state.

ISIS.037

Level: P_TRACE

Short Syntax: ISIS.037 Link State database *type* leaving wait state

Long Syntax: ISIS.037 Link State database *type* leaving wait state

Description: One of the link state databases left the waiting state.

ISIS.038

Level: P_TRACE

Short Syntax: ISIS.038 Dijkstra run on level *type*

Long Syntax: ISIS.038 The decision process (Dijkstra) is being run on level *type*

Description: The decision process (Dijkstra) is being run on one of the levels.

ISIS.040

Level: UE_ERROR

Short Syntax: ISIS.040 Verification of LSP checksum failed, checksum should be *checksum*

Long Syntax: ISIS.040 Verification of LSP checksum failed, checksum should be *checksum*

Description: Verification of a received LSP checksum failed - the user is shown what the checksum should have been.

ISIS.043

Level: U_INFO

Short Syntax: ISIS.043 Level *level* adj with IS *sysid* is now 2-way

Long Syntax: ISIS.043 Level *level* adj with IS *sysid* is now 2-way.

Description: An IS adj has gone from one-way to two-way and is now in the UP state.

ISIS.044

Level: U_INFO

Short Syntax: ISIS.044 Level *level* adj with IS *sysid* gone from two-way to one-way

Long Syntax: ISIS.044 Level *level* adj with IS *sysid* has gone from being two-way to one-way.

Description: An IS adjacency has gone from being two-way to one-way. The router will rerun the designated router election process and rebuild the pseudonode LSU if it is the designated router.

ISIS.045

Level: U_INFO

Short Syntax: ISIS.045 A new level *level* adj with IS *sysid* has been created

Long Syntax: ISIS.045 A new level *level* adj with IS *sysid* has been created.

Description: A new IS adjacency has been established and placed in the initialization state.

ISIS.046

Level: U_INFO

Short Syntax: ISIS.046 This router has been elected as the level *level* DR on circuit *cct*

Long Syntax: ISIS.046 This router has been elected as the level *level* DR on circuit *cct*

Description: This router has been elected designated router on the specified circuit.

ISIS.047

Level: U_INFO

Short Syntax: ISIS.047 This router has resigned as the level *level* DR on circuit *cct*

Long Syntax: ISIS.047 This router has resigned as the level *level* DR on circuit *cct*

Description: This router has resigned as the designated router on the specified circuit.

ISIS.048

Level: U_INFO

Short Syntax: ISIS.048 System *lanid* has been elected as the level *level* DR on circuit *cct*

Long Syntax: ISIS.048 System *lanid* has been elected as the level *level* DR on circuit *cct*.

Description: The specified system has been elected as the designated router on the specified circuit.

ISIS.050

Level: UE_ERROR

Short Syntax: ISIS.050 L1 IS-IS Hello dropped - circuit *cct_id* is L2 only

Long Syntax: ISIS.050 Level 1 IS-IS Hello dropped, circuit *cct_id* is level 2 only

Description: A level 1 ISIS hello packet was dropped because the circuit is configured as level 2 only.

ISIS.051

Level: UE_ERROR

Short Syntax: ISIS.051 LSP dropped - received from non-adjacent system

Long Syntax: ISIS.051 LSP dropped, received from non-adjacent system

Description: A link state packet was dropped because it was received from a system to which no ″up″ adjacency currently exists or an adjacency exists but is the wrong level.

ISIS.052

Level: UE_ERROR

Short Syntax: ISIS.052 SNP dropped - received from non-adjacent system

Long Syntax: ISIS.052 SNP dropped, received from non-adjacent system

Description: A sequence number packet was dropped because it was received from a system to which no ″up″ adjacency currently exists or an adjacency exists but is the wrong level.

ISIS.053

Level: UE_ERROR

Short Syntax: ISIS.053 LSP buffer size (*lspbufsz*) > datalink block size (*datalinkblksz*) on int *interface* net *nettype/ netinstance*

Long Syntax: ISIS.053 LSP buffer size (*lspbufsz*) is greater than the datalink block size (*datalinkblksz*) on cir *interface* net *nettype/ netinstance*

Description: The datalink block size of the circuit is not large enough to accommodate sending ISIS LSPs.

ISIS.054

Level: C_INFO

Short Syntax: ISIS.054 Level *level* PSNP rcvd on ifc *network* dropped - not DR

Long Syntax: ISIS.054 Level *level* Partial Sequence Number PDU received on interface *network* was dropped because this IS is not the designated router.

Description: A partial sequence number PDU was dropped because this intermediate system is not the designated router. Only the designated router processes partial sequence number PDUs.

ISIS.055

Level: UE-ERROR

Short Syntax: ISIS.055 ISIS input que ovflw

Long Syntax: ISIS.055 ISIS input queue overflow

Description: The ISO ISIS input packet queue has overflowed. Packet is dropped.

ISIS.057

Level: UE_ERROR

Short Syntax: ISIS.057 Dropped LAN ISIS Hello pckt rcvd on a PTPT link (*n_int*)

Long Syntax: ISIS.057 Dropped LAN ISIS Hello packet received on point-point link (*n_int*)

Description: The router cannot process a LAN ISIS Hello packet received on a point-to-point link and the forwarder drops the packet.

ISIS.058

Level: UE_ERROR

Short Syntax: ISIS.058 Dropped PTPT ISIS Hello pckt rcvd on a LAN link (*n_int*)

Long Syntax: ISIS.058 Dropped PTPT ISIS Hello packet received on a LAN link (*n_int*)

Description: The router cannot process a point-to-point ISIS Hello packet received on a LAN link and the forwarder drops the packet.

Chapter 52. Internet Control Message Protocol (ICMP)

This chapter describes Internet Control Message Protocol (ICMP) messages. For information on message content and how to use the message, refer to the Introduction.

ICMP.001

Level: UE-ERROR

Short Syntax: ICMP.001 bd cks 0x *received_checksum* (exp 0x *good_checksum*) *source_IP_address -> destination_IP_address*

Long Syntax: ICMP.001 bad ICMP checksum 0x *received_checksum* received (expected 0x *good_checksum*) in packet from *source_IP_address* to *destination_IP_address*

Description: A bad ICMP checksum was detected in an incoming ICMP message. The received checksum is displayed, together with the value that the checkum should have had. The received packet is discarded.

Cause: This is probably caused by an error in the source host.

Action: Contact the manufacturer of the source host and report the problem.

ICMP.002

Level: C-INFO

Short Syntax: ICMP.002 ech *source_IP_address -> destination_IP_address*

Long Syntax: ICMP.002 echo request packet received from *source_IP_address* to *destination_IP_address*

Description: An ICMP Echo Request was received from the source host by the router.

ICMP.003

Level: U-INFO

Short Syntax: ICMP.003 ech rp *source_IP_address -> destination_IP_address*

Long Syntax: ICMP.003 echo reply packet received from *source_IP_address* to *destination_IP_address*

Description: An ICMP Echo Reply was received from the source host by the router. This is a slightly suspicious event, since the router does not normally send ICMP Echo Requests.

ICMP.004

Level: CI-ERROR

Short Syntax: ICMP.004 unhnd typ *ICMP_type ICMP_code source_IP_address -> destination_IP_address*

Long Syntax: ICMP.004 unhandled message type *ICMP_type ICMP_code* from *source_IP_address* to *destination_IP_address*

Description: An ICMP message came in with a type that the router software does not handle.

ICMP.005

Level: U-TRACE

Short Syntax: ICMP.005 unhnd brd typ *ICMP_type ICMP_code source_IP_address -> destination_IP_address*

Long Syntax: ICMP.005 unhandled broadcast message type *ICMP_type ICMP_code* from *source_IP_address* to *destination_IP_address*

Description: A broadcast ICMP message came in with a type that the router software does not handle.

ICMP.006

Level: UE-ERROR

Short Syntax: ICMP.006 bd typ *ICMP_type ICMP_code source_IP_address -> destination_IP_address*

Long Syntax: ICMP.006 bad message type *ICMP_type ICMP_code* from *source_IP_address* to *destination_IP_address*

Description: An ICMP message came in with a type that is not legal.

ICMP.007

Level: C-INFO

Short Syntax: ICMP.007 addr msk *source_IP_address -> destination_IP_address*

Long Syntax: ICMP.007 address mask request received from *source_IP_address* to *destination_IP_address*

Description: An ICMP Address Mask Request was received from the source host by the router.

ICMP.008

Level: C-TRACE

Short Syntax: ICMP.008 addr msk rep *source_IP_address -> destination_IP_address*

Long Syntax: ICMP.008 addres mask reply received from *source_IP_address* to *destination_IP_address*

Description: An ICMP Address Mask Reply was received from the source host by the router.

ICMP.009

Level: UI-ERROR

Short Syntax: ICMP.009 no pkt or mem

Long Syntax: ICMP.009 heap memory or packet buffer not available

Description: Internal resources in the router necessary to reply to the incoming message were unavailable.

Cause: Temporarily heavy traffic, or not enough memory for configuration.

Action: If this message occurs persistently and with other messages that indicate the router is out of memory or buffers, the router may not have enough memory to support this configuration. Display the memory statistics in the gateway console to check the status of heap memory and global buffers. Add more memory, or disable unnecessary forwarders, protocols and networks to reduce demand for memory.

ICMP.010

Level: UE-ERROR

Short Syntax: ICMP.010 amb addr msk *source_IP_address -> destination_IP_address*

Long Syntax: ICMP.010 ambiguous address mask request received from *source_IP_address* to *destination_IP_address*

Description: An incoming address mask request on an interface which contained more than one IP source address contained a destination address which could not be localized to one of the addresses, so no reply could be generated.

ICMP.011

Level: UI-ERROR

Short Syntax: ICMP.011 err *code* sndng pkt to nt *network ID*

Long Syntax: ICMP.011 error *code* sending packet to net *network ID*

Description: An outgoing reply packet was dropped as the result of some problem in the router.

Cause: There are many potential causes of this problem; an overloaded output queue, a down network, etc.

Action: Consult logging output from the relevant network subsystem for more information.

ICMP.012

Level: C-INFO

Short Syntax: ICMP.012 rdr *source_IP_address -> destination_IP_address* to *new_next_hop_IP_address*

Long Syntax: ICMP.012 sending redirect for packet from *source_IP_address* to *destination_IP_address* to use router *new_next_hop_IP_address*

Description: The router is sending an ICMP Redirect, advising a source host on a directly connected network that there is a better first hop router for this traffic.

ICMP.013

Level: U-INFO

Short Syntax: ICMP.013 bd prm off *problem_offset source_IP_address -> destination_IP_address*

Long Syntax: ICMP.013 sending parameter problem message problem offset *problem_offset* for packet from *source_IP_address* to *destination_IP_address*

Description: The router is sending an ICMP Parameter Problem message, for an unspecified problem at the given offset.

ICMP.014

Level: U-TRACE

Short Syntax: ICMP.014 snd *ICMP_type ICMP_code* pkt *source_IP_address -> destination_IP_address*

Long Syntax: ICMP.014 sending packet type *ICMP_type* code *ICMP_code* for packet from *source_IP_address* to *destination_IP_address*

Description: The router is sending an ICMP packet of the specified type about a packet from the source host to the destination.

ICMP.015

Level: UE-ERROR

Short Syntax: ICMP.015 shrt ICMP hdr *header_length* src *source_ip_address*

Long Syntax: ICMP.015 short ICMP packet *header_length* received in packet from *source_ip_address*

Description: This message is generated when an ICMP packet's indicated header length is below the minimum possible length for an ICMP packet.

Cause: Most likely, this is a damaged packet. It may be that another node is building an incorrect header.

Action: If the problem persists, examine a line trace to determine where the packet is being damaged.

ICMP.016

Level: U-TRACE

Short Syntax: ICMP.016 *current_next_hop* rdr dest *IP_destination* to *better_next_hop*

Long Syntax: ICMP.016 *current_next_hop* has redirected traffic for *IP_destination* to *better_next_hop*

Description: A redirect has been received, changing the next hop for the given destination. Redirects are only processed when running in host mode.

ICMP.017

Level: UE-ERROR

Short Syntax: ICMP.017 Bad rdr from *gateway_address*, rsn: *reason*

Long Syntax: ICMP.017 Redirect received from *gateway_address* was bad for the reason: *reason*

Description: A redirect was received from a router, but rejected for the specified reason.

ICMP.018

Level: U-TRACE

Short Syntax: ICMP.018 Router advertisement received from *router_address*

Long Syntax: ICMP.018 Router advertisement received from *router_address*

Description: An ICMP Router Advertisement (Gateway Discovery) message has been received from the specified router.

ICMP.019

Level: UE-ERROR

Short Syntax: ICMP.019 Bad router adv from *gateway_address*, rsn: *reason*

Long Syntax: ICMP.019 Router advertisement received from *gateway_address* was bad for the reason: *reason*

Description: An ICMP Router Advertisement (Gateway Discovery) message has been received from the specified router, but was rejected for the specified reason.

ICMP.020

Level: U-INFO

Short Syntax: ICMP.020 rcvd typ *ICMP_type ICMP_code source_IP_address -> destination_IP_address*

Long Syntax: ICMP.020 received message type *ICMP_type ICMP_code* from *source_IP_address* to *destination_IP_address*

Description: The router has received an ICMP message of the specified type from the source host.

ICMP.021

Level: C-INFO

Short Syntax: ICMP.021 Dropping ech *source_IP_address -> destination_IP_address*

Long Syntax: ICMP.021 Dropping echo request packet received from *source_IP_address* to *destination_IP_address*

Description: An ICMP Echo Request was received from the source host by the router. The router has been configured to drop the request with no response.

Chapter 53. Internet Control Message Protocol for IPv6 (ICM6)

This chapter describes Internet Control Message Protocol for IPv6 (ICM6) messages. For information on message content and how to use the message, refer to the Introduction.

ICM6.001

Level: UE-ERROR

Short Syntax: ICM6.001 Bad checksum 0x *received_checksum* (expected 0x *good_checksum*) *source_IP_address -> destination_IP_address*

Long Syntax: ICM6.001 Bad ICMP6 checksum 0x *received_checksum* received (expected 0x *good_checksum*) in packet from *source_IP_address* to *destination_IP_address*

Description: A bad ICMP6 checksum was detected in an incoming ICMP6 message. The received checksum is displayed, together with the value that the checkum should have had. The received packet is discarded.

Cause: This is probably caused by an error in the source host.

Action: Contact the manufacturer of the source host and report the problem.

ICM6.002

Level: C-INFO

Short Syntax: ICM6.002 Echo request received *source_IP_address -> destination_IP_address*

Long Syntax: ICM6.002 Echo request packet received from *source_IP_address* to *destination_IP_address*

Description: An ICMP6 Echo Request was received from the source host by the router.

ICM6.003

Level: C-INFO

Short Syntax: ICM6.003 Echo reply received *source_IP_address -> destination_IP_address*

Long Syntax: ICM6.003 Echo reply packet received from *source_IP_address* to *destination_IP_address*

Description: An ICMP6 Echo Reply was received from the source host by the router.

ICM6.004

Level: CI-ERROR

Short Syntax: ICM6.004 Msg type not handled *ICMP6_type ICMP6_code source_IP_address -> destination_IP_address*

Long Syntax: ICM6.004 Message type not handled *ICMP6_type ICMP6_code* from *source_IP_address* to *destination_IP_address*

Description: An ICMP6 message came in with a type that the router software does not handle.

ICM6.005

Level: U-TRACE

Short Syntax: ICM6.005 Multicast type not handled *ICMP6_type ICMP6_code source_IP_address -> destination_IP_address*

Long Syntax: ICM6.005 Multicast message type not handled *ICMP6_type ICMP6_code* from *source_IP_address* to *destination_IP_address*

Description: A multicast ICMP6 message came in with a type that the router software does not handle.

ICM6.006

Level: UE-ERROR

Short Syntax: ICM6.006 Bad type *ICMP6_type ICMP6_code source_IP_address -> destination_IP_address*

Long Syntax: ICM6.006 Bad message type *ICMP6_type ICMP6_code* from *source_IP_address* to *destination_IP_address*

Description: An ICMP6 message came in with a type that is not legal.

ICM6.007

Level: U-INFO

Short Syntax: ICM6.007 Received msg type *ICMP6_type ICMP6_code source_IP_address -> destination_IP_address*

Long Syntax: ICM6.007 Received message type *ICMP6_type ICMP6_code* from *source_IP_address* to *destination_IP_address*

Description: The router has received an ICMP6 message of the specified type from the source host.

ICM6.008

Level: C-INFO

Short Syntax: ICM6.008 Dropping echo

source_IP_address -> destination_IP_address

Long Syntax: ICM6.008 Dropping echo request packet received from *source_IP_address* to *destination_IP_address*

Description: An ICMP6 Echo Request was received from the source host by the router. The router has been configured to drop the request with no response.

ICM6.009

Level: UI-ERROR

Short Syntax: ICM6.009 No mem available

Long Syntax: ICM6.009 Heap memory or packet buffer not available

Description: Internal resources in the router necessary to reply to the incoming message were unavailable.

Cause: Temporarily heavy traffic, or not enough memory for configuration.

Action: If this message occurs persistently and with other messages that indicate the router is out of memory or buffers, the router may not have enough memory to support this configuration. Display the memory statistics in the gateway console to check the status of heap memory and global buffers. Add more memory, or disable unnecessary forwarders, protocols and networks to reduce demand for memory.

ICM6.010

Level: UI-ERROR

Short Syntax: ICM6.010 ICMP packet (*sourceaddress*) send *destaddress -> network ID* with rc *reasoncode*

Long Syntax: ICM6.010 Sending an ICMP packet (*sourceaddress*) via direct n_send from *destaddress* to *network ID* failed for reason *reasoncode*

Description: Sending ICMP packet with destination address is a link local addess failed due to some problem in router. The reason_code gives the cause.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

ICM6.011

Level: U-TRACE

Short Syntax: ICM6.011 Received packet too big for *destination_address*, pmtu *mtu*

Long Syntax: ICM6.011 Packet Too Big ICMP message received for *destination_address*, path MTU is *mtu*

Description: A packet too big message has been received for a packet originated by this router. Path MTU Discovery will start for this destination address.

ICM6.012

Level: U-INFO

Short Syntax: ICM6.012 Sending Pkt Too Big, next hop mtu *next_hop_mtu source_IP_address -> destination_IP_address*

Long Syntax: ICM6.012 Sending Packet Too Big message, next_hop_mtu *next_hop_mtu* for packet from *source_IP_address* to *destination_IP_address*

Description: The router is sending an ICMP6 Packet Too Big message because the next hop MTU is less than the packet size.

ICM6.013

Level: U-INFO

Short Syntax: ICM6.013 Sending bad parm msg, offset *problem_offset source_IP_address -> destination_IP_address*

Long Syntax: ICM6.013 Sending parameter problem message - offset *problem_offset* for packet from *source_IP_address* to *destination_IP_address*

Description: The router is sending an ICMP6 Parameter Problem message, for an unspecified problem at the given offset.

ICM6.014

Level: U-TRACE

Short Syntax: ICM6.014 Sending *ICMP6_pkt_type ICMP6_type ICMP6_code* pkt *source_IP_address -> destination_IP_address*

Long Syntax: ICM6.014 Sending (*ICMP6_pkt_type*) packet type *ICMP6_type* code *ICMP6_code* for packet from *source_IP_address* to *destination_IP_address*

Description: The router is sending an ICMP6 packet of the specified type about a packet from the source host to the destination.

ICM6.015

Level: UE-ERROR

Short Syntax: ICM6.015 Short ICMP6 hdr *header_length* src *source_ip_address*

Long Syntax: ICM6.015 Short ICMP6 header *header_length* received in packet from *source_ip_address*

Description: This message is generated when an ICMP6 packet's indicated header length is below the minimum possible length for an ICMP6 packet.

Cause: Most likely, this is a damaged packet. It may be that another node is building an incorrect header.

Action: If the problem persists, examine a line trace to determine where the packet is being damaged.

ICM6.016

Level: U-TRACE

Short Syntax: ICM6.016 Path *destination_address* aged, pmtu *mtu*

Long Syntax: ICM6.016 Path *destination_address* aged out of table, path MTU *mtu*

Description: The path MTU aging timer has expired for the specified destination address. The path MTU will be reset to the output net MTU and path MTU discovery will be started on the next packet to the destination address.

ICM6.017

Level: UE-ERROR

Short Syntax: ICM6.017 Rcvd pkt too big from *destination_address* pmtu *mtu*, pmtu disc disabled

Long Syntax: ICM6.017 Received packet too big message from *destination_address* pmtu *mtu*, path MTU discovery is disabled

Description: Path MTU discovery is disabled, however a Packet Too Big ICMP message has been received. If Path MTU Discovery has recently been disabled, this is not a problem. However, if Path MTU Discovery has been disabled for some time, this ICMP message should not be received, since with Path MTU Discovery disabled, the maximum size packets generated are 1280 bytes. All IPV6 routers should be able to support this packet size.

ICM6.018

Level: UI-ERROR

Short Syntax: ICM6.018 No mem for pmtu disc for *destination_address*

Long Syntax: ICM6.018 There is no memory available to perform Path MTU Discovery for *destination_address*

Description: There is not enough memory in the router to allocate the control blocks necessary for Path MTU Discovery for packets to the specified address.

Chapter 54. Internet Packet Exchange (IPX)

This chapter describes Internet Packet Exchange (IPX) messages. For information on message content and how to use the message, refer to the Introduction.

IPX.002

Level: UI-ERROR

Short Syntax: IPX.002 q ovf *source_net/ source_node -> dest_net/ dest_node*, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.002 Queue overflow, *source_net/ source_node -> dest_net/ dest_node*, circ *IPX_circuit* net *network ID*

Description: IPX forwarder input queue has overflowed.

Cause: More packets are being received than the forwarder can forward.

IPX.003

Level: UE-ERROR

Short Syntax: IPX.003 bad hst chksm frm *source_net/ source_node*

Long Syntax: IPX.003 Bad host checksum from *source_net/ source_node*

Description: This message is generated when a packet arrives for this host with an incorrect checksum.

IPX.004

Level: U-INFO

Short Syntax: IPX.004 err pkt *error_type* frm *source_net/ source_node*

Long Syntax: IPX.004 Error packet, errno *error_type*, received from *source_net/ source_node*

Description: This message is generated when an error packet is received.

IPX.005

Level: U-TRACE

Short Syntax: IPX.005 no hndlr for skt *destination_socket* typ *packet_type* frm *source_net/ source_node*

Long Syntax: IPX.005 No handler for socket *destination_socket* type *packet_type* from *source_net/ source_node*

Description: A packet arrived for an unknown or unsupported socket or type. The packet was a broadcast packet.

IPX.006

Level: UE-ERROR

Short Syntax: IPX.006 no hndlr for skt *destination_socket* typ *packet_type* frm *source_net/ source_node*

Long Syntax: IPX.006 No handler for socket *destination_socket* type *packet_type* from *source_net/ source_node*

Description: A packet arrived for an unknown or unsupported socket or type. The packet was addressed to the router.

IPX.007

Level: UI-ERROR

Short Syntax: IPX.007 no hst addr set for cir *IPX_circuit* nt *network ID*, not enabled

Long Syntax: IPX.007 no host address set for circ *IPX_circuit* net *network ID*, not enabled

Description: The forwarder was bringing up the specified IPX circuit, but no host address was set so the IPX circuit was not enabled.

Cause: IPX circuit enabled on a serial line with no host address.

Action: Set IPX host address.

IPX.008

Level: UE-ERROR

Short Syntax: IPX.008 SAP bad typ *packet_type* frm *source_net/ source_node*

Long Syntax: IPX.008 SAP bad type *packet_type* from *source_net/ source_node*

Description: This message is generated when a packet is received with a bad SAP type.

IPX.009

Level: C-TRACE

Short Syntax: IPX.009 SAP gen rply frm *source_net/ source_node*

Long Syntax: IPX.009 SAP general reply from *source_net/ source_node*

Description: SAP has received a General Reply packet

from the specified host. The data in the packet will be used to update the SAP database.

IPX.010

Level: UI-ERROR

Short Syntax: IPX.010 SAP tbl ovrfl, dsc type *service_type* nm [*service_name*]

Long Syntax: IPX.010 SAP table overflow, discarded type *service_type* name [*service_name*]

Description: A new entry cannot be added to the SAP table because it is full. The new entry is discarded.

Cause: SAP table is smaller than number of services on IPX internet.

Action: Increase the size of the SAP table.

IPX.011

Level: UE-ERROR

Short Syntax: IPX.011 SAP srvc typ *service_type* nm [*service_name*] mvd to *new_net*/ *new_node*

Long Syntax: IPX.011 SAP service type *service_type* name [*service_name*] moved to *new_net*/ *new_node*

Description: A SAP General Reply was received with a different network/address pair than is presently in the SAP database.

Cause: Duplicate name assigned for service.

Action: Eliminate duplicated name.

Cause: Service physically moved faster than SAP timeout.

Action: Do not move services so fast.

IPX.012

Level: U-INFO

Short Syntax: IPX.012 SAP del typ *service_type* nm [*service_name*]

Long Syntax: IPX.012 SAP deleted type *service_type* name [*service_name*]

Description: A SAP table entry has been declared dead. It will be advertised as unreachable for another 60 seconds, and then removed from the SAP table.

Cause: No SAP General Reply has been heard containing data on this service type/name pair in 240 seconds.

Action: None, unless service should be up.

IPX.013

Level: UE-ERROR

Short Syntax: IPX.013 SAP bd nearest qry frm *source_net*/ *source_node* ln *length*

Long Syntax: IPX.013 SAP bad length Nearest Service Query from *source_net*/ *source_node*, len *length*

Description: A SAP Nearest Service Query was received with an illegal length.

Cause: Programming error in remote node.

IPX.014

Level: C-TRACE

Short Syntax: IPX.014 SAP nearest qry frm *source_net*/ *source_node*

Long Syntax: IPX.014 SAP nearest query from *source_net*/ *source_node*

Description: A SAP Nearest Service Query was received from the specified node. It will be answered as appropriate.

IPX.015

Level: C-TRACE

Short Syntax: IPX.015 SAP gen qry frm *source_net*/ *source_node*

Long Syntax: IPX.015 SAP general query from *source_net*/ *source_node*

Description: A SAP General Service Query was received from the specified node. It will be answered as appropriate.

IPX.016

Level: U-TRACE

Short Syntax: IPX.016 SAP qry sent, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.016 SAP General Service Query sent, circ *IPX_circuit* net *network ID*

Description: A SAP General Service Query was sent on the specified IPX circuit. One is sent on an IPX circuit when it comes up.

IPX.017

Level: UI-ERROR

Level: OOM

Short Syntax: IPX.017 No mem fr SAP bcst, cir *IPX_circuit* nt *network ID*, *count* pkts snt

Long Syntax: IPX.017 No memory for SAP General

Service Query or Reply, circ *IPX_circuit* net *network ID*, *count* packets sent

Description: This message is generated when no buffer is available to send a SAP General Service Query or Reply packet. Since a General Service Reply can require multiple packets, the message notes how many packets were sent before they ran out.

IPX.018

Level: C-TRACE

Short Syntax: IPX.018 SAP gen rply sent, cir *IPX_circuit* nt *network ID*, *count* pkts

Long Syntax: IPX.018 SAP General Service Reply sent, circ *IPX_circuit* net *network ID*, *count* packets

Description: A SAP General Service Reply has just been sent on the specified IPX circuit. It took the specified number of packets to send the complete SAP database.

IPX.019

Level: P-TRACE

Short Syntax: IPX.019 NB brd *source_net*/ *source_node* -> *dest_net*/ *dest_node*, cir *IPX_circuit* nt *network ID*, *hop_count* hops

Long Syntax: IPX.019 NETBIOS broadcast *source_net*/ *source_node* -> *dest_net*/ *dest_node*, circ *IPX_circuit* net *network ID*, *hop_count* hops

Description: A NETBIOS emulation multi-network broadcast packet has been received for forwarding to other IPX circuits. The IPX hop count indicates how many routers it has been through.

IPX.020

Level: U-TRACE

Short Syntax: IPX.020 NB too many hops frm *source_net*/ *source_node*, cir *IPX_circuit* nt *network ID*, ign

Long Syntax: IPX.020 NETBIOS too many hops from *source_net*/ *source_node* circ *IPX_circuit* net *network ID*, ignored

Description: A NETBIOS emulation broadcast packet has been through more than 8 routers. It will be dropped.

Cause: Normal looping due to multiple paths from source of broadcast packet.

Action: None. This is a normal consequence of the protocol used.

Cause: IPX NETBIOS traffic trying to go across more than 8 hops (networks) between source and destination.

Action: Reconfigure network.

IPX.021

Level: C-TRACE

Short Syntax: IPX.021 NB frm *source_net*/ *source_node*, cir *IPX_circuit* nt *network ID*, already on *connected_network*, ign

Long Syntax: IPX.021 NETBIOS from *source_net*/ *source_node* circ *IPX_circuit* net *network ID*, already on *connected_network*, ignored

Description: This IPX NETBIOS emulation broadcast packet has already been on one of the directly attached IPX circuits. It will not be forwarded, as that would generate a duplicate.

Cause: Normal side-effect of the protocol used.

IPX.022

Level: UI-ERROR

Level: OOM

Short Syntax: IPX.022 NB frm *source_net*/ *source_node*, no mem to cpy

Long Syntax: IPX.022 NETBIOS from *source_net*/ *source_node*, no memory to copy

Description: No memory available to make working copy of this NETBIOS emulation packet to send it out multiple IPX circuits.

IPX.023

Level: CI-ERROR

Short Syntax: IPX.023 NB frm *source_net*/ *source_node*, non-brd cir *IPX_circuit* nt *network ID* unsupp

Long Syntax: IPX.023 NETBIOS from *source_net*/ *source_node*, non-broadcast circ *IPX_circuit* net *network ID* unsupported

Description: Attempting to send NETBIOS emulation packet on an IPX circuit that does not support broadcast. The packet will not be sent on that network.

IPX.024

Level: UI-ERROR

Short Syntax: IPX.024 NB frm *source_net*/ *source_node*, un-numbrd cir *IPX_circuit* nt *network ID* unsupp

Long Syntax: IPX.024 NETBIOS from *source_net*/ *source_node*, un-numbered circ *IPX_circuit* net *network ID* unsupported

Description: Attempting to send NETBIOS emulation packet on an IPX circuit with no network number. The packet will not be sent on that IPX circuit.

Cause: Serial line network operating without a network number.

Action: If you want to run NETBIOS emulation across a serial line network, it must have a network number.

IPX.025

Level: UI-ERROR

Level: OOM

Short Syntax: IPX.025 NB frm *source_net*/ *source_node*, no buf to cpy

Long Syntax: IPX.025 NETBIOS from *source_net*/ *source_node*, no buffer to copy

Description: No packet buffer available to copy this NETBIOS emulation broadcast packet into in order to send it on an IPX circuit.

IPX.026

Level: UI-ERROR

Short Syntax: IPX.026 NB snd dsc, cir *IPX_circuit* nt *network ID*, rsn *reason_code*

Long Syntax: IPX.026 NETBIOS send discarded, circ *IPX_circuit* net *network ID*, reason *reason_code*

Description: An outgoing NETBIOS emulation broadcast packet was not successfully transmitted for the reason indicated by the error code.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network ID.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

IPX.027

Level: UE-ERROR

Short Syntax: IPX.027 bad RIP typ *RIP_opcode* frm *source_net*/ *source_node*

Long Syntax: IPX.027 Bad RIP type *RIP_opcode* from *source_net*/ *source_node*

Description: RIP packet received which was not a request or response.

Cause: Programming error on remote node.

IPX.028

Level: C-TRACE

Short Syntax: IPX.028 RIP resp frm *source_net*/ *source_node*

Long Syntax: IPX.028 RIP response from *source_net*/ *source_node*

Description: This message is generated when a RIP response packet is received. It will be parsed, and the data incorporated into the routing table.

IPX.029

Level: UE-ERROR

Short Syntax: IPX.029 bad net *network* in RIP frm *source_net*/ *source_node*

Long Syntax: IPX.029 Bad network *network* in RIP from *source_net*/ *source_node*

Description: A RIP response was received with an entry having a network number of 00000000 or FFFFFFFF. That entry will be ignored.

Cause: Programming error on remote node.

IPX.030

Level: UI-ERROR

Short Syntax: IPX.030 net route table ovrfl, dscrd *network*

Long Syntax: IPX.030 Network routing table overflow, discarding *network*

Description: This message is generated when a new network cannot be added to the routing table because it is full. The entry is discarded.

Cause: Routing table too small.

Action: Reconfigure IPX protocol to make routing table larger.

IPX.031

Level: C-INFO

Short Syntax: IPX.031 *type* route to *network* now via *router_net*/ *router_node*, *hop_count* hops

Long Syntax: IPX.031 *type* route to network *network* now via *router_net*/ *router_node*, *hop_count* hops

Description: This message is generated when the route to a network changes. The specified router_net/router_node is now the best route to this network, with the noted number of hops. The type of the new route is reported as well (RIP or STATIC).

Cause: Newly reachable network (if preceeded by message IPX.055).

Cause: Change in network topology causes best route

to a network to change. This can happen when networks come up, or go down.

Action: Determine what changes in network topology occurred.

IPX.032

Level: U-INFO

Short Syntax: IPX.032 RIP route to *network* aged away

Long Syntax: IPX.032 RIP route to *network* aged away

Description: This message is generated when a network is declared unreachable because no routing updates have been heard for it in 240 seconds. It will be advertised as unreachable for another 60 seconds, and then deleted from the routing table.

Cause: Intervening router that was advertising this network crashed.

IPX.033

Level: C-TRACE

Short Syntax: IPX.033 Rspnd to RIP rqst frm *source_net/ source_node*

Long Syntax: IPX.033 Responding to RIP Request from *source_net/ source_node*

Description: This message is generated when a RIP Request packet is being parsed for a Reply.

IPX.034

Level: UE-ERROR

Short Syntax: IPX.034 RIP rqst frm *source_net/ source_node* shrt, ln *packet_length*

Long Syntax: IPX.034 RIP Request from *source_net/ source_node* too short, len *packet_length*

Description: A RIP request packet was received which is too short to contain one RIP entry. It will be discarded.

Cause: Programming error on remote node.

IPX.035

Level: U-TRACE

Short Syntax: IPX.035 RIP qry sent, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.035 RIP Query sent, circ *IPX_circuit* net *network ID*

Description: A RIP Query has been sent on the specified IPX circuit. A Query is sent on each IPX circuit when it comes up.

IPX.036

Level: UI-ERROR

Level: OOM

Short Syntax: IPX.036 No mem for RIP pkt, cir *IPX_circuit* nt *network ID*, *packet_count* pkts snt

Long Syntax: IPX.036 No memory for RIP packet, circ *IPX_circuit* net *network ID*, *packet_count* packets sent

Description: This message is generated when no buffer is available to send a RIP Query or Response packet.

IPX.037

Level: C-TRACE

Short Syntax: IPX.037 RIP resp sent, cir *IPX_circuit* nt *network ID*, *packet_count* pkts

Long Syntax: IPX.037 RIP Response sent circ *IPX_circuit* net *network ID*, *packet_count* packets

Description: This message is generated when a RIP Response is sent. The response was sent in the specified number of packets.

IPX.038

Level: U-TRACE

Short Syntax: IPX.038 *source_net/ source_node* -> *dest_net/ dest_node* ign

Long Syntax: IPX.038 Packet from *source_net/ source_node* for *dest_net/ dest_node* ignored

Description: This message is generated when an IPX packet arrives on a network and the IPX forwarder is not active on that network.

IPX.039

Level: C-TRACE

Short Syntax: IPX.039 RIP delta resp sent, cir *IPX_circuit* nt *network ID*, *packet_count* pkts

Long Syntax: IPX.039 RIP delta Response sent circ *IPX_circuit* net *network ID*, *packet_count* packets

Description: This message is generated when a RIP delta Response is sent. This response only includes those networks whose data changed in the last update period. The response was sent in the specified number of packets.

IPX.040

Level: UI-ERROR

Short Syntax: IPX.040 RIP resp snd dsc, cir *IPX_circuit* nt *network ID*, rsn *reason_code*

Long Syntax: IPX.040 RIP Response send discarded,

circ *IPX_circuit* net *network ID*, reason *reason_code*

Description: An outgoing RIP response packet was not successfully transmitted for the reason indicated by the error code.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network ID.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

IPX.041

Level: UI-ERROR

Short Syntax: IPX.041 RIP query snd dsc, cir *IPX_circuit* nt *network ID*, rsn *reason_code*

Long Syntax: IPX.041 RIP Query send discarded, circ *IPX_circuit* net *network ID*, reason *reason_code*

Description: An outgoing RIP query packet was not successfully transmitted for the reason indicated by the error code.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network ID.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

IPX.042

Level: C-TRACE

Short Syntax: IPX.042 SAP GNS reply typ *service_type* nm [*service_name*] to *source_net*/ *source_node*, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.042 SAP giving Nearest Server Reply of type *service_type* name [*service_name*] to *source_net*/ *source_node*, circ *IPX_circuit* net *network ID*

Description: A SAP Nearest Service Reply will be sent to the specified node. The service_type is the type of service, and the service_name is the name of the service.

Cause: Node sent Nearest Server Request, and the nearest server of that type is reachable through this router.

IPX.043

Level: C-TRACE

Short Syntax: IPX.043 SAP nearest qry for typ *service_type* frm *source_net*/ *source_node*, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.043 SAP Nearest Query for service type *service_type* from *source_net*/ *source_node*, circ *IPX_circuit* net *network ID*

Description: A SAP Nearest Service Query was received from the specified node via the specified IPX circuit. If this router is the best route to the closest server of the specified service_type, this router will answer.

Cause: New IPX node booting on an attached LAN, looking for a first file server (service_type of 4).

Cause: Node attempting to locate a particular server by service type, such as a communications server or database server.

IPX.044

Level: C-TRACE

Short Syntax: IPX.044 SAP delta gen rply, cir *IPX_circuit* nt *network ID*, *count* pkts

Long Syntax: IPX.044 SAP delta General Service Reply sent, circ *IPX_circuit* net *network ID*, *count* packets

Description: A SAP delta General Service Reply has just been sent on the specified IPX circuit. This Reply only includes those services whose data changed in the last update period. It took the specified number of packets to send the changes in SAP database.

IPX.045

Level: U-INFO

Short Syntax: IPX.045 *type* new serv typ *service_type* nm [*service_name*] via *via_net*/ *via_node*, *hop_count* hops, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.045 *type* new service route to service type *service_type* name [*service_name*] via *via_net*/

via_node, *hop_count* hops, circ *IPX_circuit* net *network ID*

Description: This message is generated when a new service is added to the SAP table The specified via_net/via_node is the route to this service, with the noted number of hops.

Cause: New service started on IPX internetwork.

Cause: Existing service becomes reachable, due to change in network connectivity.

IPX.046

Level: U-TRACE

Short Syntax: IPX.046 SAP nearest qry frm *source_net/ source_node* ignored, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.046 SAP Nearest Query from *source_net/ source_node* ignored, circ *IPX_circuit* net *network ID*

Description: A SAP Nearest Service Query was received from the specified node via the specified IPX circuit, but processing of these packets has been administratively disabled on this network. The query will be ignored.

Cause: User has used IPX Config command DISABLE REPLY-TO-GET-NEAREST-SERVER.

Action: If this is the desired action, none. To enable response (the default), use the IPX Config command ENABLE REPLY-TO-GET-NEAREST-SERVER.

IPX.047

Level: UI-ERROR

Short Syntax: IPX.047 SAP query snd dsc, cir *IPX_circuit* nt *network ID*, rsn *reason_code*

Long Syntax: IPX.047 SAP Query send discarded, circ *IPX_circuit* net *network ID*, reason *reason_code*

Description: An outgoing SAP query packet was not successfully transmitted for the reason indicated by the error code.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network ID.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

IPX.048

Level: UI-ERROR

Short Syntax: IPX.048 SAP resp snd dsc, cir *IPX_circuit* nt *network ID*, rsn *reason_code*

Long Syntax: IPX.048 SAP Response send discarded, circ *IPX_circuit* net *network ID*, reason *reason_code*

Description: An outgoing SAP response packet was not successfully transmitted for the reason indicated by the error code.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network ID.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

IPX.049

Level: U-TRACE

Short Syntax: IPX.049 SAP no serv typ *service_type* for *source_net/ source_node*, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.049 SAP no server of type *service_type* for Query from *source_net/ source_node*, circ *IPX_circuit* net *network ID*

Description: A SAP Nearest Service Query was received from source_net/source_node, but the SAP database has no service registered of the desired service_type. No response will be sent.

Cause: Service of desired service_type is down or unreachable.

Action: Find out why service is down or unreachable.

Cause: Workstation looking for non-existent service_type.

Action: Correct query on workstation.

IPX.050

Level: U-INFO

Short Syntax: IPX.050 SAP dead serv typ *service_type* nm [*service_name*] from *via_net*/ *via_node*, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.050 SAP dead service route to service type *service_type* name [*service_name*] reported by *via_net*/ *via_node* has become unreachable, circ *IPX_circuit* net *network ID*

Description: This message is generated when a previously reachable service becomes unreachable, and is marked as Dead in the SAP table. The specified via_net/via_node is the server or router that announced the service as being unreachable.

Cause: Server administratively disabled, as with :DOWN command.

Action: None.

Cause: Server crashed.

Action: Find out why server crashed.

Cause: Network on which service is provided has become unreachable.

Action: Use IPX console DUMP command to see if network is reachable.

IPX.051

Level: U-INFO

Short Syntax: IPX.051 RIP route died to *network* from *router_net*/ *router_node*

Long Syntax: IPX.051 RIP route died to network *network* from router *router_net*/ *router_node*

Description: This message is generated when a previously reachable network becomes unreachable, and is marked as Dead in the RIP table. The specified router_net/router_node is the router that announced the network as being unreachable.

Cause: Remote network, or intervening network, went down.

Action: Find out why remote network went down.

Cause: Network is on router that went down.

Action: Find out why remote router went down.

Cause: Network is connected via File Server that was administratively taken down.

IPX.052

Level: UE-ERROR

Short Syntax: IPX.052 RIP resp frm wrong net *source_net*/ *source_node* not *local_net*, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.052 RIP response from wrong network *source_net*/ *source_node* not local network *local_net*, circ *IPX_circuit* net *network ID*

Description: This message is generated when a RIP response packet is received with a source network number that is not the same as the network number of this IPX circuit. The packet will be ignored.

Cause: Misconfiguration of router on this network.

Action: Correct configuration.

IPX.053

Level: UE-ERROR

Short Syntax: IPX.053 SAP resp frm wrong net *source_net*/ *source_node* not *local_net*, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.053 SAP response from wrong network *source_net*/ *source_node* not local network *local_net*, circ *IPX_circuit* net *network ID*

Description: This message is generated when a SAP response packet is received with a source network number that is not the same as the network number of this IPX circuit. The packet will be ignored.

Cause: Misconfiguration of router on this network.

Action: Correct configuration.

IPX.054

Level: C-INFO

Short Syntax: IPX.054 *type* serv typ *service_type* nm [*service_name*] now via *via_net*/ *via_node*, *hop_count* hops, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.054 *type* service route to service type *service_type* name [*service_name*] is now via *via_net*/ *via_node*, *hop_count* hops, circ *IPX_circuit* net *network ID*

Description: This message is generated when the route to a service in the SAP table changes. The specified via_net/via_node is the new route to this service, with the noted number of hops. The type of the route (RIP or STATIC) is also reported.

Cause: Newly reachable service (if proceeded by IPX.045).

Cause: Change in network topology causes best route to a service to change. This can happen when new networks come up, or go down.

Action: Determine what changes in network topology occurred.

IPX.055

Level: U-INFO

Short Syntax: IPX.055 new *network* net *router_net* via *router_node/ hop_count*, hops

Long Syntax: IPX.055 New *network* network number *router_net* via *router_node/ hop_count*, hops

Description: This message is generated when a new network is added to the RIP routing table. The new network was advertised by router_net/router_node, which is now the route to this network, with the noted number of hops.

IPX.056

Level: U-TRACE

Short Syntax: IPX.056 RIP route to *network* garbage coll

Long Syntax: IPX.056 RIP route to *network* garbage collected

Description: This message is generated when a network is removed from the RIP routing table because no routing updates have been heard for it in 300 seconds. This normally happens 60 seconds after an IPX.012 message on the same service.

Cause: Intervening router that was advertising this network went down.

IPX.057

Level: U-INFO

Short Syntax: IPX.057 SAP del typ *service_type* nm [*service_name*], cir *IPX_circuit* nt *network ID* down

Long Syntax: IPX.057 SAP deleted type *service_type* name [*service_name*], circ *IPX_circuit* net *network ID* down

Description: The specified IPX circuit has gone down, and this SAP service having a first hop on that IPX circuit will be placed in the dead state. It will be advertised as unreachable for another 60 seconds, and then removed from the SAP table. However, if there are alternate routes to the same service, they will be learned about within 60 seconds.

Cause: The network via which we reached this service went down.

Action: Bring up network.

IPX.058

Level: U-TRACE

Short Syntax: IPX.058 SAP typ *service_type* nm [*service_name*] garbage coll

Long Syntax: IPX.058 SAP type *service_type* name [*service_name*] garbage collected

Description: This message is generated when a network is removed from the SAP routing table because no SAP responses have been heard for it in 300 seconds.

Cause: Intervening router that was advertising this service went down.

IPX.059

Level: CE-ERROR

Short Syntax: IPX.059 SAP unreach serv typ *service_type* nm [*service_name*] at *service_net/ service_node* from *via_net/ via_node*, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.059 SAP unreachable service type *service_type* name [*service_name*] at *service_net/ service_node* from *via_net/ via_node*, circ *IPX_circuit* net *network ID*

Description: This message is generated when an advertisement for a service is received, but that service in on an IPX network (service_net) that this router has no route to. This advertisement will be ignored.

Cause: Configuration error on node service_net/service_node.

Action: Correct configuration error.

Cause: Service information for some new service has propogated faster than the associated routing information.

Action: None needed if service_net becomes reachable shortly, and this message does not repeat.

IPX.065

Level: U-INFO

Short Syntax: IPX.065 routing cache cleared

Long Syntax: IPX.065 routing cache cleared

Description: The IPX routing cache has been cleared, probably as the result of a routing table change.

IPX.066

Level: U-INFO

Short Syntax: IPX.066 routing cache garbage collecting...

Long Syntax: IPX.066 routing cache garbage collecting...

Description: The IPX routing cache is collecting nonsense data. This takes several passes, and is only done when the cache starts overflowing.

IPX.068

Level: UI-ERROR

Short Syntax: IPX.068 no memory left for IPX local network/node cache entries

Long Syntax: IPX.068 no memory left for IPX local network/node cache entries

Description: The IPX routing local network/node cache needs memory before it can add a new local network and its table into the IPX cache.

IPX.070

Level: UI-ERROR

Short Syntax: IPX.070 rte ovrfl, dst *destination_net*

Long Syntax: IPX.070 route overflow, destination *destination_net*

Description: This message is generated when a new alternate entry cannot be made to routing table because alternate entry space for a given route is already full.

Cause: Maximum routes per destination network is too small.

Action: Increase maximum routing entries per destination network for this protocol.

IPX.072

Level: UI-ERROR

Short Syntax: IPX.072 Error building IPXWAN *iw_pkttype* on cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.072 Error building IPXWAN *iw_pkttype* on circ *IPX_circuit* net *network ID*

Description: An IPXWAN Response is built from a Request. An attempt has been made to build the response without a request.

IPX.073

Level: UI-ERROR

Short Syntax: IPX.073 Name and Node ID must be config'd before IPXWAN can operate

Long Syntax: IPX.073 Router name and Node ID must be configured before IPXWAN can operate

Description: The IPX configuration parameters Name and Node ID must be configured before IPXWAN can operate on any IPX circuit.

IPX.076

Level: UE-ERROR

Short Syntax: IPX.076 IPXWAN *iw_pkttype* pkt dropped, rcv'd on cir *IPX_circuit* nt *network ID*, unsupported int type

Long Syntax: IPX.076 IPXWAN *iw_pkttype* packet dropeed because it was received on an unsupported interface type, circ *IPX_circuit* net *network ID*

Description: An IPXWAN packet was dropped because it was received on an unsupported interface type.

IPX.077

Level: UI-ERROR

Short Syntax: IPX.077 IPXWAN *iw_pkttype* pkt dropped, rcv'd on disabled cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.077 IPXWAN *iw_pkttype* packet dropped - it was received on circ *IPX_circuit* net *network ID* which is disabled for IPXWAN traffic

Description: An IPXWAN packet was dropped because it was received on an IPX circuit on which IPXWAN is disabled.

IPX.078

Level: UE-ERROR

Short Syntax: IPX.078 IPXWAN *iw_pkttype* pkt rejected on cir *IPX_circuit* nt *network ID*, confid id chck failed

Long Syntax: IPX.078 IPXWAN *iw_pkttype* packet received on circ *IPX_circuit* net *network ID* was rejected due to the confidence ID check failing

Description: An IPXWAN packet was rejected because the confidence ID check failed.

IPX.079

Level: UI-ERROR

Short Syntax: IPX.079 IPXWAN *iw_pkttype* pkt rejected on cir *IPX_circuit* nt *network ID*, non-unique node id

Long Syntax: IPX.079 IPXWAN *iw_pkttype* packet received on circ *IPX_circuit* net *network ID* was rejected because its node id is identical to the local node id

Description: An IPXWAN packet was rejected because the node id reported in it was identical to the local node id.

Action: Reconfigure the local IPX node ID with a unique value.

IPX.080

Level: UI-ERROR

Short Syntax: IPX.080 No memory to build IPXWAN packet

Long Syntax: IPX.080 Not able to get a buffer to build an IPXWAN packet

Description: An attempt to get a buffer to build an IPXWAN packet failed.

IPX.081

Level: UI-ERROR

Short Syntax: IPX.081 Failed to send an IPXWAN *iw_pkttype* pkt on cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.081 An attempt to send an IPXWAN *iw_pkttype* packet on circ *IPX_circuit* net *network ID* failed

Description: An attempt to send an IPXWAN packet failed.

IPX.082

Level: UI-ERROR

Short Syntax: IPX.082 IPXWAN *iw_pkttype,* pkt rejected on cir *IPX_circuit* nt *network ID*, seq num mismatch

Long Syntax: IPX.082 IPXWAN *iw_pkttype,* packet received on circ *IPX_circuit* net *network ID* was rejected due to a sequence number mismatch

Description: An IPXWAN packet was dropped due to a sequence number mismatch.

IPX.083

Level: UE-ERROR

Short Syntax: IPX.083 IPXWAN *iw_pkttype* rejected on cir *IPX_circuit* nt *network ID - opt_type* opt not accepted

Long Syntax: IPX.083 IPXWAN *iw_pkttype* rejected on circ *IPX_circuit* net *network ID - opt_type* option not accepted

Description: An IPXWAN packet was rejected because an option was not accepted by the other side of the link.

IPX.084

Level: U-INFO

Short Syntax: IPX.084 IPXWAN connection to be retried on cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.084 IPXWAN connection to be retried on circ *IPX_circuit* net *network ID*

Description: A previously timed-out IPXWAN connection is to be retried.

IPX.085

Level: U-INFO

Short Syntax: IPX.085 IPXWAN connection on cir *IPX_circuit* nt *network ID* timed-out

Long Syntax: IPX.085 IPXWAN connection on circ *IPX_circuit* net *network ID* timed-out

Description: An IPXWAN connection attempt timed-out.

IPX.086

Level: C-INFO

Short Syntax: IPX.086 IPXWAN *iw_pkttype* pkt rcvd on cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.086 IPXWAN *iw_pkttype* packet received on circ *IPX_circuit* net *network ID*

Description: An IPXWAN packet was successfully received, accepted, and processed.

IPX.087

Level: C-INFO

Short Syntax: IPX.087 IPXWAN *iw_pkttype* pkt sent on cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.087 IPXWAN *iw_pkttype* packet sent on circ *IPX_circuit* net *network ID*

Description: An IPXWAN packet was successfully sent on the given IPX circuit.

IPX.088

Level: C-INFO

Short Syntax: IPX.088 IPXWAN connection up on cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.088 IPXWAN connection has come up on circ *IPX_circuit* net *network ID*

Description: An IPXWAN connection is up on the given IPX circuit.

IPX.089

Level: U-INFO

Short Syntax: IPX.089 IPXWAN connection down on cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.089 IPXWAN connection has gone down on circ *IPX_circuit* net *network ID*

Description: An IPXWAN connection has gone down the given IPX circuit. This can happen if the link goes down, if the protocol goes down on the link (IPXCP goes down) or if a Timer Request packet is received.

IPX.090

Level: U-TRACE

Short Syntax: IPX.090 SAP no server reply required for typ *service_type* for *source_net/ source_node*, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.090 SAP no server reply required for type *service_type* for Query from *source_net/ source_node*,

circ *IPX_circuit* net *network ID*

Description: The router received a SAP Nearest Service Query from source_net/source_node. The SAP database indicates that a server exists on the same network as the client. The server will be allowed to respond for itself.

Cause: There is an eligible server on the client's network that is capable of replying for itself.

Action: No action is required.

IPX.091

Level: UI-ERROR

Short Syntax: IPX.091 short NB frm *source_net/ source_node*, cir *IPX_circuit* nt *network ID*, ign

Long Syntax: IPX.091 short NETBIOS frame from *source_net/ source_node* circ *IPX_circuit* net *network ID*, ignored

Description: A NETBIOS type 20 packet must be at least 62 bytes in length. The forwarder drops the packet.

Cause: Unknown.

Action: None. Fix Novell application that is sending the packet.

IPX.092

Level: U-INFO

Short Syntax: IPX.092 Add kpalv proxy *source_net/ source_node. source_socket <-> dest_net/ dest_node. dest_socket*

Long Syntax: IPX.092 Add KeepAlive proxy connection *source_net/ source_node. source_socket <-> dest_net/ dest_node. dest_socket*

Description: A new pair of stations have been added to the proxy keepalive table.

IPX.093

Level: U-INFO

Short Syntax: IPX.093 Del kpalv proxy *source_net/ source_node. source_socket <-> dest_net/ dest_node. dest_socket*

Long Syntax: IPX.093 Delete KeepAlive proxy connection *source_net/ source_node. source_socket <-> dest_net/ dest_node. dest_socket*

Description: A pair of stations have been removed from the proxy keepalive table.

IPX.094

Level: UI-ERROR

Short Syntax: IPX.094 no memory for IPX kpalv proxy tbl

Long Syntax: IPX.094 no memory for IPX KeepAlive Proxy connection table

Description: The IPX Keepalive Proxy initialization routine was unable to allocate memory for its connection table (IPX KeepAlive Proxy feature will not be enabled).

IPX.095

Level: U-INFO

Short Syntax: IPX.095 Serial pkt dropped *source_net/ source_node -> dest_net/ dest_node*, filt cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.095 Serial packet dropped *source_net/ source_node -> dest_net/ dest_node*, filtered circ *IPX_circuit* net *network ID*

Description: A Serialization packet was dropped because the output IPX circuit has IPX KeepAlive filtering enabled.

IPX.096

Level: UI-ERROR

Level: OOM

Short Syntax: IPX.096 No mem fr prxy kpalv req/rsp

Long Syntax: IPX.096 No memory for proxy keepalive request or response

Description: This message is generated when no buffer is available to send an IPX keepalive message.

IPX.097

Level: U-INFO

Short Syntax: IPX.097 Proxy kpalv *type source_net/ source_node. source_socket -> dest_net/ dest_node. dest_socket*

Long Syntax: IPX.097 Sent proxy keepalive *type source_net/ source_node. source_socket -> dest_net/ dest_node. dest_socket*

Description: This message is generated when a proxy keepalive packet is sent.

IPX.098

Level: P-TRACE

Short Syntax: IPX.098 RIP RTR flt PASS pkt frm *source_net/ source_node*, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.098 RIP Router filter PASS packet

from *source_net/ source_node*, circ *IPX_circuit* net *network ID*

Description: A RIP packet has successfully PASSED through the RIP Router filter on the given IPX circuit. The RIP Router filter is based upon the source_node in the IPX packet header, which is displayed by the message.

IPX.099

Level: P-TRACE

Short Syntax: IPX.099 RIP RTR flt DROP pkt frm *source_net/ source_node*, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.099 RIP Router filter DROP packet from *source_net/ source_node*, circ *IPX_circuit* net *network ID*

Description: A RIP packet has been DROPPED by the RIP Router filter on the given IPX circuit. The RIP Router filter is based upon the source_node in the IPX packet header, which is displayed by the message.

IPX.100

Level: P-TRACE

Short Syntax: IPX.100 RIP *iodir* flt PASS IPX net *ipx_network* , cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.100 RIP *iodir* filter PASS IPX network number *ipx_network*, circ *IPX_circuit* net *network ID*

Description: A RIP routing information entry for the given IPX network number has successfully PASSED through the RIP filter on the given IPX circuit in the stated filtering direction, which is either inbound or outbound.

IPX.101

Level: P-TRACE

Short Syntax: IPX.101 RIP *iodir* flt DROP net *ipx_network* , cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.101 RIP *iodir* filter DROP network *ipx_network*, circ *IPX_circuit* net *network ID*

Description: A RIP routing information entry for the given IPX network number has been DROPPED by the RIP filter on the given IPX circuit in the stated filtering direction, which is either inbound or outbound.

IPX.102

Level: P-TRACE

Short Syntax: IPX.102 SAP *iodir* flt PASS typ *service_type* nm [*service_name*], *service_hops* hops, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.102 SAP *iodir* filter PASS type *service_type* name [*service_name*], hops *service_hops*, circ *IPX_circuit* net *network ID*

Description: A SAP routing information entry for the given IPX network number has successfully PASSED through the SAP filter on the given IPX circuit in the stated filtering direction, which is either inbound or outbound.

IPX.103

Level: P-TRACE

Short Syntax: IPX.103 SAP *iodir* flt DROP typ *service_type* nm [*service_name*], *service_hops* hops, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.103 SAP *iodir* filter DROP type *service_type* name [*service_name*], hops *service_hops*, circ *IPX_circuit* net *network ID*

Description: A SAP routing information entry for the given IPX network number has been DROPPED by the SAP filter on the given IPX circuit in the stated filtering direction, which is either inbound or outbound.

IPX.104

Level: P-TRACE

Short Syntax: IPX.104 IPX *iodir* flt PASS typ *packet_type source_net/ source_node. source_socket -> dest_net/ dest_node. dest_socket*, *service_hops* hops, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.104 IPX *iodir* filter PASS type *packet_type*, *source_net/ source_node. source_socket -> dest_net/ dest_node. dest_socket*, *service_hops* hops, circ *IPX_circuit* net *network ID*

Description: An IPX packet has PASSED through the IPX filter on the given IPX circuit in the stated filtering direction, which is either inbound or outbound.

IPX.105

Level: P-TRACE

Short Syntax: IPX.105 IPX *iodir* flt DROP typ *packet_type source_net/ source_node. source_socket -> dest_net/ dest_node. dest_socket*, *service_hops* hops, cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.105 IPX *iodir* filter DROP type *packet_type*, *source_net/ source_node. source_socket -> dest_net/ dest_node. dest_socket*, *service_hops* hops, circ *IPX_circuit* net *network ID*

Description: An IPX packet has been DROPPED by the IPX filter on the given IPX circuit in the stated filtering direction, which is either inbound or outbound.

IPX.106

Level: U-INFO

Short Syntax: IPX.106 rcvd ping *packet_type* pkt *source_net*/ *source_node* -> *destination_net*/ *destination_node*

Long Syntax: IPX.106 received IPXPING *packet_type* packet from *source_net*/ *source_node* to *destination_net*/ *destination_node*

Description: This message is generated when an IPXPING packet is received.

IPX.107

Level: UI-ERROR

Short Syntax: IPX.107 IPXWAN *iw_pkttype* pkt rcvd on cir *IPX_circuit* nt *network ID* has common net zero

Long Syntax: IPX.107 IPXWAN *iw_pkttype* packet received on circ *IPX_circuit* net *network ID* has a common network number of zero

Description: An IPXWAN packet was received indicating that the common network number assigned by the link master is zero. However, the network number must be nonzero since numbered RIP is the negotiated routing type to be used on this link. This can happen when the link master prefers unnumbered RIP and even though it will accept numbered RIP, it does not know how to assign a common network number.

Action: Reconfigure the local IPX node id to be greater than the remote IPX node id to guarantee that this router is the link master and assigns the common network number.

IPX.108

Level: UI-ERROR

Short Syntax: IPX.108 IPXWAN circ *IPX_circuit* configd on ifc *IPX_interface_number*, invalid IPX network number

Long Syntax: IPX.108 IPXWAN circuit *IPX_circuit* attempting to be configured on interface *IPX_interface_number*, invalid IPX network number

Description: IPX was not enabled on the specified IPX circut because the IPX network number had an invalid value

Cause: An invalid IPX network number was configured.

Action: Configure a valid IPX network number on the IPX circuit.

IPX.109

Level: UI-ERROR

Short Syntax: IPX.109 IPXWAN stat rte for cir *IPX_circuit* nt *network ID*, but stat rte glob disabled

Long Syntax: IPX.109 IPXWAN static routing enabled on circ *IPX_circuit* net *network ID*, but static routes and services are globally disabled

Description: Static routes and static services are globally disabled, however the IPXWAN routing type is set to static on the specified IPX circuit.

Cause: Configuration error

Action: Either globally enable static routes and static services or configure IPXWAN to use a routing type other than static.

IPX.110

Level: UI-ERROR

Short Syntax: IPX.110 stat rte invalid on cir *IPX_circuit* nt *network ID*

Long Syntax: IPX.110 Static route cannot be configured on circ *IPX_circuit* net *network ID*

Description: Static route cannot be configured on this type of interface.

Cause: Configuration error

Action: Do not configure static routes on this type of interface.

IPX.111

Level: UI-ERROR

Short Syntax: IPX.111 no memory left for *ipx_structure*

Long Syntax: IPX.111 no memory left for *ipx_structure*

Description: There was not enough memory available to allocate the indicated IPX structure. The IPX component requiring this structure will not be enabled.

IPX.112

Level: C_INFO

Short Syntax: IPX.112 use IPX reset cmd to activate or reset *ipx_component*

Long Syntax: IPX.112 use IPX reset cmd to activate or reset *ipx_component*

Description: The indicated IPX component must be reset using the IPX reset command in order to activate configuration changes.

Cause: The indicated IPX component was configured on an IPX circuit which was either activated or reset.

Action: Use the IPX reset command to activate or reset

any configuration changes made to the indicated IPX component.

IPX.113

Level: UI-ERROR

Short Syntax: IPX.113 IPX cir *IPX_circuit* configd on ifc *IPX_interface_number*, ifc > nnets

Long Syntax: IPX.113 IPX circ *IPX_circuit* configured on interface *IPX_interface_number*, interface is greater than the maximun number of interfaces

Description: The interface that the circuit is suppose to be configure on is invalid. It is greater than the maximum interface number.

Cause: I have no idea how this could happen

Action: Reconfigure the circuit with a valid interface number

---

IPX.114

Level: UI-ERROR

Short Syntax: IPX.114 IPX cir *IPX_circuit* configd on ifc *IPX_interface_number*, ifc type not supported

Long Syntax: IPX.114 IPX circ *IPX_circuit* configured on interface *IPX_interface_number*, interface type is not supported

Description: The interface that the circuit is suppose to be configured on is of a type that does not support IPX.

Cause: After configuring a circuit on an interface, the type of interface was changed to one that does not support IPX

Action: Reconfigure the the circuit onto an interface that does support IPX or reconfigure the interface type to one that supports IPX.

IPX.115

Level: UI-ERROR

Short Syntax: IPX.115 IPX cir *IPX_circuit* configd on ifc *IPX_interface_number*, too many bcast circs on ifc

Long Syntax: IPX.115 IPX circ *IPX_circuit* configured on interface *IPX_interface_number*, too many broadcast circs on interface.

Description: There can only be one broadcast circuit configured per interface. Thereis already one configured on this interface

Cause: I have no idea how this could happen

Action: Reconfigure the circuit onto another interface or delete the original broadcast circuit.

IPX.116

Level: UI-ERROR

Short Syntax: IPX.116 IPXWAN cir *IPX_circuit* configd on ifc *IPX_interface_number*, too many IPX cirs on PPP ifc

Long Syntax: IPX.116 IPXWAN circ *IPX_circuit* configured on interface *IPX_interface_number*, too many IPX circs on PPP interface

Description: On a PPP interface, there can be either a broadcast or an IPXWAN circuit. In this case, there is already an IPXWAN circuit defined.

Cause: I have no idea how this could happen

Action: Reconfigure the IPXWAN circuit on another interface or delete the original IPXWAN circuit.

IPX.117

Level: UI-ERROR

Short Syntax: IPX.117 IPXWAN cir *IPX_circuit* configd on ifc *IPX_interface_number*, not allowed on RLAN ifc

Long Syntax: IPX.117 IPXWAN circ *IPX_circuit* configured on interface *IPX_interface_number*, IPXWAN not allowed on RLAN interface

Description: An IPXWAN circuit has been configured on an RLAN interface. IPXWAN is not supported on RLAN interfaces.

Cause: I have no idea how this could happen

Action: Reconfigure the circuit on a non-RLAN interface or make it a broadcast circuit.

IPX.118

Level: UI-ERROR

Short Syntax: IPX.118 non-PPP IPXWAN cir *IPX_circuit* configd on PPP ifc *IPX_interface_number*

Long Syntax: IPX.118 non-PPP IPXAWN circ *IPX_circuit* configured on a PPP interface *IPX_interface_number*

Description: An IPXWAN circuit whose circuit type was configured as PPP has been configured on a non-PPP type interface

Cause: After configuring a PPP type IPXWAN circuit, the data type of the underlying interface has been changed to a non-PPP value

Action: Redefine the interface to be a PPP interface or redefine the IPXWAN circuit type to match the interface.

IPX.119

Level: UI-ERROR

Short Syntax: IPX.119 non-FR IPXWAN cir *IPX_circuit* configd on FR ifc *IPX_interface_number*

Long Syntax: IPX.119 non-FR IPXAWN circ *IPX_circuit* configured on a FR interface *IPX_interface_number*

Description: An IPXWAN circuit whose circuit type was configured as FR has been configured on a non-FR type interface

Cause: After configuring a FR type IPXWAN circuit, the data type of the underlying interface has been changed to a non-FR value

Action: Redefine the interface to be a FR interface or redefine the IPXWAN circuit type to match the interface.

IPX.120

Level: UI-ERROR

Short Syntax: IPX.120 IPXWAN cir *IPX_circuit* configd on ifc *IPX_interface_number*, invalid circuit type

Long Syntax: IPX.120 IPXWAN circ *IPX_circuit* configured on interface *IPX_interface_number*, invalid circuit type

Description: An IPXWAN circuit has been configured with an invalid circuit type value

Cause: I have no idea how this could happen

Action: Reconfigure the circuit with a valid circuit type

IPX.121

Level: UI-ERROR

Short Syntax: IPX.121 IPXWAN cir *IPX_circuit* nt *network ID*, n_cirreg failed rc: *reason_code*

Long Syntax: IPX.121 IPXWAN circ *IPX_circuit* net *network ID*, n_cirreg failed reason code: *reason_code*

Description: An IPXWAN circuit was attempting to register on a FR circuit and the registration failed for the reason indicated

Cause: No memory

Action: Take dump of router and contact customer service.

Cause: Unknown circuit

Action: Reconfigure IPXWAN circuit with a known FR circuit number

---

IPX.122

Level: P-TRACE

Short Syntax: IPX.122 NB brd pkt drop, cir *IPX_circuit* nt *network ID*, NB brd not enabled

Long Syntax: IPX.122 NETBIOS broadcast packet dropped, circ *IPX_circuit* net *network ID*, NETBIOS broadcast not enabled

Description: NETBIOS broadcast has been disabled on this circuit, so all NETBIOS broadcast packets (incoming or outgoing) will be dropped.

IPX.123

Level: UI-ERROR

Short Syntax: IPX.123 IPX cir *IPX_circuit* configd on ifc *IPX_interface_number*, IPX not supported on ifc

Long Syntax: IPX.123 IPX circ *IPX_circuit* configured on interface *IPX_interface_number*, IPX is not supported

Description: The interface that the circuit is suppose to be configured does not support IPX

Cause: The interface may be a member interface (i.e. LAG, PPP multi-link)

Action: Configure IPX on the aggregate interface

Chapter 55. Internet Protocol (IP)

This chapter describes Internet Protocol (IP) messages. For information on message content and how to use the message, refer to the Introduction.

IP.001

Level: U-INFO

Short Syntax: IP.001 q ovrf *source_ip_address* -> *destination_ip_address* nt *network ID*

Long Syntax: IP.001 Queue overflow on packet from *source_ip_address* for *destination_ip_address* from net *network ID*

Description: This message is generated when the forwarder must discard a packet that was not forwarded via the IP cache because of an input queue overflow. Note that this event does not get counted in ELS, it is instead counted in the IP console. The counters (kept per input network) can be read using the IP>COUNTERS command.

Cause: Input queue overflows happen when a packet is received from an interface that is short on buffers, the destination is not in the IP cache, and the length of the IP queue is greater than the fair share. This may be caused by either a burst or steady state of traffic arriving faster than the IP forwarder can forward it.

Action: Reduce traffic bursts. Upgrade to a faster router.

Cause: Excessive IP routing cache misses, causing most IP packets to go through the cache miss forwarder.

Action: Increase the size of the IP cache.

IP.002

Level: UE-ERROR

Short Syntax: IP.002 not V4 hdr *version_number* nt *network ID*

Long Syntax: IP.002 Not version 4 header (*version_number*) in packet from net *network ID*

Description: This message is generated when a packet has an incorrect version number.

Cause: Most likely, this packet was damaged since there should be no other versions of IP running.

Action: If the problem persists, examine a line trace to determine where the packet is being damaged.

IP.003

Level: UE-ERROR

Short Syntax: IP.003 shrt hdr *header_length* pkt ln *packet_length* nt *network ID*

Long Syntax: IP.003 Header too short (*header_length* bytes) in *packet_length* byte packet from net *network ID*

Description: This message is generated when a packet's indicated header length is below the minimum possible length.

Cause: Most likely, this is a damaged packet. It may be that another node is building an incorrect header.

Action: If the problem persists, examine a line trace to determine where the packet is being damaged.

IP.004

Level: UE-ERROR

Short Syntax: IP.004 bd hdr cks 0x *checksum* (exp 0x *expected_checksum*) *source_ip_address* -> *destination_ip_address*

Long Syntax: IP.004 Bad header checksum 0x *checksum* (expected 0x *expected_checksum*) in packet from *source_ip_address* for *destination_ip_address*

Description: This message is generated when a packet has an invalid checksum. The received checksum, together with the correct checksum, are displayed.

Cause: Most likely, this is a damaged packet. It may be that another node is building an incorrect header.

Action: If the problem persists, examine a line trace to determine where the packet is being damaged.

IP.005

Level: UE-ERROR

Short Syntax: IP.005 pkt trunc *specified_length* pkt ln *true_length source_ip_address -> destination_ip_address*

Long Syntax: IP.005 Packet truncated from *specified_length* to *true_length* bytes from *source_ip_address* for *destination_ip_address*

Description: This message is generated when the packet length specified in the header is greater than the packet buffer length.

Cause: Packet corruption in transit.

Action: If problem persists, check networks and routers.

Cause: Programming error in remote note.

IP.006

Level: CI-ERROR

Short Syntax: IP.006 pkt *source_ip_address* -> *destination_ip_address* dsc rsn *reason_code*, nt *Network ID*

Long Syntax: IP.006 Packet from *source_ip_address* for *destination_ip_address* discarded for reason *reason_code*, network *Network ID*

Description: An attempt was made to send the packet on the specified network, but it was not accepted for transmission on that network. The reason_code indicates why the packet was not accepted. If the reason was flow-control, an ICMP source quench will be sent to the sender, otherwise an ICMP destination unreachable will be sent.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

IP.007

Level: P-TRACE

Short Syntax: IP.007 *source_ip_address* -> *destination_ip_address*

Long Syntax: IP.007 Accepting packet from *source_ip_address* for *destination_ip_address*

Description: This message is generated for each packet which has passed first-level reasonableness checks.

IP.008

Level: U-INFO

Short Syntax: IP.008 no rte *source_ip_address* -> *destination_ip_address* dsc

Long Syntax: IP.008 No route for packet from *source_ip_address* for *destination_ip_address*; packet discarded

Description: This message is generated when a packet is discarded because there is no route to the destination.

IP.009

Level: CE-ERROR

Short Syntax: IP.009 TTL zero *source_ip_address* -> *destination_ip_address*

Long Syntax: IP.009 Time-to-live expired on packet from *source_ip_address* for *destination_ip_address*

Description: This message is generated when a packet is discarded because the time-to-live expired.

Cause: The packet has been through more routers than the initial value placed in the time-to-live field of the IP header by the originator. Many older systems use values of 15 or 30, which are not standard-conformant, and are often too small for current networks.

Action: Increase initial time-to-live value.

Cause: The packet was in a routing loop, going through a sequence of routers over and over until the time-to-live expired.

Action: Check the routing from the source of the packet to the destination, and see that there are no loops. However, temporary loops are an inevitable result of the timing out of routes in some routing protocols.

IP.011

Level: C-INFO

Level: PARAM

Short Syntax: IP.011 unsup mcst *source_ip_address* -> *destination_ip_address*

Long Syntax: IP.011 Unsupported multicast from *source_ip_address* for *destination_ip_address*

Description: This message is generated when an unsupported multicast packet is received.

IP.012

Level: UE-ERROR

Level: PARAM

Short Syntax: IP.012 bd nt cl *source_ip_address* -> *destination_ip_address*

Long Syntax: IP.012 Bad network class from *source_ip_address* for *destination_ip_address*

Description: This message is generated when a packet

is destined for a network which is not class A, B, C or D.

Cause: The indicated source node has sent a packet which the router cannot forward because the network class is unsupported.

IP.013

Level: C-INFO

Level: PARAM

Short Syntax: IP.013 unsup bcst *source_ip_address* -> *destination_ip_address*

Long Syntax: IP.013 Unsupported broadcast from *source_ip_address* for *destination_ip_address*

Description: This message is generated when an unsupported broadcast packet is received.

IP.017

Level: UI-ERROR

Level: PARAM

Short Syntax: IP.017 nt *network_address* add fail, tbl ovrfl

Long Syntax: IP.017 Add failed for net *network_address*; routing table overflow

Description: This message is generated when a network cannot be added to the routing table because the table is full.

Cause: The IP routing table contains the maximum number of entries.

Action: System administrator reduce table size by subnetworking.

IP.018

Level: UI-ERROR

Level: PARAM

Short Syntax: IP.018 nt *network_address* add fail, bd nt

Long Syntax: IP.018 Add failed for net *network_address*; bad network number

Description: This message is generated when a network cannot be added to the routing table because of a bad network number.

Cause: This software considers the net above to be invalid.

Action: If the net is valid, contact customer service.

IP.019

Level: U-INFO

Short Syntax: IP.019 re-add stat rt to *network*

Long Syntax: IP.019 Re-adding static route to net *network*

Description: This message is generated when a static route to a network is brought back into use.

IP.020

Level: UI-ERROR

Level: PARAM

Short Syntax: IP.020 int for *network* add fail, dup addr

Long Syntax: IP.020 Add of interface for net *network* failed; duplicate address

Description: This message is generated when a network cannot be added to the routing table because the access was denied.

Cause: There are multiple interface addresses configured which access the same network. The software only allows one.

Action: Reconfigure such that interface addresses and masks define unique networks.

IP.022

Level: U-INFO

Short Syntax: IP.022 add nt *net_ip_address* int *int_ip_address* nt *network ID*

Long Syntax: IP.022 Added network *net_ip_address* to interface *int_ip_address* on net *network ID*

Description: This message is generated when a new directly-connected network is added to the routing table.

IP.024

Level: CE-ERROR

Short Syntax: IP.024 ign stat rt to *network*, mask *mask*

Long Syntax: IP.024 Ignoring bad static route/filter to *network*, mask *mask*

Description: This message is generated when a bad static route or IP filter is encountered.

IP.025

Level: U-INFO

Short Syntax: IP.025 add nt *network* rt via *network* nt *network ID*

Long Syntax: IP.025 Added network *network* with

route via *network* on net *network ID*

Description: This message is generated when a new indirectly-connected network is added to the routing table.

IP.031

Level: ALWAYS

Short Syntax: IP.031 Unnum addr rej, nt *network ID*

Long Syntax: IP.031 Unnumbered address rejected, net *network ID*

Description: An attempt has been made to configure an interface as unnumbered, yet either the interface is not a serial line or the interface already has been assigned an IP address. The unnumbered configuration request is ignored.

IP.032

Level: CI-ERROR

Short Syntax: IP.032 fq ovf *source_ip_address* -> *destination_ip_address* nt *network ID*

Long Syntax: IP.032 Fragment queue overflow from *source_ip_address* for *destination_ip_address* on net *network ID*

Description: This message is generated when an incoming fragment is discarded because the fragment queue overflowed.

IP.033

Level: CE-ERROR

Short Syntax: IP.033 cant frg *source_ip_address* -> *destination_ip_address* nt *network ID*

Long Syntax: IP.033 Cannot fragment packet from *source_ip_address* for *destination_ip_address* net *network ID*

Description: This message is generated when an outgoing packet needs to be fragmented but has the "don't fragment" bit set.

IP.034

Level: CE-ERROR

Short Syntax: IP.034 bd frg *source_ip_address* -> *destination_ip_address* foff *offset*

Long Syntax: IP.034 Bad fragment from *source_ip_address* for *destination_ip_address* with fragment offset *offset*

Description: This message is generated when an outgoing packet has an invalid length of fragment offset.

IP.035

Level: CI-ERROR

Short Syntax: IP.035 cant alloc for frg nt *network ID*

Long Syntax: IP.035 Cannot allocate buffer for fragment for net *network ID*

Description: This message is generated when no buffer is available to fragment a packet.

IP.036

Level: P-TRACE

Short Syntax: IP.036 rcv pkt prt *protocol* frm *source_ip_address*

Long Syntax: IP.036 Received packet for protocol *protocol* from *source_ip_address*

Description: This message is generated for each packet destined for the router.

IP.037

Level: C-TRACE

Short Syntax: IP.037 brd pkt *source_ip_address* -> *destination_ip_address* prot *protocol* no srvr

Long Syntax: IP.037 Broadcast packet from *source_ip_address*, for *destination_ip_address*, protocol *protocol*; no server

Description: This message is generated when a broadcast packet arrives for an unknown protocol.

IP.038

Level: U-INFO

Short Syntax: IP.038 pkt *source_ip_address* -> *destination_ip_address* prt *protocol* no srvr

Long Syntax: IP.038 Packet from *source_ip_address*, for *destination_ip_address*, protocol *protocol*; no server

Description: This message is generated when a packet arrives for an unknown protocol. The packet was destined for the router.

IP.042

Level: CE-ERROR

Short Syntax: IP.042 illgl ARP sbnt req *source_ip_address* -> *destination_ip_address*

Long Syntax: IP.042 Illegal ARP subnet request in packet from *source_ip_address* for *destination_ip_address*

Description: This message is generated when an ARP subnet request is not honored due to illegal source or destination IP addresses in the ARP packet.

Cause: No route to requested subnet.

Action: Determine why subnet is not reachable.

Cause: Request is for different IP network than source address.

Action: ARP subnet routing is only for subnets of the host's network. Correct routing code in host.

Cause: IP network is not subnetted.

Action: ARP subnet routing is only supported on subnets.

IP.043

Level: P-TRACE

Short Syntax: IP.043 rcvd ARP sbnt rqst *source_ip_address* -> *destination_ip_address*

Long Syntax: IP.043 Received ARP subnet route request from *source_ip_address* for *destination_ip_address*

Description: This message is generated when an ARP subnet request is received.

IP.044

Level: C-TRACE

Short Syntax: IP.044 ARP sbnt rqst ign *source_ip_address* -> *destination_ip_address*

Long Syntax: IP.044 ARP subnet request ignored from *source_ip_address* for *destination_ip_address*

Description: This message is generated when an ARP subnet request is not answered because the route to the target subnet is via another router on the same physical network as the originator.

Cause: ARP subnet routing code will only respond when this router is the best route to the target subnet.

Action: The best router should respond to the ARP subnet request.

IP.045

Level: C-INFO

Short Syntax: IP.045 snt ARP rte *source_ip_address* -> *destination_ip_address*

Long Syntax: IP.045 Sent ARP route from *source_ip_address* for *destination_ip_address*

Description: This message is generated when an ARP subnet request is answered.

IP.046

Level: C-INFO

Short Syntax: IP.046 unkn opt *option* frm *source_ip_address*

Long Syntax: IP.046 Unknown option *option* from *source_ip_address*

Description: This message is generated when an unknown option is specified in the IP header of a packet.

IP.047

Level: UE-ERROR

Short Syntax: IP.047 opt *option* bd fmt frm *source_ip_address*

Long Syntax: IP.047 Bad format for option *option* from *source_ip_address*

Description: This message is generated when an option is incorrectly formatted in the IP header.

IP.048

Level: UE-ERROR

Short Syntax: IP.048 strict src rt bd nxt hop *source_ip_address* -> *destination_ip_address*

Long Syntax: IP.048 Bad next hop in strict source route from *source_ip_address* for *destination_ip_address*

Description: This message is generated when the next hop specified in the strict source route is invalid.

IP.049

Level: UE-ERROR

Short Syntax: IP.049 bd tmstmp fmt *timestamp* frm *source_ip_address*

Long Syntax: IP.049 Bad timestamp format *timestamp* from *source_ip_address*

Description: This message is generated when the format of the timestamp option is invalid.

IP.050

Level: CE-ERROR

Short Syntax: IP.050 tmstmp ovrf, *source_ip_address* -> *destination_ip_address*

Long Syntax: IP.050 Timestamp list overflow in packet from *source_ip_address* for *destination_ip_address*

Description: This message is generated when the timestamp list is full and the new timestamp cannot be added.

IP.051

Level: UI-ERROR

Short Syntax: IP.051 rs ovfl, port *port_number* frm *source_ip_address*

Long Syntax: IP.051 Too many re-assembly buffers active; port *port_number* from *source_ip_address*

Description: This message is generated when a new packet needs re-assembly but the maximum number of re-assembly buffers has already been assigned.

Cause: The software is attempting to reassemble more fragmented datagrams than it can handle simultaneously. This is acceptable on occasion.

Action: If this occurs frequently, attempt to reduce fragmentation by changing MSS at the source, or contact customer service.

IP.052

Level: UI-ERROR

Level: OOM

Short Syntax: IP.052 no stor for rs, port *port_number* frm *source_ip_address*

Long Syntax: IP.052 Insufficient storage for packet re-assembly; port *port_number* from *source_ip_address*

Description: This message is generated when a new packet needs re-assembly but there is not enough storage to allocate a re-assembly buffer.

Cause: Not enough memory to support this configuration and traffic.

Action: Check memory statistics in GWCON to verify packet buffer level. Upgrade for more memory, or disable unnecesary forwarders/protocols or get more memory.

IP.053

Level: UE-ERROR

Short Syntax: IP.053 reas pkt too big (*packet_size* byt), port *port_number* frm *source_ip_address*

Long Syntax: IP.053 Re-assembled packet too large (*packet_size* bytes); port *port_number* from *source_ip_address*

Description: This message is generated when a new packet needs re-assembly but it is larger than the maximum size re-assembly buffer. The packet_size is how large the packet would be after adding this fragment, which may not be the last.

IP.054

Level: U-INFO

Short Syntax: IP.054 rs TTL exp, port *port_number* frm *source_ip_address*

Long Syntax: IP.054 Re-assembly TTL expired; port *port_number* from *source_ip_address*

Description: This message is generated when a packet being re-assembled has its time-to-live expire.

IP.055

Level: P-TRACE

Short Syntax: IP.055 rs free, port *port_number* frm *source_ip_address*

Long Syntax: IP.055 Re-assembly buffer free; port *port_number* from *source_ip_address*

Description: This message is generated when a re-assembly buffer is de-allocated.

IP.056

Level: U-INFO

Short Syntax: IP.056 add dflt nt gw *ip_address* nt *network ID*

Long Syntax: IP.056 Added default gateway *ip_address* net *network ID*

Description: This message is generated when an interface using a default gateway comes up.

IP.057

Level: U-INFO

Short Syntax: IP.057 del dflt nt gw *ip_address*

Long Syntax: IP.057 Deleted default gateway *ip_address*

Description: This message is generated when a default gateway is deleted.

IP.058

Level: U-INFO

Short Syntax: IP.058 del nt *network* rt via *gateway* nt *network ID*

Long Syntax: IP.058 Deleted net *network* route via *gateway* net *network ID*

Description: This message is generated when a network goes down.

IP.059

Level: U-INFO

Short Syntax: IP.059 sbnt *network* dfnd

Long Syntax: IP.059 Subnet *network* defined

Description: This message is generated when a new subnetted network is defined.

IP.060

Level: U-INFO

Short Syntax: IP.060 del sbntd nt *network*

Long Syntax: IP.060 Deleting subnetted network *network*

Description: This message is generated when a subnetted network is deleted. This happens when there are no longer any interfaces to that network.

IP.061

Level: C-TRACE

Short Syntax: IP.061 add lcl pkt to ip op q

Long Syntax: IP.061 Added locally generated packet to IP output queue

Description: This message is generated whenever a locally generated packet is put on the IP output queue.

IP.062

Level: C-TRACE

Short Syntax: IP.062 rcvd ip frg frm *source_ip_address*

Long Syntax: IP.062 Received IP fragment from *source_ip_address*

Description: This message is generated when an IP fragment, requiring re-assembly is received.

IP.063

Level: C-TRACE

Short Syntax: IP.063 rasmd pkt frm *source_ip_address*

Long Syntax: IP.063 Successfully re-assembled packet from *source_ip_address*

Description: This message is generated when an IP packet has been successfully re-assembled.

IP.064

Level: C-TRACE

Short Syntax: IP.064 frg pkt *source_ip_address* -> *destination_ip_address*

Long Syntax: IP.064 Packet from *source_ip_address* for *destination_ip_address* requires fragmentation

Description: This message is generated when an IP packet needs to be fragmented for transmission.

IP.065

Level: C-TRACE

Short Syntax: IP.065 add frg to op frg q *source_ip_address* -> *destination_ip_address*

Long Syntax: IP.065 Added fragment to output fragment queue from *source_ip_address* for *destination_ip_address*

Description: This message is generated when an IP packet fragment is put on the output fragment queue.

IP.066

Level: P-TRACE

Short Syntax: IP.066 dsc pkt *source_ip_address* -> *destination_ip_address* nt *Network ID* no IP on int

Long Syntax: IP.066 Discarded packet from *source_ip_address* for *destination_ip_address* net *Network ID*, no IP on interface

Description: This message is generated by the stub IP forwarder for each packet which is received on an interface for which IP is not enabled.

IP.067

Level: UE-ERROR

Short Syntax: IP.067 RIPv1 subnet mismatch *interface_ip_address*/ *interface_ip_mask* vs *interface_ip_address*/ *interface_ip_mask*

Long Syntax: IP.067 RIP version 1 subnets with different masks for interface *interface_ip_address*/ *interface_ip_mask* and *interface_ip_address*/ *interface_ip_mask*

Description: The router is configured with variable length subnet masks on the same network. RIP version 1 will not advertise these subnets with this configuration.

IP.068

Level: U-INFO

Short Syntax: IP.068 routing cache cleared

Long Syntax: IP.068 routing cache cleared

Description: The IP routing cache has been cleared, probably as the result of a routing table change.

IP.069

Level: U-INFO

Short Syntax: IP.069 routing cache garbage collecting...

Long Syntax: IP.069 routing cache garbage collecting...

Description: The IP routing cache is collecting nonsense data. This takes several passes, and is only done when the cache starts overflowing.

IP.070

Level: U-INFO

Short Syntax: IP.070 cache entry *ip_destination* cleared

Long Syntax: IP.070 routing cache entry for destination *ip_destination* cleared

Description: The IP routing cache entry for the listed destination has been cleared.

IP.072

Level: UE-ERROR

Short Syntax: IP.072 LL broadcast *source_ip_address* -> *destination_ip_address*, discarded

Long Syntax: IP.072 Received link level broadcast from *source_ip_address* for *destination_ip_address*, discarded

Description: This message is generated when an attempt is made to forward an IP packet that was received as a link level broadcast/multicast. Such packets are not forwarded, and are discarded without even sending back an ICMP message to the source.

IP.073

Level: UI-ERROR

Short Syntax: IP.073 can't copy *source_ip_address* -> *destination_ip_address*, discarded

Long Syntax: IP.073 Can't copy packet from *source_ip_address* for *destination_ip_address*, discarded

Description: This message is generated when an attempt is made to copy a packet for one of the router's internal applications (e.g., during multicast forwarding), and the router is unable to get a buffer. The requested service then fails.

Cause: Not enough memory to support this configuration and traffic.

Action: Check memory statistics in GWCON to verify packet buffer level. Upgrade for more memory, or disable unnecesary forwarders/protocols or get more memory.

IP.078

Level: C-TRACE

Short Syntax: IP.078 AcCtl miss drop: *source_ip_address* -> *destination_ip_address*, tos *tos_byte*, prot *protocol*, net *networkID*: *direction*

Long Syntax: IP.078 Access control miss dropped, packet from *source_ip_address* to *destination_ip_address*, tos *tos_byte*, protocol number *protocol*, net *networkID*, direction *direction*

Description: This message is generated when a IP packet matches none of the access control records. The packet will be dropped.

IP.079

Level: C-TRACE

Short Syntax: IP.079 AcCtl miss drop: *source_ip_address* -> *destination_ip_address*, tos *tos_byte*, *protocol* port *source_port* -> *destination_port*, net *networkID*: *direction*

Long Syntax: IP.079 Access control miss dropped, packet from *source_ip_address* to *destination_ip_address*, tos *tos_byte*, *protocol* port number *source_port* to *destination_port*, net *networkID*, direction *direction*

Description: This message is generated when a IP packet matches none of the access control records. The packet will be dropped.

IP.080

Level: U-TRACE

Short Syntax: IP.080 new router *router_address*

Long Syntax: IP.080 new router *router_address* has been discovered

Description: A new router has been discovered, either through static configuration, an ICMP redirect, RIP or ICMP router discovery. This message is produced only when running as an IP host (i.e., when IP routing disabled).

IP.081

Level: UE-ERROR

Short Syntax: IP.081 IP ds nt rn on *nettype/ n_net*

Long Syntax: IP.081 IP protocol does not run over *nettype/ n_net*

Description: An IP address was configured for a type of network which currently doesn't support IP.

IP.082

Level: UE-ERROR

Short Syntax: IP.082 shrt pkt ln *packet_length*, *source_ip_address* -> *destination_ip_address*

Long Syntax: IP.082 IP length of *packet_length* in packet from *source_ip_address* -> *destination_ip_address* is too short

Description: This message is generated when a packet's indicated length is below the minimum possible length. The packet is discarded.

Cause: Most likely, this packet has been incorrectly formatted by the source.

IP.083

Level: C-TRACE

Short Syntax: IP.083 AcCtl # *record_number* drop: *cache_status*, *source_ip_address* -> *destination_ip_address*, tos *tos_byte*, prot *protocol_number*, net *networkID*: *direction*

Long Syntax: IP.083 Access control number *record_number* dropped, *cache_status*, packet from *source_ip_address* to *destination_ip_address*, tos *tos_byte*, IP protocol number *protocol_number*, net *networkID*, direction *direction*

Description: This message is generated when a IP packet matches one of the exclusive access control entries. The packet will be dropped. The record_number is the number of the access control record matched, or zero for no record (end-of-list). The cache_status will be ''cache-hit'' or ''cache-miss''.

IP.084

Level: C-TRACE

Short Syntax: IP.084 AcCtl # *record_number* pass: *cache_status*, *source_ip_address* -> *destination_ip_address*, tos *tos_byte*, prot *protocol_number*, net *networkID*: *direction*

Long Syntax: IP.084 Access control number *record_number* passed, *cache_status*, packet from *source_ip_address* to *destination_ip_address*, tos *tos_byte*, IP protocol number *protocol_number*, net *networkID*, direction *direction*

Description: This message is generated when a IP packet matches one of the inclusive access control entries. The packet may be forwarded. The record_number is the number of the access control record matched. The cache_status will be ''cache-hit'' or ''cache-miss''.

IP.085

Level: C-TRACE

Short Syntax: IP.085 AcCtl # *record_number* drop: *cache_status*, *source_ip_address* -> *destination_ip_address*, tos *tos_byte*, *protocol* port *source_port* -> *destination_port*, net *networkID*: *direction*

Long Syntax: IP.085 Access control number *record_number* dropped, *cache_status*, packet from *source_ip_address* to *destination_ip_address*, tos *tos_byte*, *protocol* port number *source_port* to *destination_port*, net *networkID*, direction *direction*

Description: This message is generated when a IP packet matches one of the TCP or UDP exclusive access control entries. The packet will be dropped. The record_number is the number of the access control record matched, or zero for no record (end-of-list). The cache_status will be ''cache-hit'' or ''cache-miss''.

IP.086

Level: C-TRACE

Short Syntax: IP.086 AcCtl # *record_number* pass: *cache_status*, *source_ip_address* -> *destination_ip_address*, tos *tos_byte*, *protocol* port *source_port* -> *destination_port*, net *networkID*: *direction*

Long Syntax: IP.086 Access control number *record_number* passed, *cache_status*, packet from *source_ip_address* to *destination_ip_address*, tos *tos_byte*, *protocol* port number *source_port* to *destination_port*, net *networkID*, direction *direction*

Description: This message is generated when a IP packet matches one of the TCP or UDP inclusive access control entries. The packet may be forwarded. The record_number is the number of the access control record matched. The cache_status will be ''cache-hit'' or ''cache-miss''.

IP.087

Level: U-INFO

Short Syntax: IP.087 Host svcs not instld; no IP addr

Long Syntax: IP.087 Host services is not installed as there is no IP address

Description: This message is generated when the host services is enabled, but the IP address is either not configured, or zero.

IP.088

Level: INFO

Short Syntax: IP.088 Autocnfg IP addr for host svcs

Long Syntax: IP.088 IP host address, and default gateway are being autoconfigured

Description: This message is generated when the host services is enabled, but the IP address is either not configured, or zero. The IP address, and the default gateway (if not configured) are autoconfigured from the previous boot information, if they exist.

IP.089

Level: C-TRACE

Short Syntax: IP.089 AcCtl # *rule_number rule_type*: *source_ip_address* -> *destination_ip_address*, tos *tos_byte*, *protocol_name* port *source_port_number* -> *destination_port_number*, frg *fragment_offset* *more_fragments*, net *networkID*: *direction* (SYSLOG *syslog_level*)

Long Syntax: IP.089 Access control number *rule_number* type *rule_type*, matched packet from *source_ip_address* to *destination_ip_address*, tos *tos_byte*, *protocol_name* port number *source_port_number* to *destination_port_number*, fragment *fragment_offset*

more_fragments, net *networkID*, direction *direction* (SYSLOG level *syslog_level*)

Description: An IP packet, protocol TCP or UDP, has matched an access control rule for which SYSLOG long logging is enabled. The packet is passed or dropped depending on the rule type.

IP.090

Level: C-TRACE

Short Syntax: IP.090 AcCtl # *rule_number rule_type*: *source_ip_address* -> *destination_ip_address*, tos *tos_byte*, prot *protocol_number*, frg *fragment_offset more_fragments*, net *networkID*: *direction* (SYSLOG *syslog_level*)

Long Syntax: IP.090 Access control number *rule_number* type *rule_type*, matched packet from *source_ip_address* to *destination_ip_address*, tos *tos_byte*, protocol number *protocol_number*, fragment *fragment_offset more_fragments*, net *networkID*, direction *direction* (SYSLOG level *syslog_level*)

Description: An IP packet, not protocol TCP or UDP, has matched an access control rule for which SYSLOG long logging is enabled. The packet is passed or dropped depending on the rule type.

IP.091

Level: C-TRACE

Short Syntax: IP.091 AcCtl # *rule_number rule_type*: *source_ip_address* -> *destination_ip_address*, tos *tos_byte*, prot *protocol_number* (SYSLOG *syslog_level*)

Long Syntax: IP.091 Access control number *rule_number* type *rule_type*, matched packet from *source_ip_address* to *destination_ip_address*, tos *tos_byte*, protocol number *protocol_number* (SYSLOG level *syslog_level*)

Description: An IP packet has matched an access control rule for which SYSLOG short logging is enabled. The packet is passed or dropped depending on the rule type.

IP.092

Level: C-TRACE

Short Syntax: IP.092 AcCtl # *rule_number* pass: *source_ip_address* -> *destination_ip_address*, tos *tos_byte*, prot *protocol_number protocol_name*, frg *fragment_offset more_fragments*, net *networkID*: *direction*

Long Syntax: IP.092 Access control number *rule_number* action pass, matched packet from *source_ip_address* to *destination_ip_address*, tos *tos_byte*, protocol number *protocol_number protocol_name*, fragment offset *fragment_offset more_fragments*, net *networkID*, direction *direction*

Description: An IP packet has matched an inclusive access control rule for which SNMP logging is enabled. The packet is passed.

IP.093

Level: C-TRACE

Short Syntax: IP.093 AcCtl # *rule_number* drop: *source_ip_address* -> *destination_ip_address*, tos *tos_byte*, prot *protocol_number protocol_name*, frg *fragment_offset more_fragments*, net *networkID*: *direction*

Long Syntax: IP.093 Access control number *rule_number* action drop, matched packet from *source_ip_address* to *destination_ip_address*, tos *tos_byte*, protocol number *protocol_number protocol_name*, fragment offset *fragment_offset more_fragments*, net *networkID*, direction *direction*

Description: An IP packet has matched an exclusive access control rule for which SNMP logging is enabled. The packet is dropped.

IP.094

Level: UI-ERROR

Short Syntax: IP.094 Add appl prot *protocol_number* port *local_port* adr *local_ip_address* adp *adapter_number* conflicts adp *adapter_number*

Long Syntax: IP.094 Add application protocol *protocol_number* local port *local_port* local IP address *local_ip_address* on adapter *adapter_number* conflicts with application on adapter *adapter_number*

Description: An attempt to add to the first specified adapter an application using the specified IP protocol number, local TCP or UDP port number, and local IP address conflicts with an application on the second specified adapter using the same parameters. The application remains on the second specified adapter.

Cause: Software error.

Action: Call customer service.

IP.095

Level: UI-ERROR

Short Syntax: IP.095 Add appl prot *protocol_number* port *local_port* adr *local_ip_address* adp *adapter_number* replaces adp *adapter_number*

Long Syntax: IP.095 Add application protocol *protocol_number* local port *local_port* local IP address *local_ip_address* on adapter *adapter_number* replaces application on adapter *adapter_number*

Description: An attempt to add to the first specified adapter an application using the specified IP protocol number, local TCP or UDP port number, and local IP address conflicts with an application on the second specified adapter using the same parameters. The application on the first specified adapter takes over.

Cause: Software error.

Action: Call customer service.

IP.096

Level: UI-ERROR

Short Syntax: IP.096 Del appl prot *protocol_number* port *local_port* adr *local_ip_address* adp *adapter_number* conflicts adp *adapter_number*

Long Syntax: IP.096 Delete application protocol *protocol_number* local port *local_port* local IP address *local_ip_address* from adapter *adapter_number* conflicts with application on adapter *adapter_number*

Description: An attempt to delete from the first specified adapter an application using the specified IP protocol number, local TCP or UDP port number, and local IP address conflicts with an application on the second specified adapter using the same parameters. The application remains on the second specified adapter.

Cause: Software error.

Action: Call customer service.

IP.097

Level: U-TRACE

Short Syntax: IP.097 Route *destination_ip_address/ mask status*

Long Syntax: IP.097 Route for *destination_ip_address* with mask *mask status*

Description: The route has been filtered from the IP route table or installed as a hidden route due to route table filtering policy.

IP.098

Level: ALWAYS

Short Syntax: IP.098 Route filter *destination_ip_address/ mask/ mask_definition/ exclude_include* not added due to *problem*

Long Syntax: IP.098 The route filter for Dest: *destination_ip_address* mask: *mask* Designation: *mask_definition* and policy: *exclude_include* not added due to *problem*.

Description: The route table filter could not be added.

Cause: Either it is a duplicate or memory could not be allocated for the route table filter.

Action: Assure there is enough memory to install the route filter policy.

IP.099

Level: U-TRACE

Short Syntax: IP.099 Dropped src rt pkt *source_ip_address -> destination_ip_address*

Long Syntax: IP.099 Dropped source routed packet from *source_ip_address* to *destination_ip_address*

Description: The forwarder has dropped a packet because the packet contains a source route IP option and the user has disabled IP source routing.

IP.101

Level: C-TRACE

Short Syntax: IP.101 Frg offset chk drop, *source_ip_address -> destination_ip_address*, net *networkID*

Long Syntax: IP.101 Fragment offset check dropped packet from *source_ip_address* to *destination_ip_address*, received from net *networkID*

Description: An IP packet, protocol TCP, has been dropped because its fragment offset is 1.

IP.102

Level: C-TRACE

Short Syntax: IP.102 Src adr chk drop, *source_ip_address -> destination_ip_address*, prot *protcol_number*, net *networkID*

Long Syntax: IP.102 Source address check dropped packet from from *source_ip_address* to *destination_ip_address*, protocol number *protcol_number*, received from net *networkID*

Description: An IP packet has been dropped because it was received from a network interface by which the router would not send packets to reach the received packet's source IP address.

IP.103

Level: C-TRACE

Short Syntax: IP.103 Rcrd Rt opt drop, *source_ip_address -> destination_ip_address*

Long Syntax: IP.103 Packet with record route option dropped, from *source_ip_address* to *destination_ip_address*

Description: An IP packet containing a record route option has been dropped because forwarding of packets containing a record route option is disabled.

IP.104

Level: C-TRACE

Short Syntax: IP.104 Tmstmp opt drop, *source_ip_address -> destination_ip_address*

Long Syntax: IP.104 Packet with timestamp option dropped, from *source_ip_address* to *destination_ip_address*

Description: An IP packet containing a timestamp option has been dropped because forwarding of packets containing a timestamp option is disabled.

IP.105

Level: C-TRACE

Short Syntax: IP.105 AcCtl # *rule_number rule_type*: *source_ip_address* -> *destination_ip_address*, tos *tos_byte*, *protocol_name* port *source_port_number* -> *destination_port_number*, frg *fragment_offset more_fragments*, net *networkID*: *direction*

Long Syntax: IP.105 Access control number *rule_number* type *rule_type*, matched packet from *source_ip_address* to *destination_ip_address*, tos *tos_byte*, *protocol_name* port number *source_port_number* to *destination_port_number*, fragment *fragment_offset more_fragments*, net *networkID*, direction *direction*

Description: An IP packet, protocol TCP or UDP, has matched an access control rule for which ELS long logging is enabled. The packet is passed or dropped depending on the rule type.

IP.106

Level: C-TRACE

Short Syntax: IP.106 AcCtl # *rule_number rule_type*: *source_ip_address* -> *destination_ip_address*, tos *tos_byte*, prot *protocol_number*, frg *fragment_offset more_fragments*, net *networkID*: *direction*

Long Syntax: IP.106 Access control number *rule_number* type *rule_type*, matched packet from *source_ip_address* to *destination_ip_address*, tos *tos_byte*, protocol number *protocol_number*, fragment *fragment_offset more_fragments*, net *networkID*, direction *direction*

Description: An IP packet, not protocol TCP or UDP, has matched an access control rule for which ELS long logging is enabled. The packet is passed or dropped depending on the rule type.

IP.107

Level: C-TRACE

Short Syntax: IP.107 AcCtl # *rule_number rule_type*: *source_ip_address* -> *destination_ip_address*, tos *tos_byte*, prot *protocol_number*

Long Syntax: IP.107 Access control number *rule_number* type *rule_type*, matched packet from *source_ip_address* to *destination_ip_address*, tos *tos_byte*, protocol number *protocol_number*

Description: An IP packet has matched an access control rule for which ELS short logging is enabled. The packet is passed or dropped depending on the rule type.

IP.108

Level: C-TRACE

Short Syntax: IP.108 AcCtl miss frg *action*: *source_ip_address* -> *destination_ip_address*, tos *tos_byte*, prot *protocol*, frg *fragment_offset more_fragments*, net *networkID*: *direction*

Long Syntax: IP.108 Access control miss fragment *action*, from *source_ip_address* to *destination_ip_address*, tos *tos_byte*, protocol number *protocol*, fragment *fragment_offset more_fragments*, net *networkID*, direction *direction*

Description: An IP fragment has not matched any access control rule. The packet is either dropped or saved for processing later.

IP.109

Level: P-TRACE

Short Syntax: IP.109 Pkt drop/held by *function*, *source_ip_address* -> *destination_ip_address*, dir *direction*

Long Syntax: IP.109 Packet dropped or held by *function*, packet from *source_ip_address* to *destination_ip_address*, direction *direction*

Description: This message is generated when an access control indicates the packet should be translated and the translate function, NAT, drops or holds that packet.

IP.110

Level: U-INFO

Short Syntax: IP.110 *packet_filter_name* AcCtl # *rule_number* changed: *change_description*

Long Syntax: IP.110 *packet_filter_name* access control number *rule_number* changed at installation: *change_description*

Description: During installation of an access control rule, a conflict has been detected in the rule, and the installed rule has been changed to resolve the conflict.

Action: Reconfigure the access control rule to correct the conflict.

IP.111

Level: C-INFO

Short Syntax: IP.111 *common_event*: id=x *ip_id* src= *source_ip_address*

Long Syntax: IP.111 *common_event*: id=x *ip_id* src= *source_ip_address*

Description: This message is generated by fragment

cache processing when a normal event occurs.

IP.112

Level: C-INFO

Short Syntax: IP.112 *common_event* for frg *fragment_offset more_fragments*: id=x *ip_id* src= *source_ip_address*

Long Syntax: IP.112 *common_event* for frg *fragment_offset more_fragments*: id=x *ip_id* src= *source_ip_address*

Description: This message is generated by fragment cache processing when a normal event occurs.

IP.113

Level: U-INFO

Short Syntax: IP.113 *unusual_event*: id=x *ip_id* src= *source_ip_address*

Long Syntax: IP.113 *unusual_event*: id=x *ip_id* src= *source_ip_address*

Description: This message is generated by fragment cache processing when an unusual event occurs.

IP.114

Level: UI-ERROR

Short Syntax: IP.114 *error_event*: id=x *ip_id* src= *source_ip_address*

Long Syntax: IP.114 *error_event*: id=x *ip_id* src= *source_ip_address*

Description: This message is generated by fragment cache processing when an error event occurs.

IP.115

Level: U-INFO

Short Syntax: IP.115 nt *network* int *change/unchanged/ old-mtu* mtu *new-mtu* from to

Long Syntax: IP.115 Network *network* interface *change/unchanged/ old-mtu* MTU *new-mtu* from to

Description: This message is during network up processing when the IP mtu is modified. If the layer 2 MTU is less than the configured IP, it will remain unchanged.

IP.116

Level: UE-ERROR

Short Syntax: IP.116 rcvd pkt len *ip_datagram_length* too short for GRE *source_ip_address-> destination_ip_address*

Long Syntax: IP.116 Received packet length *ip_datagram_length* too short for GRE, from *source_ip_address* to *destination_ip_address*

Description: A received IP datagram for the Generic Router Encapsulation (GRE) protocol is too short to hold the minimum GRE header. The datagram is discarded.

IP.117

Level: UE-ERROR

Short Syntax: IP.117 bd GRE cks 0x *checksum* (exp 0x *expected_checksum*) lcl *local_ip_address* rmt *remote_ip_address* prot 0x

Long Syntax: IP.117 Bad GRE checksum 0x *checksum* (expected 0x *expected_checksum*) in packet to *local_ip_address* from *remote_ip_address* GRE protocol 0x

Description: A received IP datagram for the Generic Router Encapsulation (GRE) protocol has an invalid GRE checksum. The datagram is discarded.

IP.118

Level: UE-ERROR

Short Syntax: IP.118 GRE src rtng not supported lcl *local_ip_address* rmt *remote_ip_address* prot 0x *gre_protocol_type*

Long Syntax: IP.118 GRE source routing not supported, local address *local_ip_address*, remote address *remote_ip_address*, GRE protocol 0x *gre_protocol_type*

Description: A received IP datagram for the Generic Router Encapsulation (GRE) protocol specifies GRE source routing, and the local router does not support GRE source routing. The datagram is discarded.

Cause: A GRE source route is configured in the remote router.

Action: Remove the GRE source route from the remote router's configuration.

IP.119

Level: UI-ERROR

Short Syntax: IP.119 rcvd GRE pkt for inactive tnl lcl *local_ip_address* rmt *remote_ip_address* prot 0x *gre_protocol_type*

Long Syntax: IP.119 Received GRE packet for inactive tunnel, local address *local_ip_address*, remote address *remote_ip_address*, GRE protocol 0x *gre_protocol_type*

Description: A received IP datagram for the Generic Router Encapsulation (GRE) protocol is for a GRE tunnel that is not active on this router, or is for a GRE payload protocol that is not configured to use this GRE tunnel in this router. The datagram is discarded.

Cause: A GRE tunnel between the local router and the

remote router is configured in the remote router but not in the local router.

Action: Add the GRE tunnel in the local router, or remove it from the remote router.

Cause: The local or remote IP address configured in the local router for this GRE tunnel does not match the remote or local IP address configured in the remote router.

Action: Correct the configuration for this GRE tunnel in the local or remote router.

Cause: The GRE payload protocol is configured to use this GRE tunnel in the remote router but not in the local router.

Action: Configure the payload protocol on the GRE tunnel in the local router, or remove it from the GRE tunnel on the remote router.

IP.120

Level: U-INFO

Short Syntax: IP.120 bfr unavl, GRE pkt dropped, lcl *local_ip_address* rmt *remote_ip_address* prot 0x *gre_protocol_type*

Long Syntax: IP.120 Buffer unavailable, GRE packet dropped, local address *local_ip_address*, remote address *remote_ip_address*, GRE protocol 0x *gre_protocol_type*

Description: No buffer is avaiable for a received IP datagram for the Generic Router Encapsulation (GRE) protocol. The datagram is discarded.

IP.121

Level: P-TRACE

Short Syntax: IP.121 GRE rcvd lcl *local_ip_address* rmt *remote_ip_address* prot 0x *gre_protocol_type*

Long Syntax: IP.121 GRE packet received, local address *local_ip_address*, remote address *remote_ip_address*, GRE protocol 0x *gre_protocol_type*

Description: A received IP datagram for the Generic Router Encapsulation (GRE) protocol is being passed to the GRE payload protocol.

IP.122

Level: U-INFO

Short Syntax: IP.122 No mem for GRE tnl lcl *local_ip_address* rmt *remote_ip_address* prot 0x *gre_protocol_type*

Long Syntax: IP.122 No memory for GRE tunnel, local address *local_ip_address*, remote address *remote_ip_address*, GRE protocol 0x *gre_protocol_type*

Description: Unable to allocate memory for the specified Generic Router Encapsulation (GRE) tunnel.

IP.123

Level: P-TRACE

Short Syntax: IP.123 GRE sent lcl *local_ip_address* rmt *remote_ip_address* prot 0x *gre_protocol_type*

Long Syntax: IP.123 GRE packet sent, local address *local_ip_address*, remote address *remote_ip_address*, GRE protocol 0x *gre_protocol_type*

Description: A IP datagram for the Generic Router Encapsulation (GRE) protocol has been passed to IP for transmission.

IP.124

Level: C-TRACE

Short Syntax: IP.124 DScache *action*: *source_ip_address*-> *destination_ip_address* tos= *tos_byte* prot= *protocol_number* info= *protocol_info* inNet= *in_net_number*

Long Syntax: IP.124 DiffServ cache *action*, matched packet from *source_ip_address* to *destination_ip_address*, tos *tos_byte*, protocol number *protocol_number* info *protocol_info* inbound net number *in_net_number*

Description: An IP packet has matched an entry in the IP DiffServ cache. The packet is passed or dropped depending on the entry's action.

IP.125

Level: P_TRACE

Short Syntax: IP.125 Trace IP datagram.

Long Syntax: IP.125 Trace IP datagram.

Description: IP datagram tracing.

Panic ininitnomem

Short Syntax: IP: no mem for init

Description: The router ran out of memory during IP initialization.

Action: Add memory, or reconfigure to reduce memory usage (for example, reduce the size of the IP routing table, or reduce the number of IP addresses).

Chapter 56. Internet Protocol Next Generation (IP V6)

This chapter describes Internet Protocol Next Generation (IP V6) messages. For information on message content and how to use the message, refer to the Introduction.

IPV6.001

Level: U-INFO

Short Syntax: IPV6.001 q ovrf *source_ipv6_address* -> *destination_ipv6_address* nt *network ID*

Long Syntax: IPV6.001 Queue overflow on packet from *source_ipv6_address* for *destination_ipv6_address* from net *network ID*

Description: This message is generated when the forwarder must discard a packet that was not forwarded via the IPV6 cache because of an input queue overflow. Note that this event does not get counted in ELS, it is instead counted in the IPV6 console. The counters (kept per input network) can be read using the IPV6>COUNTERS command.

Cause: Input queue overflows happen when a packet is received from an interface that is short on buffers, the destination is not in the IPV6 cache, and the length of the IP queue is greater than the fair share. This may be caused by either a burst or steady state of traffic arriving faster than the IPV6 forwarder can forward it.

Action: Reduce traffic bursts. Upgrade to a faster router.

Cause: Excessive IPV6 routing cache misses, causing most IPV6 packets to go through the cache miss forwarder.

Action: Increase the size of the IPV6 cache.

IPV6.002

Level: UE-ERROR

Short Syntax: IPV6.002 not V6 hdr *version_number* nt *network ID*

Long Syntax: IPV6.002 Not version 6 header (*version_number*) in packet from net *network ID*

Description: This message is generated when a packet has an incorrect version number.

Cause: Most likely, this packet was damaged since there should be no other versions of IP packets received in the IPV6 forwarder.

Action: If the problem persists, examine a line trace to determine where the packet is being damaged.

IPV6.003

Level: UE-ERROR

Short Syntax: IPV6.003 shrt hdr *packet_length_just_header* pkt ln *packet_length* nt *network ID*

Long Syntax: IPV6.003 Packet too short (*packet_length_just_header* bytes) in *packet_length* byte packet from net *network ID*

Description: This message is generated when a packet's indicated length is below the minimum possible length (less than the size of and IPV6 header).

Cause: Most likely, this is a damaged packet. It may be that another node is building an incorrect header.

Action: If the problem persists, examine a line trace to determine where the packet is being damaged.

IPV6.004

Level: U-INFO

Short Syntax: IPV6.004 pkt *source_ipv6_address* -> *destination_ipv6_address* dsc, mcst src addr

Long Syntax: IPV6.004 Invalid source IP6 address (multcast or anycast) *source_ipv6_address* for *destination_ipv6_address*; packet discarded

Description: This message is generated when a packet is discarded because of invalid source IPV6 address.

IPV6.005

Level: UE-ERROR

Short Syntax: IPV6.005 pkt trunc *specified_length* pkt ln *true_length source_ipv6_address -> destination_ipv6_address*

Long Syntax: IPV6.005 Packet len too short: IP len *specified_length* bytes Buffer len *true_length* bytes, from *source_ipv6_address* for *destination_ipv6_address*; packet discard

Description: This message is generated when the packet length specified in the header is greater than the packet buffer length.

Cause: Packet corruption in transit.

Action: If problem persists, check networks and routers.

Cause: Programming error in remote note.

IPV6.006

Level: CI-ERROR

Short Syntax: IPV6.006 pkt *source_ipv6_address* -> *destination_ipv6_address* dsc rsn *reason_code*, nt *Network ID*

Long Syntax: IPV6.006 Packet from *source_ipv6_address* for *destination_ipv6_address* discarded for reason *reason_code*, network *Network ID*

Description: An attempt was made to send the packet on the specified network, but it was not accepted for transmission on that network. The reason_code indicates why the packet was not accepted. If the reason was overflow the packet will be discarded. Other reason codes will cause an ICMP destination unreachable message to be sent.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

IPV6.007

Level: P-TRACE

Short Syntax: IPV6.007 *source_ipv6_address* -> *destination_ipv6_address next_header location*

Long Syntax: IPV6.007 Accepting packet from *source_ipv6_address* for *destination_ipv6_address* next header *next_header* at *location*

Description: This message is generated for each packet which has passed first-level reasonableness checks.

IPV6.008

Level: U-INFO

Short Syntax: IPV6.008 no rte *source_ipv6_address* -> *destination_ipv6_address* dsc

Long Syntax: IPV6.008 No route for packet from *source_ipv6_address* for *destination_ipv6_address*; packet discarded

Description: This message is generated when a packet is discarded because there is no route to the destination.

IPV6.009

Level: CE-ERROR

Short Syntax: IPV6.009 hop limit zero *source_ipv6_address* -> *destination_ipv6_address*

Long Syntax: IPV6.009 Hop limit reaches zero from *source_ipv6_address* for *destination_ipv6_address*

Description: This message is generated when a packet is discarded because hop limit reaches zero.

Cause: The packet has been through more routers than the initial value placed in the hop limit field of the IPV6 header by the originator. Many older systems use values of 15 or 30, which are not standard-conformant, and are often too small for current networks.

Action: Increase initial hop limit value.

Cause: The packet was in a routing loop, going through a sequence of routers over and over until the hop limit reaches zero.

Action: Check the routing from the source of the packet to the destination, and see that there are no loops. However, temporary loops are an inevitable result of the timing out of routes in some routing protocols.

IPV6.010

Level: U-INFO

Short Syntax: IPV6.010 dsc pkt *source_IPV6_address* -> *destination_IPV6_address*, dst lnk-lcl addr not ours, nt *Network ID*

Long Syntax: IPV6.010 Discarded packet from *source_IPV6_address* for *destination_IPV6_address* net *Network ID*

Description: This message is generated when receiving a packet with destination address is a link local address that does not match this interface's link local address.

Cause: Problem at the sender of this packet.

Action: Check the sender of this packet.

IPV6.011

Level: C-INFO

Level: PARAM

Short Syntax: IPV6.011 unsup mcst *source_ipv6_address* -> *destination_ipv6_address*

Long Syntax: IPV6.011 Unsupported multicast from *source_ipv6_address* for *destination_ipv6_address*

Description: This message is generated when an

unsupported multicast packet is received.

IPV6.012

Level: U-INFO

Short Syntax: IPV6.012 Disc. pkt *source_ipv6_address* -> *destination_ipv6_address* nhd *nextheader*

Long Syntax: IPV6.012 Packet from *source_ipv6_address* for *destination_ipv6_address* discarded for protocol *nextheader*

Description: The received packet has zero source address. It will be discarded.

Cause: Upper protocol sends out a packet with zero source address

Action: Trace down this culprit by looking at the protocol header to try and identify the source of the packet.

IPV6.013

Level: C-INFO

Level: PARAM

Short Syntax: IPV6.013 unsup bcst/link local address *source_ipv6_address* -> *destination_ipv6_address*

Long Syntax: IPV6.013 Unsupported broadcast or link local address from *source_ipv6_address* for *destination_ipv6_address*

Description: This message is generated when an unsupported broadcast or link local packet is received.

IPV6.014

Level: C-TRACE

Short Syntax: IPV6.014 Acc cont # *record_number*, *packet_status*, from *source_ip_address* -> *destination_ip_address*, dir *direction*, net *networkID*

Long Syntax: IPV6.014 Access control number *record_number* matched, *packet_status*, packet from *source_ip_address* to *destination_ip_address*, direction *direction*, net *networkID*

Description: This message is generated when a IPv6 packet matches one of the access control entries. The packet may be forwarded or dropped depend on the filter rule. The record_number is the number of the access control record matched.

IPV6.015

Level: UI_ERROR

Short Syntax: IPV6.015 rtg hdr ind but not fnd, *source_address* -> *destination_address*

Long Syntax: IPV6.015 A packet was received that indicates a routing header is present, but was not found from *source_address* to *destination_address*

Description: This message is generated when a packet is being processed that indicates a routing header is present, but was not found.

Action: Contact customer service.

IPV6.016

Level: UI_ERROR

Short Syntax: IPV6.016 can't get memory for frag, *source_address* -> *destination_address*

Long Syntax: IPV6.016 Can't get memory for fragmenting a packet from *source_address* to *destination_address*, packet discarded

Description: This message is generated when a packet is being processed for fragmentation but there is not enough memory available to complete the processing.

Cause: Not enough memory to support this configuration and traffic.

Action: Check memory statistics in GWCON to verify packet memory level. Upgrade for more memory or disable unnecessary forwarders/protocols to get more memory.

IPV6.017

Level: UI-ERROR

Level: PARAM

Short Syntax: IPV6.017 nt *network_address* add fail, tbl ovrfl

Long Syntax: IPV6.017 Add failed for net *network_address*; routing table overflow

Description: This message is generated when a network cannot be added to the routing table because the table is full.

Cause: The IPV6 routing table contains the maximum number of entries.

Action: See if the system administrator can reduce the table size by subnetworking.

IPV6.018

Level: UI-ERROR

Level: PARAM

Short Syntax: IPV6.018 nt *network_address* add fail, bd nt

Long Syntax: IPV6.018 Add failed for net *network_address*; bad network number

Description: This message is generated when a network cannot be added to the routing table because of a bad network number.

Cause: This software considers the net above to be invalid.

Action: If the net is valid, contact customer service.

IPV6.019

Level: U-INFO

Short Syntax: IPV6.019 re-add stat rt to *network*

Long Syntax: IPV6.019 Re-adding static route to net *network*

Description: This message is generated when a static route to a network is brought back into use.

IPV6.020

Level: UI-ERROR

Level: PARAM

Short Syntax: IPV6.020 int for *network* add fail, dup addr

Long Syntax: IPV6.020 Add of interface for net *network* failed; duplicate address

Description: This message is generated when a network cannot be added to the routing table because the access was denied.

Cause: There are multiple interface addresses configured which access the same network. The software only allows one.

Action: Reconfigure such that interface addresses and prefix lengths define unique networks.

IPV6.021

Level: C_INFO

Short Syntax: IPV6.021 pkt rcvd with no nxt hdr *source_address -> destination_address*

Long Syntax: IPV6.021 A packet was received that contains a No Next Header from *source_address -> destination_address*

Description: This message is generated when a No Next Header is found in the received packet. The packet will be dropped.

IPV6.022

Level: U-INFO

Short Syntax: IPV6.022 add nt *net_ipv6_address* int *int_ipv6_address* nt *network ID*

Long Syntax: IPV6.022 Added network *net_ipv6_address* to interface *int_ipv6_address* on net *network ID*

Description: This message is generated when a new directly-connected network is added to the routing table.

IPV6.023

Level: UE_ERROR

Short Syntax: IPV6.023 mltcst addr fnd in rtng hdr info from *source_address* to *destination_address*, *next_hop*

Long Syntax: IPV6.023 A multi-cast address was found in the routing header information from *source_address* to *destination_address*, next hop is *next_hop*

Description: This message is generated when a multi-cast address if found either in the routing header next hop information or the destination address field of the packet with a routing header. This packet will be dropped and an ICMP error message will be generated.

Action: Contact customer service for the machine that was the source of the packet.

IPV6.024

Level: CE-ERROR

Short Syntax: IPV6.024 ign stat rt to *network*, masksize *mask*

Long Syntax: IPV6.024 Ignoring bad static route/filter to *network*, masksize *mask*

Description: This message is generated when a bad static route or IPV6 filter is encountered.

IPV6.025

Level: U-INFO

Short Syntax: IPV6.025 add nt *network* rt via *network* nt *network ID*

Long Syntax: IPV6.025 Added network *network* with route via *network* on net *network ID*

Description: This message is generated when a new indirectly-connected network is added to the routing table.

IPV6.026

Level: UE_ERROR

Short Syntax: IPV6.026 hop-by-hop options fnd not imm after IPV6 hdr *source_address -> destination_address*

Long Syntax: IPV6.026 A Hop-by-Hop Options header was found not immediately following the IPV6 header from *source_address* to *destination_address*

Description: This message is generated when a Hop-by-Hop Options Header is found that does not immediately follow the IPV6 header, which is a violation of the IPV6 architecture. This packet will be dropped and an ICMP error message will be generated.

Action: Contact customer service for the machine that was the source of the packet.

IPV6.027

Level: C-INFO

Short Syntax: IPV6.027 pkt type *type* code *code* *source_ipv4_address* -> *destination_ipv4_address* unable to gen icmpv6 pkt

Long Syntax: IPV6.027 Insufficent data in the ICMPV4 packet received from the IPV6 over IPV4 tunnel (type *type* code *code* from *source_ipv4_address* to *destination_ipv4_address*) to build an ICMPV6 packet back to the original source.

Description: This message is generated when an ICMPV4 message is received from the tunnel and the router that generated this message did not include enough of the packet in error to retrieve the original IPV6 source address. This ICMPV4 packet will be dropped.

Action: Turn on IPV4 ICMP messages to determine the error being generated in the IPV4 network.

IPV6.028

Level: UE-ERROR

Short Syntax: IPV6.028 unhndled icmpv4 pkt rcv'd type *type* code *code source_ipv4_address* -> *destination_ipv4_address*

Long Syntax: IPV6.028 An ICMPV4 packet has been received from the IPV6 over IPV4 tunnel that is not understood/handled by the router, type *type*, code *code*, from *source_ipv4_address* to *destination_ipv4_address*

Description: This message is generated when the software receives an ICMP message from the IPV4 network that is not understood.

Action: Contact customer service.

IPV6.029

Level: P-TRACE

Short Syntax: IPV6.029 pkt *ip4src* -> *ip4dest*, *ip6src* -> *ip6dest* rcvd on *tuntype* tun(*tunint*)

Long Syntax: IPV6.029 Packet from IPv4 *ip4src* to *ip4dest*, IPv6 *ip6src* to *ip6dest* was received on *tuntype* tunnel(interface *tunint*)

Description: An IPv6 packet, encapsulated in an IPv4 header, has been received on the specified tunnel interface.

IPV6.030

Level: P-TRACE

Short Syntax: IPV6.030 *source_ipv6_address* -> *destination_ipv6_address* tnled over *source_ipv4_address* -> *destination_ipv4_address*

Long Syntax: IPV6.030 Forwarding packet from *source_ipv6_address* for *destination_ipv6_address* over tunnel *source_ipv4_address* to *destination_ipv4_address*

Description: This message is generated for each packet that is ready to be forwarded into the IPV6 over IPV4 tunnel.

IPV6.031

Level: ALWAYS

Short Syntax: IPV6.031 Unnum addr rej, nt *network ID*

Long Syntax: IPV6.031 Unnumbered address rejected, net *network ID*

Description: An attempt has been made to configure an interface as unnumbered, yet either the interface is not a serial line or the interface already has been assigned an IPV6 address. The unnumbered configuration request is ignored.

IPV6.032

Level: CI-ERROR

Short Syntax: IPV6.032 fq ovf *source_ipv6_address* -> *destination_ipv6_address* nt *network ID*

Long Syntax: IPV6.032 Fragment queue overflow from *source_ipv6_address* for *destination_ipv6_address* on net *network ID*

Description: This message is generated when an incoming fragment is discarded because the fragment queue overflowed.

IPV6.033

Level: UE-ERROR

Short Syntax: IPV6.033 reas pkt too big (*packet_size* byt), frm *source_ipv6_address*

Long Syntax: IPV6.033 Re-assembled packet too large (*packet_size* bytes); from *source_ipv6_address*

Description: This message is generated when a new packet being reassembled would exceed the maximum size allowed for fragmentation (65535 bytes). The packet_size is how large the packet would be after adding this fragment, which may not be the last.

IPV6.034

Level: CE-ERROR

Short Syntax: IPV6.034 bd frg *source_ipv6_address* -> *destination_ipv6_address* foff *offset*

Long Syntax: IPV6.034 Bad fragment from *source_ipv6_address* for *destination_ipv6_address* with fragment offset *offset*

Description: This message is generated when an incoming packet fragment has a length that is not a multiple of 8 bytes and the More bit in the fragment header is set to 1.

IPV6.035

Level: CI-ERROR

Short Syntax: IPV6.035 cant alloc for frg nt *network ID*

Long Syntax: IPV6.035 Cannot allocate buffer for fragment for net *network ID*

Description: This message is generated when no buffer is available to fragment a packet.

IPV6.036

Level: P-TRACE

Short Syntax: IPV6.036 rcv pkt nxt hdr *next_header* frm *source_ipv6_address*

Long Syntax: IPV6.036 Received packet with next header *next_header* from *source_ipv6_address*

Description: This message is generated for each packet destined for the router.

IPV6.037

Level: C-TRACE

Short Syntax: IPV6.037 brd pkt *source_ipv6_address* -> *destination_ipv6_address* prot *protocol* no srvr

Long Syntax: IPV6.037 Broadcast packet from *source_ipv6_address*, for *destination_ipv6_address*, protocol *protocol*; no server

Description: This message is generated when a broadcast packet arrives for an unknown protocol.

IPV6.038

Level: U-INFO

Short Syntax: IPV6.038 pkt *source_ipv6_address* -> *destination_ipv6_address* prt *protocol* no srvr

Long Syntax: IPV6.038 Packet from *source_ipv6_address*, for *destination_ipv6_address*, protocol *protocol*; no server

Description: This message is generated when a packet arrives for an unknown protocol. The packet was destined for the router.

IPV6.039

Level: C-INFO

Short Syntax: IPV6.039 pkt size = *packet_size* needs frag, tnl *tunnel_addr*, frag *frag_state*

Long Syntax: IPV6.039 A *packet_size* byte IPV6 packet will be fragmented over an IPV4 tunnel *tunnel_addr* with a configured fragmentation of *frag_state*

Description: This message is generated when an IPV6 packet is being tunneled that requires fragmentation. If fragmentation is enabled or the tunnel MTU is less than or equal to 1300 bytes, the DNF bit will not be set and fragmentation might occur in the tunnel. If fragmentation is not allowed and the tunnel MTU is greater than 1300 bytes, a Packet Too Big message will be generated back to the IPV6 source and the packet will be dropped.

IPV6.040

Level: UI-ERROR

Short Syntax: IPV6.040 Tnl endpnts not found for *ipsrc-> ipdest*

Long Syntax: IPV6.040 Tunnel endpoints were not found in table for IPv4 source address *ipsrc* and dest address *ipdest*

Description: This message is generated when the IPV6 over IPV4 tunneling code receives an encapsulated IPv6 packet and cannot determine which configured tunnel the packet arrived over.

Action: Check network's tunnel configuration parameters to verify that the IPv4 source and destination addresses are consistent at each endpoint.

IPV6.041

Level: UI-ERROR

Short Syntax: IPV6.041 rs ovfl, frm *source_ipv6_address*

Long Syntax: IPV6.041 Too many re-assembly buffers active; from *source_ipv6_address*

Description: This message is generated when a new packet needs re-assembly but the maximum number of re-assembly buffers has already been assigned.

Cause: The software is attempting to reassemble more fragmented datagrams than it can handle simultaneously. This is acceptable on occasion.

Action: If this occurs frequently, attempt to reduce fragmentation by changing MSS at the source, or contact customer service.

IPV6.042

Level: UI-ERROR

Short Syntax: IPV6.042 no stor for rs, frm *source_ipv6_address*

Long Syntax: IPV6.042 Insufficient storage for packet re-assembly; from *source_ipv6_address*

Description: This message is generated when a new packet needs re-assembly but there is not enough storage to allocate a re-assembly buffer.

Cause: Not enough memory to support this configuration and traffic.

Action: Check memory statistics in GWCON to verify packet buffer level. Upgrade for more memory, or disable unnecesary forwarders/protocols or get more memory.

IPV6.043

Level: UE-ERROR

Short Syntax: IPV6.043 reas pkt too big (*packet_size* byt), frm *source_ipv6_address*

Long Syntax: IPV6.043 Re-assembled packet too large (*packet_size* bytes); from *source_ipv6_address*

Description: This message is generated when a new packet needs re-assembly but it is larger than the maximum size re-assembly buffer. The packet_size is how large the packet would be after adding this fragment, which may not be the last.

IPV6.044

Level: U-INFO

Short Syntax: IPV6.044 rs TTL exp, frm *source_ipv6_address*

Long Syntax: IPV6.044 Re-assembly TTL expired; from *source_ipv6_address*

Description: This message is generated when a packet being re-assembled has its time-to-live expire.

IPV6.045

Level: P-TRACE

Short Syntax: IPV6.045 rs free, frm *source_ipv6_address*

Long Syntax: IPV6.045 Re-assembly buffer free; from *source_ipv6_address*

Description: This message is generated when a re-assembly buffer is de-allocated.

IPV6.046

Level: U-INFO

Short Syntax: IPV6.046 del nt *network* rt via *gateway* nt *network ID*

Long Syntax: IPV6.046 Deleted net *network* route via *gateway* net *network ID*

Description: This message is generated when a network goes down.

IPV6.047

Level: U-INFO

Short Syntax: IPV6.047 sbnt *network* dfnd

Long Syntax: IPV6.047 Subnet *network* defined

Description: This message is generated when a new subnetted network is defined.

IPV6.048

Level: U-INFO

Short Syntax: IPV6.048 del sbntd nt *network*

Long Syntax: IPV6.048 Deleting subnetted network *network*

Description: This message is generated when a subnetted network is deleted. This happens when there are no longer any interfaces to that network.

IPV6.049

Level: C-TRACE

Short Syntax: IPV6.049 add lcl pkt to ipv6 op q

Long Syntax: IPV6.049 Added locally generated packet to IPV6 output queue

Description: This message is generated whenever a locally generated packet is put on the IPV6 output queue.

IPV6.050

Level: C-TRACE

Short Syntax: IPV6.050 rcvd IPV6 frg frm *source_IPV6_address*, frg id *frag_id*, frg off *frag_off*

Long Syntax: IPV6.050 Received IPV6 fragment from *source_IPV6_address*, fragment id *frag_id*, fragment offset *frag_off*

Description: This message is generated when an IPV6 fragment requiring re-assembly is received.

IPV6.051

Level: C-TRACE

Short Syntax: IPV6.051 rasmd pkt frm *source_IPV6_address*

Long Syntax: IPV6.051 Successfully re-assembled packet from *source_IPV6_address*

Description: This message is generated when an IPV6 packet has been successfully re-assembled.

IPV6.052

Level: C-TRACE

Short Syntax: IPV6.052 frg pkt *source_IPV6_address* -> *destination_IPV6_address*

Long Syntax: IPV6.052 Packet from *source_IPV6_address* for *destination_IPV6_address* requires fragmentation

Description: This message is generated when an IPV6 packet needs to be fragmented for transmission.

IPV6.053

Level: C-TRACE

Short Syntax: IPV6.053 add frg to op frg q *source_IPV6_address* -> *destination_IPV6_address*

Long Syntax: IPV6.053 Added fragment to output fragment queue from *source_IPV6_address* for *destination_IPV6_address*

Description: This message is generated when an IPV6 packet fragment is put on the output fragment queue.

IPV6.054

Level: P-TRACE

Short Syntax: IPV6.054 dsc pkt *source_IPV6_address* -> *destination_IPV6_address* nt *Network ID* no IPV6 on int

Long Syntax: IPV6.054 Discarded packet from *source_IPV6_address* for *destination_IPV6_address* net *Network ID*, no IPV6 on interface

Description: This message is generated by the stub IPV6 forwarder for each packet which is received on an interface for which IPV6 is not enabled.

IPV6.055

Level: UE-ERROR

Short Syntax: IPV6.055 RIP6 disabld on int *interface_IPV6_address* var len sbnt msks

Long Syntax: IPV6.055 RIP6 disabled on *interface_IPV6_address* variable length subnet masks

Description: The router is configured with variable length subnet masks on the same network, which RIP can't handle. Thus RIP6 is disabled on the interface.

IPV6.056

Level: U-INFO

Short Syntax: IPV6.056 routing cache cleared for IPV6

Long Syntax: IPV6.056 routing cache cleared for IPV6

Description: The IPV6 routing cache has been cleared, probably as the result of a routing table change.

IPV6.057

Level: U-INFO

Short Syntax: IPV6.057 routing cache garbage collecting for IPV6

Long Syntax: IPV6.057 Routing cache garbage collecting for IPV6.

Description: The IPV6 routing cache is collecting nonsense data. This takes several passes, and is only done when the cache starts overflowing.

IPV6.058

Level: U-INFO

Short Syntax: IPV6.058 cache entry *IPV6_destination* cleared

Long Syntax: IPV6.058 routing cache entry for destination *IPV6_destination* cleared

Description: The IPV6 routing cache entry for the listed destination has been cleared.

IPV6.059

Level: UE-ERROR

Short Syntax: IPV6.059 LL broadcast *source_ipv6_address* -> *destination_ipv6_address*, discarded

Long Syntax: IPV6.059 Received link level broadcast from *source_ipv6_address* for *destination_ipv6_address*, discarded

Description: This message is generated when an attempt is made to forward an IPV6 packet that was received as a link level broadcast/multicast. Such packets are not forwarded, and are discarded without even sending back an ICMP message to the source.

IPV6.060

Level: UI-ERROR

Short Syntax: IPV6.060 can't copy *source_ipv6_address* -> *destination_ipv6_address*, discarded

Long Syntax: IPV6.060 Can't copy packet from *source_ipv6_address* for *destination_ipv6_address*, discarded

Description: This message is generated when an attempt is made to copy a packet for one of the router's internal applications (e.g., during multicast forwarding), and the router is unable to get a buffer. The requested service then fails.

Cause: Not enough memory to support this configuration and traffic.

Action: Check memory statistics in GWCON to verify packet buffer level. Upgrade for more memory, or disable unnecesary forwarders/protocols or get more memory.

IPV6.061

Level: C-TRACE

Short Syntax: IPV6.061 Acc cont miss dropped, *source_ipv6_address* -> *destination_ipv6_address*, dir *direction*, *autotunn*net *networkID*

Long Syntax: IPV6.061 Access control miss dropped, packet from *source_ipv6_address* to *destination_ipv6_address*, direction *direction*, *autotunn*net *networkID*

Description: This message is generated when a IPV6 packet matches none of the access control records. The packet will be dropped.

IPV6.062

Level: UE-ERROR

Short Syntax: IPV6.062 IPV6 ds nt rn on *nettype/ n_net*

Long Syntax: IPV6.062 IPV6 protocol does not run over *nettype/ n_net*

Description: An IPV6 address was configured for a type of network which currently doesn't support IPV6.

IPV6.063

Level: UE-ERROR

Short Syntax: IPV6.063 shrt pkt ln *packet_length*, *source_ipv6_address* -> *destination_ipv6_address*

Long Syntax: IPV6.063 IPV6 length of *packet_length* in packet from *source_ipv6_address* -> *destination_ipv6_address* is too short

Description: This message is generated when a packet's length in the IPV6 header is below less than the length indicated in the recieve The packet is discarded.

Cause: Most likely, this packet has been incorrectly formatted by the source.

IPV6.064

Level: C-TRACE

Short Syntax: IPV6.064 Acc cont # *record_number* dropped, *cache_status*, *source_ipv6_address* -> *destination_ipv6_address*, dir *direction*, *autotunn*net *networkID*

Long Syntax: IPV6.064 Access control number *record_number* dropped, *cache_status*, packet from *source_ipv6_address* to *destination_ipv6_address*, direction *direction*, *autotunn*net *networkID*

Description: This message is generated when a IPV6 packet matches one of the exclusive access control entries. The packet will be dropped. The record_number is the number of the access control record matched, or zero for no record (end-of-list). The cache_status will be ''cache-hit'' or ''cache-miss''.

IPV6.065

Level: C-TRACE

Short Syntax: IPV6.065 Acc cont # *record_number* passed, *cache_status*, *source_ipv6_address* -> *destination_ipv6_address*, dir *direction*, *autotunn*net *networkID*

Long Syntax: IPV6.065 Access control number *record_number* passed, *cache_status*, packet from *source_ipv6_address* to *destination_ipv6_address*, direction *direction*, *autotunn*net *networkID*

Description: This message is generated when a IPV6 packet matches one of the inclusive access control entries. The packet may be forwarded. The record_number is the number of the access control record matched. The cache_status will be ''cache-hit'' or ''cache-miss''.

IPV6.066

Level: UE-ERROR

Short Syntax: IPV6.066 LinkAddr *source_ipv6_address* -> *destination_ipv6_address*, discarded

Long Syntax: IPV6.066 Received link local address from *source_ipv6_address* for *destination_ipv6_address*, discarded

Description: Router never relays an IPV6 with either link local source or dest. address

---

IPV6.067

Level: U-INFO

Short Syntax: IPV6.067 Route *destination_ip_address/ mask status*

Long Syntax: IPV6.067 Route for *destination_ip_address* with mask *mask status*

Description: The route has been filtered from the IP route table or installed as a hidden route due to route table filtering policy.

IPV6.068

Level: UE-ERROR

Short Syntax: IPV6.068 mtu < 1280 for nt *network ID*

Long Syntax: IPV6.068 The MTU is less than 1280 bytes for net *network ID*

Description: The net has an MTU that is less than the minimum required to run IPV6 (1280 bytes).

Action: Correct the configuration of this net to make the MTU at least 1280 bytes or do not configure IPV6 on this net.

IPV6.069

Level: UI-ERROR

Short Syntax: IPV6.069 tnl *tunnel_id*, int *interface* is not IP64

Long Syntax: IPV6.069 Tunnel *tunnel_id* has a virtual interface *interface* which is not an IP64 tunnel

Description: The configured tunnel is pointing to an interface which is not an IP64 tunnel. This is caused by loading a prior release on the box with an IP64 tunnel interface defined. This tunnel will not be installed.

Action: Delete the tunnel definition. Then re-configure the tunnel.

IPV6.070

Level: P_TRACE

Short Syntax: IPV6.070 Trace IPV6 datagram.

Long Syntax: IPV6.070 Trace IPV6 datagram.

Description: IPV6 datagram tracing.

IPV6.071

Level: P_TRACE

Short Syntax: IPV6.071 Trace in IPV6 datagram at DLC.

Long Syntax: IPV6.071 Trace incoming IPV6 datagram at DLC.

Description: Incoming IPV6 datagram DLC tracing.

IPV6.072

Level: C-TRACE

Short Syntax: IPV6.072 Acc cont # *record_number* dropped, *cache_status*, *source_ipv6_address* -> *destination_ipv6_address*, *sport* port *dport* -> *direction*, dir *autotunn*, *networkID*net

Long Syntax: IPV6.072 Access control number *record_number* dropped, *cache_status*, packet from *source_ipv6_address* to *destination_ipv6_address*, *sport* port *dport* -> *direction*, direction *autotunn*, *networkID*net

Description: This message is generated when a IPV6 packet matches one of the exclusive access control entries. The packet will be dropped. The record_number is the number of the access control record matched, or zero for no record (end-of-list). The cache_status will be ''cache-hit'' or ''cache-miss''.

IPV6.073

Level: C-TRACE

Short Syntax: IPV6.073 Acc cont # *record_number* passed, *cache_status*, *source_ipv6_address* -> *destination_ipv6_address*, *sport* port *dport* -> *direction*, dir *autotunn*, *networkID*net

Long Syntax: IPV6.073 Access control number *record_number* passed, *cache_status*, packet from *source_ipv6_address* to *destination_ipv6_address*, *sport* port *dport* -> *direction*, direction *autotunn*, *networkID*net

Description: This message is generated when a IPV6 packet matches one of the inclusive access control entries. The packet may be forwarded. The record_number is the number of the access control record matched. The cache_status will be ''cache-hit'' or ''cache-miss''.

IPV6.074

Level: U-INFO

Short Syntax: IPV6.074 pkt *ip4src* -> *ip4dest*, *ip6src* -> *ip6dest* dsc on tun(*tunint*): *reason*

Long Syntax: IPV6.074 Packet from IPv4 *ip4src* to *ip4dest*, IPv6 *ip6src* to *ip6dest* was discarded on tunnel(interface *tunint*): *reason*

Description: An IPv6 packet, encapsulated in an IPv4 header, has been discarded on the specified tunnel interface for the specified reason.

Panic in6initnomem

Short Syntax: IPV6: no mem for init

Description: The router ran out of memory during IPV6 initialization.

Action: Add memory, or reconfigure to reduce memory usage (for example, reduce the size of the IPV6 routing table, or reduce the number of IPV6 addresses).

Chapter 57. IP Protocol Network (IPPN)

This chapter describes IP Protocol Network (IPPN) messages. For information on message content and how to use the message, refer to the Introduction.

IPPN.002

Level: U-INFO

Short Syntax: IPPN.002 SRT *src_IP*-> *dst_IP* (UDP *src_port*-> *dst_port*) ign, no bdg on tunnel

Long Syntax: IPPN.002 SRT packet from *src_IP* to *dst_IP* (from UDP socket *src_port* to *dst_port*) ignored, no bridging on tunnel

Description: A IP packet was received for one of the SRT tunnel's UDP ports, but bridging is not enabled on the tunnel. The packet will be discarded.

Cause: Another bridge is configured to have this bridge as a participant in a SRT bridging tunnel, but this bridge is not so configured.

Action: Make configuration consistent.

Cause: Some other application on the IP network is sending packets to one of the SRT tunnel UDP ports on this router.

Action: Either change application, or ignore this message.

IPPN.003

Level: P-TRACE

Short Syntax: IPPN.003 SRT *src_IP*-> *dst_IP* (UDP *src_port*-> *dst_port*) ign, port blocked

Long Syntax: IPPN.003 SRT packet from *src_IP* to *dst_IP* (from UDP socket *src_port* to *dst_port*) ignored, port blocked

Description: A IP packet (which was not a BPDU) was received for one of the SRT tunnel's UDP ports, but that port is in "blocking" or "listening" state. The packet will be discarded.

Cause: Perfectly normal when one port into the tunnel blocks. However, ports to the tunnel will not ordinarily block unless there is an alternative bridging path in parallel with the tunnel.

IPPN.004

Level: P-TRACE

Short Syntax: IPPN.004 Old SRB *src_IP*-> *dst_IP* (UDP *src_port*-> *dst_port*), compat mode

Long Syntax: IPPN.004 Old SRB packet from *src_IP* to *dst_IP* (from UDP socket *src_port* to *dst_port*), in compatability mode

Description: A packet has been received from a node participating in the IP tunnel which is using the SRB tunnel encapsulation used prior to Release 12.0. This packet will be processed normally, but the tunnel will now remain in the mode compatible with the old encapsulation. This means that FCS will never be preserved for 802.5 frames across the tunnel.

Cause: Node running SRB tunnel software from before Release 12.0.

Action: Update all participants in tunnel to Release 12.0, and you will not get this message, and will be able to preserve 802.5 FCS across the tunnel.

Panic ippnudpregfail

Short Syntax: IPPN UDP socket registration failure

Description: The IPPN protocol net was unable to register one of the UDP sockets it requires with the UDP protocol.

Cause: Bug in software.

Action: Contact customer service.

Chapter 58. IP Security Protocol (IPsec)

This chapter describes IP Security Protocol (IPsec) messages. For information on message content and how to use the message, refer to the Introduction.

IPSP.001

Level: U-INFO

Short Syntax: IPSP.001 IPsec init

Long Syntax: IPSP.001 IPsec initialization

Description: This message is printed when IPsec is going through initialization.

IPSP.002

Level: UI-ERROR

Short Syntax: IPSP.002 IPsec unable to get mem

Long Syntax: IPSP.002 IPsec unable to get memory

Description: IPsec was unable to allocate the necessary memory. IPsec is unable to run because of this.

Cause: There is a shortage in heap memory, possibly because too many memory intensive forwarders/protocols are running.

Action: Disable unnecesary forwarders/protocols or get more memory.

IPSP.003

Level: U-INFO

Short Syntax: IPSP.003 q ovrf *source_ip_address* -> *destination_ip_address* nt *network ID*

Long Syntax: IPSP.003 Queue overflow on packet from *source_ip_address* for *destination_ip_address* from net *network ID*

Description: This message is generated when the IP forwarder must discard a packet that was to be secured because of an IPsec input queue overflow.

Cause: IPsec input queue overflows happen when a packet is received from an interface that is short on buffers. Length of the IPsec queue is greater than the fair share. This may be caused by either a burst or steady state of traffic arriving faster than the IP forwarder can encaped(Secured) it.

Action: Reduce traffic bursts. Upgrade to a faster router.

IPSP.004

Level: P-TRACE

Short Syntax: IPSP.004 rcv pkt for encap *source_ip_address* -> *destination_ip_address* wth tid *tunnel_id*

Long Syntax: IPSP.004 Accepting packet for encapsulation from *source_ip_address* to *destination_ip_address* with tunnel_id *tunnel_id*

Description: This message is generated for each IP packet which is passing through the IPsec encapsulation module.

IPSP.005

Level: P-TRACE

Short Syntax: IPSP.005 rcv pkt for decap *source_ip_address* -> *destination_ip_address*

Long Syntax: IPSP.005 Accepting packet for decapsulation from *source_ip_address* to *destination_ip_address*

Description: This message is generated for each IP packet which is passing through the IPsec decapsulation module.

IPSP.006

Level: U-INFO

Short Syntax: IPSP.006 dsc IPsec pkt *source_ip_address* -> *destination_ip_address* nt *Network ID* no IPsec

Long Syntax: IPSP.006 Discarded IPsec packet from *source_ip_address* for *destination_ip_address* net *Network ID*, IPsec not enabled.

Description: This message is generated when an IP packet containing an IPsec protocol header is received and IPsec is not enabled. The packet is dropped since there are no active IPsec tunnels available to decapsulate the contents of the IPsec packet.

Cause: Received an IPsec protocol packet, but IPsec is not enabled.

IPSP.007

Level: UI-ERROR

Short Syntax: IPSP.007 IPsec *function_name*: tunl *tunnel_id* not active

Long Syntax: IPSP.007 IPsec *function_name*: tunnel *tunnel_id* is not active.

Description: An IP packet could not be secured because the designated tunnel is not active. The packet has been dropped.

IPSP.008

Level: UE-ERROR

Short Syntax: IPSP.008 addr msmtch IP src *pkt_src_addr* tunl src *tunl_src_addr* IP dst *pkt_dst_addr* tunl dst *tunl_dst_addr* tunl *tunnel_id*

Long Syntax: IPSP.008 address mismatch for transport mode tunnel - IP packet source address *pkt_src_addr*, tunnel source address *tunl_src_addr*, IP packet destination address *pkt_dst_addr*, tunnel destination address *tunl_dst_addr*, tunnel *tunnel_id*

Description: In transport mode, there is a mismatch in the IP packet addresses and the secure tunnel IP addresses.

IPSP.009

Level: CI-ERROR

Short Syntax: IPSP.009 *error_message* tunnl *tunnel_id*

Long Syntax: IPSP.009 Error: *error_message* tunnel *tunnel_id*

Description: There is an error as indicated by the error message.

IPSP.010

Level: UE-ERROR

Short Syntax: IPSP.010 pkt too short: pkt len *length* hdr len *header_len*

Long Syntax: IPSP.010 Packet too short: packet len *length* header len *header_len*

Description: An IPsec packet was received with a payload that was less than 8 bytes long.

IPSP.011

Level: P-TRACE

Short Syntax: IPSP.011 esp encap in *mode* mode alg *algorithm* tunl *tunnel_id*

Long Syntax: IPSP.011 esp encapsulation in *mode* mode algorithm *algorithm* tunnel *tunnel_id*

Description: An IP packet is being encapsulated using the IPsec Encapsulating Security Payload (ESP).

IPSP.012

Level: P-TRACE

Short Syntax: IPSP.012 esp encap with pad len *pad_length* spi *SPI* iv *IV_1 IV_2* tunl *tunnel_id*

Long Syntax: IPSP.012 esp encapsulation with pad length *pad_length* security parameter index *SPI* initialization vector *IV_1 IV_2* tunnel *tunnel_id*

Description: An IPsec ESP packet has been constructed.

IPSP.013

Level: P-TRACE

Short Syntax: IPSP.013 Module *trc_msg*

Long Syntax: IPSP.013 Module *trc_msg*

Description: This message is for internal informational purposes.

IPSP.014

Level: P-TRACE

Short Syntax: IPSP.014 esp decap with alg *algorithm* tunl *tunnel_id*

Long Syntax: IPSP.014 esp decapsulation with algorithm *algorithm* tunnel *tunnel_id*

Description: An IP packet containing the IPsec Encapsulating Security Payload (ESP) was received.

IPSP.015

Level: UE-ERROR

Short Syntax: IPSP.015 ESP decap: bad payload len *payload_length* tunl *tunnel_id*

Long Syntax: IPSP.015 ESP decapsulation: bad payload length *payload_length* tunnel *tunnel_id*

Description: An IPsec ESP packet was received that had an invalid payload length (lacked the proper payload padding).

IPSP.016

Level: UE-ERROR

Short Syntax: IPSP.016 ESP decap: bad payload len *payload_len* - pad len *padding_length* tunl *tunnel_id*

Long Syntax: IPSP.016 ESP decapsulation: bad payload length *payload_len* for padding length *padding_length* tunnel *tunnel_id*

Description: The payload length of an IPsec ESP packet is not correct since it is shorter than, or equal to, the padding length.

IPSP.017

Level: P-TRACE

Short Syntax: IPSP.017 ah encap in *mode* mode alg *algorithm* tunl *tunnel_id*

Long Syntax: IPSP.017 ah encapsulation in *mode* mode algorithm *algorithm* tunnel *tunnel_id*

Description: An IP packet is being encapsulated using the IPsec Authentication Header (AH).

IPSP.018

Level: P-TRACE

Short Syntax: IPSP.018 ah decap with alg *algorithm* tunl *tunnel_id*

Long Syntax: IPSP.018 ah decapsulation with algorithm *algorithm* tunnel *tunnel_id*

Description: An IP packet containing the IPsec Authentication Header (AH) was received.

IPSP.019

Level: UE-ERROR

Short Syntax: IPSP.019 AH decap: bad packet len *payload_len* tunl *tunnel_id*

Long Syntax: IPSP.019 AH decapsulation: bad packet length *payload_len* tunnel *tunnel_id*

Description: An IPsec AH packet was received that had an invalid payload length.

IPSP.020

Level: UI-ERROR

Short Syntax: IPSP.020 Module Decap: no tunl for src *src_addr* dst *dst_addr* spi *spi*

Long Syntax: IPSP.020 Module Decap: no active tunnel list entry for source address *src_addr*, destination address *dst_addr*, and security parameter index *spi*

Description: There was no active tunnel list entry for the IPsec packet received.

IPSP.021

Level: UI-ERROR

Short Syntax: IPSP.021 Init: init error for tunn ID *tunnel_id*, errcode= *error_code*

Long Syntax: IPSP.021 IPsec initialization: initialization error for tunnel ID *tunnel_id*, error code = *error_code*.

Description: An IPsec initialization error occurred. Save configuration file, record error code, and contact Customer Service.

IPSP.022

Level: U-INFO

Short Syntax: IPSP.022 tunl list add tunl *tunnel_id* - *reason*

Long Syntax: IPSP.022 An active tunnel list entry was added for tunnel ID *tunnel_id* - reason is *reason*.

Description: An entry in the active tunnel list was added.

IPSP.023

Level: U-INFO

Short Syntax: IPSP.023 tunl list del tunl *tunnel_id* - *reason*

Long Syntax: IPSP.023 An active tunnel list entry was deleted for tunnel ID *tunnel_id* - reason is *reason*.

Description: An entry in the active tunnel list was deleted.

IPSP.024

Level: U-INFO

Short Syntax: IPSP.024 IPsec enabled from console

Long Syntax: IPSP.024 The IPsec feature was enabled from the console.

Description: The IPsec feature was enabled from the console by the ENABLE IPSEC command.

IPSP.025

Level: U-INFO

Short Syntax: IPSP.025 IPsec disabled from console - *disable_mode* mode

Long Syntax: IPSP.025 The IPsec feature was disabled from the console. Disable mode is *disable_mode*.

Description: The IPsec feature was disabled from the console by the DISABLE IPSEC command.

IPSP.026

Level: UI-ERROR

Short Syntax: IPSP.026 IPsec Encryption Algorithm *which_esp* is not allowed on this tun id *tun_id*.

Long Syntax: IPSP.026 IPsec Encryption Algorithm *which_esp* is not allowed on this tunnel id *tun_id*.

Description: The configured ESP algorithm is not available on this router library.

IPSP.027

Level: P-TRACE

Short Syntax: IPSP.027 rcv pkt for encap *source_ip_address* -> *destination_ip_address* wth tid *tunnel_id*

Long Syntax: IPSP.027 Accepting packet for encapsulation from *source_ip_address* to *destination_ip_address* with tunnel_id *tunnel_id*

Description: This message is generated for each IPv6 packet which is passing through the IPsec encapsulation module.

IPSP.028

Level: UE-ERROR

Short Syntax: IPSP.028 addr msmtch IP src *pkt_src_addr* tunl src *tunl_src_addr* IP dst *pkt_dst_addr* tunl dst *tunl_dst_addr* tunl *tunnel_id*

Long Syntax: IPSP.028 address mismatch for transport mode tunnel - IP packet source address *pkt_src_addr*, tunnel source address *tunl_src_addr*, IP packet destination address *pkt_dst_addr*, tunnel destination address *tunl_dst_addr*, tunnel *tunnel_id*

Description: In transport mode, there is a mismatch in the IPv6 packet addresses and the secure tunnel IP addresses.

IPSP.029

Level: UI-ERROR

Short Syntax: IPSP.029 Module Decap: no tunl for src *src_addr* dst *dst_addr* spi *spi*

Long Syntax: IPSP.029 Module Decap: no active tunnel list entry for source address *src_addr*, destination address *dst_addr*, and security parameter index *spi*

Description: There was no active tunnel list entry for the IPsec packet received.

IPSP.030

Level: U-INFO

Short Syntax: IPSP.030 q ovrf *source_ip_address* -> *destination_ip_address* nt *network ID*

Long Syntax: IPSP.030 Queue overflow on packet from *source_ip_address* for *destination_ip_address* from net *network ID*

Description: This message is generated when the IP forwarder must discard a packet that was to be secured because of an IPsec input queue overflow.

Cause: IPsec input queue overflows happen when a packet is received from an interface that is short on buffers. Length of the IPsec queue is greater than the fair share. This may be caused by either a burst or steady state of traffic arriving faster than the IP forwarder can encaped(Secured) it.

Action: Reduce traffic bursts. Upgrade to a faster router.

IPSP.031

Level: U-INFO

Short Syntax: IPSP.031 dsc IPsec pkt *source_ip_address* -> *destination_ip_address* nt *Network ID* no IPsec

Long Syntax: IPSP.031 Discarded IPsec packet from *source_ip_address* for *destination_ip_address* net *Network ID*, IPsec not enabled.

Description: This message is generated when an IPv6 packet containing an IPsec protocol header is received and IPsec is not enabled. The packet is dropped since there are no active IPsec tunnels available to decapsulate the contents of the IPsec packet.

Cause: Received an IPsec protocol packet, but IPsec is not enabled.

IPSP.032

Level: P-TRACE

Short Syntax: IPSP.032 rcv pkt for decap *source_ip_address* -> *destination_ip_address*

Long Syntax: IPSP.032 Accepting packet for decapsulation from *source_ip_address* to *destination_ip_address*

Description: This message is generated for each IPv6 packet which is passing through the IPsec decapsulation module.

IPSP.033

Level: U-INFO

Short Syntax: IPSP.033 pkt bigger than PMTU *source_ip_address* -> *destination_ip_address*, pmtu *pmtu*, pkt size *pktsize*

Long Syntax: IPSP.033 Packet bigger than PMTU from *source_ip_address* to *destination_ip_address*, pmtu is *pmtu*, packet size is *pktsize*

Description: This message is generated for IPv6 packets that are being sent over a secure tunnel mode tunnel that are larger than the Path MTU of that tunnel. An ICMP "packet too big" message will be generated back to the host.

IPSP.034

Level: U-TRACE

Short Syntax: IPSP.034 tunnel *tunnel_id* aged, pmtu *mtu*

Long Syntax: IPSP.034 Tunnel *tunnel_id* aged out of table, path MTU *mtu*

Description: The path MTU aging timer has expired for the specified tunnel. The path MTU will be reset to the maximum MTU value and path MTU discovery will be started on the next packet to the traverse this tunnel.

IPSP.035

Level: U-TRACE

Short Syntax: IPSP.035 pkt too big for tunnel *tunnel_id*, pmtu *mtu*

Long Syntax: IPSP.035 Packet Too Big ICMP message received for tunnel *tunnel_id*, path MTU is *mtu*

Description: A packet too big message has been received for a packet originated by this router on the specified tunnel. Path MTU Discovery will start for this tunnel.

IPSP.036

Level: UI-ERROR

Short Syntax: IPSP.036 Path MTU: no tunl for src *src_addr* dst *dst_addr* spi *spi*

Long Syntax: IPSP.036 Path MTU: no active tunnel list entry for source address *src_addr*, destination address *dst_addr*, and security parameter index *spi*

Description: There was no active tunnel list entry for the ICMP Packet Too Big packet received.

IPSP.037

Level: UI-ERROR

Short Syntax: IPSP.037 no mem for pmtu disc for *tunnel_id*

Long Syntax: IPSP.037 There is no memory available to perform Path MTU Discovery for *tunnel_id*

Description: There is not enough memory in the router to allocate the control blocks necessary for Path MTU Discovery for packets on the specified tunnel.

IPSP.038

Level: UI-ERROR

Short Syntax: IPSP.038 Path MTU: no tunl for src *src_addr* dst *dst_addr* spi *spi*

Long Syntax: IPSP.038 Path MTU: no active tunnel list entry for source address *src_addr*, destination address *dst_addr*, and security parameter index *spi*

Description: There was no active tunnel list entry for the ICMP Packet Too Big packet received.

IPSP.039

Level: U-INFO

Short Syntax: IPSP.039 pkt bigger than PMTU *source_ip_address* -> *destination_ip_address*, pmtu *pmtu*, pkt size *pktsize*

Long Syntax: IPSP.039 Packet bigger than PMTU from *source_ip_address* to *destination_ip_address*, pmtu is *pmtu*, packet size is *pktsize*

Description: This message is generated for IPv4 packets that are being sent over a secure tunnel mode tunnel that are larger than the Path MTU of that tunnel and the DF bit is set in the outer header. An ICMP ″packet too big″ message will be generated back to the host.

IPSP.040

Level: U-INFO

Short Syntax: IPSP.040 df bit not copied/set, sec pkt bigger than minimum, *source_ip_address* -> *destination_ip_address*, tnl *tunnel*

Long Syntax: IPSP.040 df bit in the outer header cannot be copied/set, the secured packet is greater than the minimum MTU, *source_ip_address* to *destination_ip_address*, tunnel *tunnel*

Description: This message is generated for IPv4 packets that are being sent over a secure tunnel mode tunnel. After the packet is secured, it is larger than the Path MTU of that tunnel. However, the incoming packet was less than or equal to the minimum MTU of 576, so an ICMP error message will not lower the size of the incoming packets. The configuration for this tunnel has the DF bit in the outer header being copied from the inner header or set, but this will not occur, since the packet must be allowed to fragment. The DF bit in the outer header will not be set.

IPSP.041

Level: P-TRACE

Short Syntax: IPSP.041 *str_message* alg *algorithm* tunl *tunnel_id*

Long Syntax: IPSP.041 *str_message* with algorithm *algorithm* tunnel *tunnel_id*

Description: Performing ESP authentication on a packet

IPSP.042

Level: P-TRACE

Short Syntax: IPSP.042 Add P2 tunl *tunnel_id* by IKE is *msg*

Long Syntax: IPSP.042 Add phase 2 tunnel *tunnel_id* by IKE is *msg*

Description: Result of adding a phase 2 tunnel by ISAKMP

IPSP.043

Level: P-TRACE

Short Syntax: IPSP.043 tunl *tunnel_id* SA *msg* thrhld rchd. Strt rfrsh.

Long Syntax: IPSP.043 Tunnel *tunnel_id* SA *msg* threshold reached. Start refreshing.

Description: A condition is met, start refreshing the SA.

IPSP.044

Level: CI-ERROR

Short Syntax: IPSP.044 *error_message*

Long Syntax: IPSP.044 Error: *error_message*

Description: There is an error as indicated by the error message.

IPSP.045

Level: P-TRACE

Short Syntax: IPSP.045 Rfrshng SA for P2 tunl *tunnel_id* by IKE is *msg*

Long Syntax: IPSP.045 Refreshing SA for phase 2 tunnel *tunnel_id* by IKE is *msg*

Description: Result of refreshing SA for a phase 2 tunnel by ISAKMP

IPSP.046

Level: P-TRACE

Short Syntax: IPSP.046 *msg tunnel_id*

Long Syntax: IPSP.046 *msg tunnel_id*

Description: General tracing message

IPSP.047

Level: CI-ERROR

Short Syntax: IPSP.047 Conformance check failed from *from_addr* to *to_addr* tunn *tun_id*

Long Syntax: IPSP.047 Conformance check failed from *from_addr* to *to_addr* tunn *tun_id*

Description: Conformance check receiving packet failed.

Chapter 59. ISAKMP

This chapter describes ISAKMP messages. For information on message content and how to use the message, refer to the Introduction.

IKE.001

Level: P_TRACE

Short Syntax: IKE.001 Trace IKE packet *string ipaddress*

Long Syntax: IKE.001 Trace IKE packet *string ipaddress*

Description: ISAKMP packet tracing.

IKE.002

Level: P_TRACE

Short Syntax: IKE.002 Trace IKE payload *string* Peer: *ipaddress*

Long Syntax: IKE.002 Trace IKE payload *string* Peer: *ipaddress*

Description: ISAKMP packet tracing.

IKE.003

Level: U-INFO

Short Syntax: IKE.003 Processing *messageID*

Long Syntax: IKE.003 Processing *messageID*

Description: Indicates which message is being processed.

IKE.004

Level: UI-ERROR

Short Syntax: IKE.004 Error *errorMessage*

Long Syntax: IKE.004 Error *errorMessage*

Description: Error message

IKE.005

Level: UI-ERROR

Short Syntax: IKE.005 Error *errorMessage integer*

Long Syntax: IKE.005 Error *errorMessage integer*

Description: Error message with one integer parameter

IKE.006

Level: UI-ERROR

Short Syntax: IKE.006 Error *errorMessage integer integer*

Long Syntax: IKE.006 Error *errorMessage integer integer*

Description: Error message with two integer parameters

IKE.007

Level: UI-ERROR

Short Syntax: IKE.007 IKE unable to get mem: *location*

Long Syntax: IKE.007 IKE unable to get memory: *location*

Description: IKE was unable to allocate the necessary memory IKE is unable to run because of this.

Cause: There is a shortage in heap memory, possibly because too many memory intensive forwarders/protocols are running.

Action: Disable unnecesary forwarders/protocols or get more memory.

IKE.008

Level: U-INFO

Short Syntax: IKE.008 Load *direction* SA: Alg= *algorithm* Prot= *protocol* Sec= *lifeSeconds* KB= *lifeSize* SPI= *spi*

Long Syntax: IKE.008 Load Ipsec *direction* SA: Alg= *algorithm* Proto= *protocol* LifeSec= *lifeSeconds* LifeKB= *lifeSize* SPI= *spi*

Description: Notification of injection of ipsec sa by ike

IKE.009

Level: U-INFO

Short Syntax: IKE.009 Begin

Long Syntax: IKE.009 Begin

Description: Indicates which state or function is being entered.

IKE.010

Level: U-INFO

Short Syntax: IKE.010 Finished

Long Syntax: IKE.010 Finished

Description: Indicates which state is being exited.

IKE.011

Level: U-INFO

Short Syntax: IKE.011 *Message*

Long Syntax: IKE.011 *Message*

Description: Info message with one string parameter

IKE.012

Level: UI-ERROR

Short Syntax: IKE.012 Error *errorMessage long*

Long Syntax: IKE.012 Error *errorMessage long*

Description: Error message with one long parameter

IKE.013

Level: U-INFO

Short Syntax: IKE.013 *Info*

Long Syntax: IKE.013 *Info*

Description: Info message for string variable

IKE.014

Level: U-INFO

Short Syntax: IKE.014 *Message address*

Long Syntax: IKE.014 *Message address*

Description: Info message with one address parameter

IKE.015

Level: U-INFO

Short Syntax: IKE.015 *Info Message*

Long Syntax: IKE.015 *Info Message*

Description: Info message with one integer parameter

IKE.016

Level: UI-ERROR

Short Syntax: IKE.016 *Info Message*

Long Syntax: IKE.016 *Info Message*

Description: Info message with one string parameter

IKE.017

Level: U-INFO

Short Syntax: IKE.017 *Info message long*

Long Syntax: IKE.017 *Info message long*

Description: Info message with two long parameters

IKE.018

Level: U-INFO

Short Syntax: IKE.018 IKE Public Key module init *status*. Exit point: *fun_name code*

Long Syntax: IKE.018 IKE Public Key module init *status*. Exit point: *fun_name code*

Description: The status of IKE public key module initialization

IKE.019

Level: U-INFO

Short Syntax: IKE.019 IKE signature private key signing *status*

Long Syntax: IKE.019 IKE signature private key signing *status*

Description: The status of IKE signature private key sign

IKE.020

Level: U-INFO

Short Syntax: IKE.020 IKE signature public key verification *status*

Long Syntax: IKE.020 IKE signature public key verification *status*

Description: The status of IKE signature public key verification

IKE.021

Level: U-INFO

Short Syntax: IKE.021 IKE signature *status*

Long Syntax: IKE.021 IKE signature *status*

Description: The status of IKE signature matching

IKE.022

Level: U-INFO

Short Syntax: IKE.022 *Info message string integer string integer*

Long Syntax: IKE.022 *Info message string integer string integer*

Description: Info message

IKE.023

Level: U-INFO

Short Syntax: IKE.023 *Info message integer*

Long Syntax: IKE.023 *Info message integer*

Description: Info message with two integer parameters

IKE.024

Level: UI-ERROR

Short Syntax: IKE.024 *errorMessage address*

Long Syntax: IKE.024 *errorMessage address*

Description: error message with one address parameter

IKE.025

Level: U-INFO

Short Syntax: IKE.025 *Info message*

Long Syntax: IKE.025 *Info message*

Description: Info message with one long parameters

IKE.026

Level: U-INFO

Short Syntax: IKE.026 Received *Info* notify message from Peer: *message*

Long Syntax: IKE.026 Received *Info* notify message from Peer: *message*

Description: Info message indicating that a notify message was received and lists it's type

Chapter 60. ISDN Coordinating and Management Entity (CEME)

This chapter describes ISDN Coordinating and Management Entity (CEME) messages. For information on message content and how to use the message, refer to the Introduction.

CEME.001

Level: U-INFO

Short Syntax: CEME.001 START_RQ recvd switch type = *switch* on isdn/ *intf*

Long Syntax: CEME.001 Request to initiate L2 for switch *switch* on network *intf*

Description: Request to initiate L2 and tei request for this interface

Action: None

CEME.003

Level: U-INFO

Short Syntax: CEME.003 RELEASE Layer 3 prim=0x *prim* on nt isdn/ *intf*

Long Syntax: CEME.003 layer 3 Release received primitive=0x *prim* on ISDN/ *intf*

Description: Release all the calls for this interface and inform l2

Action: None

CEME.004

Level: U-INFO

Short Syntax: CEME.004 Establish request received, primitive 0x *prim* on isdn/ *intf*

Long Syntax: CEME.004 Establish layer 3 primitive value (0x *prim*) on network *intf*

Description: Establish layer 3 start D-Channel Communication

Action: None

CEME.005

Level: U-INFO

Short Syntax: CEME.005 TEI *tei* REMOVED no response from network on isdn/ *intf*

Long Syntax: CEME.005 tei time out, no network response for *tei* on isdn *intf*

Description: Remove TEI

Action: None

CEME.006

Level: U-INFO

Short Syntax: CEME.006 TEI *tei* REMOVED by request on isdn/ *intf*

Long Syntax: CEME.006 tei *tei* has been removed by request on isdn *intf*

Description: Remove TEI

Action: None

Panic cemeym

Short Syntax: YDC ISDN: mem alloc fld

Description: The YDC ISDN network handler failed to allocate sufficient memory during the initialization phase.

Action: Contact customer service.

Chapter 61. ISDN LAYER 2 Link Access Procedure on the D Channel (LAPD)

This chapter describes ISDN LAYER 2 Link Access Procedure on the D Channel (LAPD) messages. For information on message content and how to use the message, refer to the Introduction.

LAPD.001

Level: U-INFO

Short Syntax: LAPD.001 SABME recvd on isdn/ *intf*

Long Syntax: LAPD.001 Request to initiate Asynchronous balanced mode on isdn/ *intf*

Description: Start connection oriented Layer 2 services

Action: None

LAPD.002

Level: U-INFO

Short Syntax: LAPD.002 SABME sent on isdn/ *intf*

Long Syntax: LAPD.002 Request to initiate Asynchronous balanced mode on isdn/ *intf*

Description: Start connection oriented Layer 2 services

Action: None

LAPD.003

Level: U-INFO

Short Syntax: LAPD.003 UA recvd on isdn/ *intf*

Long Syntax: LAPD.003 Response to SABME/ DISC initiate/terminate Asynchronous balanced mode on isdn/ *intf*

Description: Start/stop connection oriented Layer 2 services

Action: None

LAPD.004

Level: U-INFO

Short Syntax: LAPD.004 UA sent on isdn/ *intf*

Long Syntax: LAPD.004 Respond to request to initiate/terminate Asynchronous balanced mode on isdn/ *intf*

Description: Start/stop connection oriented Layer 2 services

Action: None

LAPD.005

Level: U-INFO

Short Syntax: LAPD.005 L2-DISC recv on isdn/ *intf*

Long Syntax: LAPD.005 Layer 2 disconnect received to terminate Asynchronous balanced mode on isdn/ *intf*

Description: Stop connection oriented Layer 2 services

Action: None

LAPD.006

Level: U-INFO

Short Syntax: LAPD.006 DM recv on isdn/ *intf*

Long Syntax: LAPD.006 Disconnect Mode (DM) recv terminate Asynchronous balanced mode on isdn/ *intf*

Description: Stop connection oriented Layer 2 services

Action: None

Panic lapdym

Short Syntax: YDC ISDN: mem alloc fld

Description: The YDC ISDN network handler failed to allocate sufficient memory during the initialization phase.

Action: Contact customer service.

Chapter 62. ISO OSI Connectionless Network Layer (ISO)

This chapter describes ISO OSI Connectionless Network Layer (ISO) messages. For information on message content and how to use the message, refer to the Introduction.

ISO.001

Level: UE-ERROR

Short Syntax: ISO.001 rcvd incmplt pkt

Long Syntax: ISO.001 received incomplete packet

Description: A packet fragment recognized as an ISO CLNP data packet was received.

ISO.002

Level: UE-ERROR

Short Syntax: ISO.002 rcvd pkt bad NSAP len (= *length*)

Long Syntax: ISO.002 received packet with a bad NSAP length (= *length*)

Description: An ISO CLNP data packet was received with an illegal NSAP length.

ISO.003

Level: UE-ERROR

Short Syntax: ISO.003 rcvd pkt bad chksm = *pkt_chksum*

Long Syntax: ISO.003 received packet with a bad checksum = *pkt_chksum*

Description: An ISO CLNP data packet was received but had a bad checksum.

ISO.004

Level: UE-ERROR

Short Syntax: ISO.004 rcvd pkt bad vers # = *version_number*

Long Syntax: ISO.004 received packet with a bad version number (vers = *version_number*)

Description: An ISO CLNP data packet was received but had a bad or unsupported version number.

ISO.005

Level: UE-ERROR

Short Syntax: ISO.005 rcvd pkt bad typ # = *type_field*

Long Syntax: ISO.005 received packet with a bad type field (vers = *type_field*)

Description: An ISO CLNP data packet was received but had a bad or unsupported type field.

ISO.006

Level: UE-ERROR

Short Syntax: ISO.006 rcvd pkt life exp *source_NSAP -> destination_NSAP*

Long Syntax: ISO.006 received packet with an expired liftime *source_NSAP -> destination_NSAP*

Description: An ISO CLNP data packet was received but had a bad checksum.

ISO.007

Level: UE-ERROR

Short Syntax: ISO.007 rcvd pkt bad opt *source_NSAP -> destination_NSAP*

Long Syntax: ISO.007 received packet with a bad optional parameter *source_NSAP -> destination_NSAP*

Description: An ISO CLNP data packet was received with a bad optional parameter.

ISO.008

Level: UE-ERROR

Short Syntax: ISO.008 rcvd pkt dest unkwn *source_NSAP -> destination_NSAP*

Long Syntax: ISO.008 received packet - destination unknown *source_NSAP -> destination_NSAP*

Description: An ISO CLNP data packet is received but can not be routed since there is no routing table entry for destination.

ISO.009

Level: UE-ERROR

Short Syntax: ISO.009 rcvd pkt no seg prmit *source_NSAP -> destination_NSAP*

Long Syntax: ISO.009 received packet-no segmentation permitted *source_NSAP -> destination_NSAP*

Description: An ISO CLNP data packet was received

which needed segmentation, but the segmentation permitted flag was not set.

ISO.010

Level: UE-ERROR

Short Syntax: ISO.010 rcvd pkt cnnt fwd *source_NSAP* -> *destination_NSAP* hndlr err (= *error_code*)

Long Syntax: ISO.010 received packet cannot forward, handler error *source_NSAP* -> *destination_NSAP* (err= *error_code*)

Description: An ISO CLNP data packet was received and routed but couldn't be forwarded because of a handler error.

ISO.011

Level: UE-ERROR

Short Syntax: ISO.011 CLNP input que ovflw *source_NSAP* -> *destination_NSAP*

Long Syntax: ISO.011 CLNP input queue overflow *source_NSAP* -> *destination_NSAP*

Description: The ISO CLNP input packet queue has overflowed. Packet is dropped.

ISO.012

Level: UE-ERROR

Short Syntax: ISO.012 no iob avail to snd err pkt

Long Syntax: ISO.012 no i/o buffer available to send error packet

Description: An attempt to send an ISO CLNP error packet failed because of a lack of system i/o buffers.

ISO.014

Level: P-TRACE

Short Syntax: ISO.014 rcvd pkt *source_NSAP* -> *destination_NSAP*

Long Syntax: ISO.014 received packet *source_NSAP* -> *destination_NSAP*

Description: An ISO CLNP data packet was received and passed error checking.

ISO.015

Level: UE-ERROR

Short Syntax: ISO.015 cnnt fwd err pkt hndlr err (= *error_code*) *source_NSAP* -> *destination_NSAP*

Long Syntax: ISO.015 cannot forward an error packet, handler error (err= *error_code*) *source_NSAP* -> *destination_NSAP*

Description: An ISO CLNP error packet coundn't be forwarded because of a handler error.

ISO.017

Level: UE-ERROR

Short Syntax: ISO.017 OSI unknwn init prot id

Long Syntax: ISO.017 OSI unknown initial protocol identifier

Description: An ISO CLNP packet has been received with an unknown or unsupported initial protocol identifier.

ISO.018

Level: P-TRACE

Short Syntax: ISO.018 rcvd ERR pkt *source_NSAP* -> *destination_NSAP* cd= *error_code*

Long Syntax: ISO.018 received Error packet *source_NSAP* -> *destination_NSAP* code = *error_code*

Description: An ISO CLNP Error packet was received for this router.

ISO.019

Level: UE-ERROR

Short Syntax: ISO.019 rcvd DT loc *source_NSAP* -> *destination_NSAP*

Long Syntax: ISO.019 received Data Packet Local *source_NSAP* -> *destination_NSAP*

Description: An ISO CLNP Data packet was received with destination NSAP indicating one of the router's NSAP's.

ISO.020

Level: P-TRACE

Short Syntax: ISO.020 sent ERR pkt *destination_NSAP*

Long Syntax: ISO.020 sent Error packet *destination_NSAP*

Description: An ISO CLNP Error packet was sent on receipt of a bad packet.

ISO.021

Level: UE-ERROR

Short Syntax: ISO.021 SRAM err-no NSAP for sbnet

Long Syntax: ISO.021 SRAM error-no NSAP for subnet

Description: A subnet was defined with no NSAP defined for the subnet or domain.

ISO.022

Level: UE-ERROR

Short Syntax: ISO.022 SRAM err-unconcted sbnet

Long Syntax: ISO.022 SRAM error- unconnected subnet

Description: A subnet was defined with no NSAP defined for the subnet or domain.

ISO.023

Level: UE-ERROR

Short Syntax: ISO.023 SRAM err-rte not insrted err= *error_code* Rt *Destination*

Long Syntax: ISO.023 SRAM error- route not intserted error code = *error_code* Route to *Destination*

Description: A statically configured route could not be inserted into routing table.

ISO.024

Level: UE-ERROR

Short Syntax: ISO.024 SRAM err-no adj structs

Long Syntax: ISO.024 SRAM error-no adjacency structures available

Description: Not enough ajacency structures have been configured.

ISO.025

Level: UE-ERROR

Short Syntax: ISO.025 SRAM err-bad ES rte no sub dom = *domain* int= *interface*

Long Syntax: ISO.025 SRAM error-bad static encoded ES route- no subnet domain = *domain* int = *interface*

Description: An encoded end system route was defined for a non-existent subnet.

ISO.027

Level: UE-ERROR

Short Syntax: ISO.027 SRAM err-bad glbl conf

Long Syntax: ISO.027 SRAM error-bad global configuration

Description: The OSI forwarder has been enabled, but either no domains have been defined, or the number of routes or adjacency is set to 0.

ISO.028

Level: UE-ERROR

Short Syntax: ISO.028 SRAM err-not enough mem

Long Syntax: ISO.028 SRAM error-not enough memory

Description: The OSI forwarder could not get the memory needed to operate.

ISO.029

Level: UE-ERROR

Short Syntax: ISO.029 OSI configured to be disabled

Long Syntax: ISO.029 OSI forwarder is configured to be disabled

Description: The OSI forwarder has not been enabled, either because no global information has been entered or the forwarder has been explicitly disabled.

ISO.030

Level: UE-ERROR

Short Syntax: ISO.030 OSI not starting - check config

Long Syntax: ISO.030 OSI forwarder not starting - check configuration

Description: The OSI forwarder is not starting because of the way it's configured.

ISO.031

Level: UE-ERROR

Short Syntax: ISO.031 rcvd echo dest unkwn *source_NSAP -> destination_NSAP*

Long Syntax: ISO.031 received echo packet - destination unknown *source_NSAP -> destination_NSAP*

Description: An ISO CLNP echo packet is received but can not be routed since there is no routing table entry for destination.

ISO.032

Level: UE-ERROR

Short Syntax: ISO.032 no iob avail to snd echo pkt

Long Syntax: ISO.032 no i/o buffer available to send echo packet

Description: An attempt to send an ISO CLNP echo packet failed because of a lack of system i/o buffers.

ISO.033

Level: UE-ERROR

Short Syntax: ISO.033 cnnt fwd echo pkt hndlr err (= *error_code*) *source_NSAP -> destination_NSAP*

Long Syntax: ISO.033 cannot send an echo packet, handler error (err= *error_code*) *source_NSAP -> destination_NSAP*

Description: An ISO CLNP echo packet couldn't be sent because of a handler error.

ISO.034

Level: P-TRACE

Short Syntax: ISO.034 sent ECHO rply pkt *destination_NSAP*

Long Syntax: ISO.034 sent ECHO reply packet *destination_NSAP*

Description: An ISO CLNP ECHO reply packet was sent on receipt of a bad packet.

ISO.035

Level: P-TRACE

Short Syntax: ISO.035 sent ECHO pkt rqst *destination_NSAP*

Long Syntax: ISO.035 sent ECHO request packet *destination_NSAP*

Description: An ISO CLNP ECHO request packet was sent on receipt of a bad packet.

ISO.036

Level: P-TRACE

Short Syntax: ISO.036 rcvd ECHO rqst *source_NSAP -> destination_NSAP*

Long Syntax: ISO.036 received Echo Requet *source_NSAP -> destination_NSAP*

Description: An ISO CLNP Echo packet was received.

ISO.037

Level: P-TRACE

Short Syntax: ISO.037 rcvd ECHO rply *source_NSAP -> destination_NSAP*

Long Syntax: ISO.037 received ECHO reply *source_NSAP -> destination_NSAP*

Description: An ISO CLNP ECHO reply was received.

ISO.038

Level: P-TRACE

Short Syntax: ISO.038 DNA pkt forwarded via OSI at level *rtg_lvl*

Long Syntax: ISO.038 DNA packet forwarded via OSI at level *rtg_lvl*

Description: A DNA packet was received and then passed to OSI for forwarding.

ISO.039

Level: P-TRACE

Short Syntax: ISO.039 DNA pkt translated to OSI pkt *source_NSAP -> destination_NSAP*

Long Syntax: ISO.039 DNA pkt translated to OSI pkt: *source_NSAP -> destination_NSAP*

Description: A DNA data packet was successfully translated to an OSI data packet.

ISO.040

Level: P-TRACE

Short Syntax: ISO.040 Translation of DNA pkt to OSI pkt failed

Long Syntax: ISO.040 Translation of DNA pkt to OSI pkt failed

Description: An attempt to translate a DNA data packet to an OSI data packet failed.

ISO.041

Level: P-TRACE

Short Syntax: ISO.041 OSI pkt translated to DNA pkt *src -> dst*

Long Syntax: ISO.041 OSI pkt translated to DNA pkt: *src -> dst*

Description: An OSI data packet was successfully translated to a DNA data packet.

ISO.043

Level: P-TRACE

Short Syntax: ISO.043 OSI pkt forwarded via DNA at level *rtg_lvl*

Long Syntax: ISO.043 OSI packet forwarded via DNA at level *rtg_lvl*

Description: An OSI packet was received and then passed to DNA for forwarding.

ISO.044

Level: UE-ERROR

Short Syntax: ISO.044 Can't send echo message to local router

Long Syntax: ISO.044 Can't send an echo message to the local router.

Description: An attempt was made to send an echo message to the local router. This could occur if a user enters the send command from the console with the local router's NSAP as the destination address.

ISO.045

Level: UE-ERROR

Short Syntax: ISO.045 Error PDU rcvd from *src_nsap* on nt *network ID* dropped - SP, MS or E/R flag set

Long Syntax: ISO.045 Error PDU received from *src_nsap* on network *network ID* dropped because either the segmentation permitted, more segments, or error report flag was set

Description: An error report PDU was received with either the segmentation permitted, more segments, or error report flag set. These flags are always supposed to be zero for an error PDU. The error PDU is dropped.

ISO.046

Level: UE-ERROR

Short Syntax: ISO.046 max SVC adj reached on cir (*routing-circuit*)

Long Syntax: ISO.046 maximum SVC adjacencies reached on circuit *routing-circuit*

Description: The router cannot forward data on a DA circuit because the circuit already reached maximum allowed adjacencies.

ISO.047

Level: UE-ERROR

Short Syntax: ISO.047 no usable DTEs on cir (*routing-circuit*)

Long Syntax: ISO.047 no usable DTEs on DA ciruit (*routing-circuit*)

Description: Call failures, and all remote DTEs to the DA circuit have timestamps that are more recent than the Recall timer.

ISO.048

Level: UE-ERROR

Short Syntax: ISO.048 call tmplt not found for cir (*routing-circuit*)

Long Syntax: ISO.048 call template not found for circuit (*routing-circuit*)

Description: Call failure, the router cannot find a Call template for the circuit.

ISO.049

Level: C-TRACE

Short Syntax: ISO.049 rcvd clr on cir (*routing-circuit*)

Long Syntax: ISO.049 received Clear on ciruit (*routing-circuit*)

Description: The router received a Clear Indication on a circuit.

ISO.050

Level: C-TRACE

Short Syntax: ISO.050 recall timeout on cir (*routing-circuit*)

Long Syntax: ISO.050 recall timeout on DA circuit (*routing-circuit*)

Description: The recall timer on the DA circuit expired.

ISO.051

Level: C-TRACE

Short Syntax: ISO.051 rsvr timeout on cir (*routing-circuit*)

Long Syntax: ISO.051 reserve timeout on DA circuit (*routing-circuit*)

Description: The reserve timer on a DA SVC expired.

ISO.052

Level: C-TRACE

Short Syntax: ISO.052 idle timeout on cir (*routing-circuit*)

Long Syntax: ISO.052 idle timeout on DA circuit (*routing-circuit*)

Description: The idle timer on a DA SVC expired.

ISO.053

Level: C-TRACE

Short Syntax: ISO.053 calling on cir (*routing-circuit*)

Long Syntax: ISO.053 calling on circuit (*routing-circuit*)

Description: The router placed a call for the circuit.

ISO.054

Level: UE-ERROR

Short Syntax: ISO.054 max calls on cir (*routing-circuit*)

Long Syntax: ISO.054 maximum call attempts made on circuit (*routing-circuit*)

Description: The router made call failures and maximum call attempts on the circuit.

ISO.055

Level: UE-ERROR

Short Syntax: ISO.055 cnnt reg with WAN ser on intf *interface*

Long Syntax: ISO.055 cannot register with WAN services on interface *interface*

Description: The protocol cannot register with WAN services on the interface.

ISO.056

Level: UE-ERROR

Short Syntax: ISO.056 op on non-exist cir (*routing-circuit*)

Long Syntax: ISO.056 attempt to operate on a non-existent circuit (*routing-circuit*)

Description: The router attempted to operate (Enable/Disable) on an unconfigured circuit.

ISO.057

Level: UE-ERROR

Short Syntax: ISO.057 cnnt get X.121 from NASP

Long Syntax: ISO.057 cannot extract the X.121 address from the NSAP given

Description: The destination NSAP is not in X.121-extractable format.

ISO.058

Level: UE-ERROR

Short Syntax: ISO.058 que ovflw on cir (*routing-circuit*)

Long Syntax: ISO.058 buffer queue overflow on DA circuit (*routing-circuit*)

Description: An ISO CLNP output packet queue overflowed. The forwarder dropped the packet.

Chapter 63. LAN Emulation Client Functions (LEC)

This chapter describes LAN Emulation Client Functions (LEC) messages. For information on message content and how to use the message, refer to the Introduction.

LEC.001

Level: C-INFO

Short Syntax: LEC.001 LEC function entry/exit tracing

Long Syntax: LEC.001 LEC function entry/exit tracing

Description: The user can enable/disable the function entry and exit tracing of the LEC by simply turning on/off the display of this message.

LEC.002

Level: C-INFO

Short Syntax: LEC.002 nt *network entry_exit log_point*

Long Syntax: LEC.002 network *network*: lec trace log: *entry_exit log_point*

Description: LEC generic function entry/exit

LEC.003

Level: C-INFO

Short Syntax: LEC.003 nt *network entry_exit log_point*, D1= *arg1*

Long Syntax: LEC.003 network *network*: lec trace log: *entry_exit log_point*, D1= *arg1*

Description: LEC generic function entry/exit with one arg

LEC.004

Level: C-INFO

Short Syntax: LEC.004 nt *network entry_exit log_point*, D1= *arg1*, D2= *arg2*

Long Syntax: LEC.004 network *network*: lec trace log: *entry_exit log_point*, D1= *arg1*, D2= *arg2*

Description: LEC generic function entry/exit with two args

LEC.006

Level: C-INFO

Short Syntax: LEC.006 nt *network trace_type log_point*, conn_handle= *conn_handle*

Long Syntax: LEC.006 network *network*: lec trace log: *trace_type log_point*, conn_handle= *conn_handle*

Description: LEC generic trace msg with one arg - a conn handle

---

LEC.007

Level: C-INFO

Short Syntax: LEC.007 nt *network trace_type log_point*, client_state= *client_state*

Long Syntax: LEC.007 network *network*: lec trace log: *trace_type log_point*, client_state= *client_state*

Description: LEC generic trace msg with one arg - the client state

---

LEC.008

Level: UE-ERROR

Short Syntax: LEC.008 LEC inbnd fr dscrd, bad FC, on nt *network ID*, *word1 word2 word3 word4*

Long Syntax: LEC.008 LEC inbnd fr dscrd, bad FC, on network *network ID*, *word1 word2 word3 word4*

Description: LEC inbound data frame was discarded - bad FC byte

---

LEC.009

Level: U-INFO

Short Syntax: LEC.009 nt *network* LEC state chng from *old_state* to *new_state*

Long Syntax: LEC.009 network *network* LEC client state machine changed from *old_state* to *new_state*

Description: The LEC client state machine (CLSM) keeps track of which state the LEC is currently in. The possible states are: IDLE, LECS_SETUP, CONFIGURE, LES SETUP, JOINING, ARPING FOR BUS, BUS SETUP, and OPERATIONAL.

LEC.010

Level: U-INFO

Short Syntax: LEC.010 nt *network* dest state chng from *old_state* to *new_state*

Long Syntax: LEC.010 network *network* LEC

destination state machine changed from *old_state* to *new_state*

Description: The LEC destination machine (DSM) keeps track of what state the ARP entry is in. The possible states are: UNKNOWN, ARPING, CYCLING, KNOWN, FLUSHING, and CONNECTED.

LEC.011

Level: P_TRACE

Short Syntax: LEC.011 Trace LEC data packet

Long Syntax: LEC.011 Trace LEC data packet

Description: Trace LEC data packet

LEC.012

Level: P_TRACE

Short Syntax: LEC.012 Trace LEC control packet

Long Syntax: LEC.012 Trace LEC control packet

Description: Trace LEC control packet

LEC.013

Level: C-TRACE

Short Syntax: LEC.013 nt *network* Rcvd *ctrl_frame* on conn handle *conn_handle* with xid *xid*

Long Syntax: LEC.013 network *network* Received *ctrl_frame* control frame on conn handle *conn_handle* with tran id of *xid*

Description: The LEC received a control frame from the ATM network

---

LEC.014

Level: C-TRACE

Short Syntax: LEC.014 nt *network* Sent *ctrl_frame* on conn handle *conn_handle* with xid *xid*

Long Syntax: LEC.014 network *network* Sent *ctrl_frame* control frame on conn handle *conn_handle* with tran id of *xid*

Description: The LEC sent a control frame over the ATM network

---

LEC.015

Level: U-INFO

Short Syntax: LEC.015 nt *network trace_type log_point*

Long Syntax: LEC.015 network *network*: lec trace log: *trace_type log_point*

Description: lec general information

LEC.017

Level: U-INFO

Short Syntax: LEC.017 nt *network trace_type log_point*, D1= *arg1*, D2= *arg2*

Long Syntax: LEC.017 network *network*: lec trace log: *trace_type log_point*, D1= *arg1*, D2= *arg2*

Description: lec general information with two args

LEC.020

Level: UE-ERROR

Short Syntax: LEC.020 nt *network error_lvl log_point*

Long Syntax: LEC.020 network *network*: lec error log: *error_lvl log_point*

Description: lec generic error

LEC.021

Level: UE-ERROR

Short Syntax: LEC.021 nt *network error_lvl log_point*, D1= *arg1*

Long Syntax: LEC.021 network *network*: lec error log: *error_lvl log_point*, D1= *arg1*

Description: lec generic error with one arg

LEC.022

Level: UE-ERROR

Short Syntax: LEC.022 nt *network error_lvl log_point*, D1= *arg1*, D2= *arg2*

Long Syntax: LEC.022 network *network*: lec error log: *error_lvl log_point*, D1= *arg1*, D2= *arg2*

Description: lec generic error with two args

LEC.024

Level: UI-ERROR

Short Syntax: LEC.024 open frame SAP failed on nt *n_net*, rc= *retcd*

Long Syntax: LEC.024 open frame SAP failed on network *n_net*, rc = *retcd*

Description: open frame SAP failed

LEC.025

Level: UI-ERROR

Short Syntax: LEC.025 open call SAP failed on nt *n_net*, rc= *retcd*

Long Syntax: LEC.025 open call SAP failed on network *n_net*, rc = *retcd*

Description: open call SAP failed

LEC.026

Level: UI-ERROR

Short Syntax: LEC.026 open data path failed for outgoing call, on nt *n_net*, rc= *retcd*

Long Syntax: LEC.026 open data path failed for outgoing call, on network *n_net*, rc = *retcd*

Description: open data path failed for outgoing call

LEC.027

Level: UI-ERROR

Short Syntax: LEC.027 open data path failed for incoming call, on nt *n_net*, rc= *retcd*

Long Syntax: LEC.027 open data path failed for incoming call, on network *n_net*, rc = *retcd*

Description: open data path failed for incoming call

LEC.028

Level: C-INFO

Short Syntax: LEC.028 Function *function_name* called, nt *network ID*

Long Syntax: LEC.028 Function *function_name* called, on network *network ID*

Description: ATM LEC function called

LEC.029

Level: UI-ERROR

Short Syntax: LEC.029 Start failed, on nt *network ID*, rc= *retcd*

Long Syntax: LEC.029 Start failed, on network *network ID*, rc = *retcd*

Description: Start failed for LEC object

LEC.030

Level: UI-ERROR

Short Syntax: LEC.030 create LEC object failed, on nt *network ID*, rc= *retcd*

Long Syntax: LEC.030 create LEC object failed, on network *network ID*, rc = *retcd*

Description: Could not create LEC object

LEC.031

Level: UI-ERROR

Short Syntax: LEC.031 usr reg failed, on nt *network ID*, rc= *retcd*

Long Syntax: LEC.031 user registration failed, on network *network ID*, rc = *retcd*

Description: LEC could not register

LEC.032

Level: UI-ERROR

Short Syntax: LEC.032 nt *network ID*, ATM nt *network ID* nt nbld

Long Syntax: LEC.032 on network *network ID*, ATM network *network ID* not enabled

Description: ATM interface not enabled

LEC.033

Level: UI-ERROR

Short Syntax: LEC.033 LEC activate failed, on nt *network ID*, rc= *retcd*

Long Syntax: LEC.033 LEC activate failed, on network *network ID*, rc = *retcd*

Description: LEC activate failed

LEC.034

Level: UI-ERROR

Short Syntax: LEC.034 LEC activate complete, on nt *network ID*, rc= *retcd*

Long Syntax: LEC.034 LEC activate complete, on network *network ID*, rc = *retcd*

Description: LEC activate failed.

LEC.035

Level: UI-ERROR

Short Syntax: LEC.035 Outbound frame freed, on nt *network ID*

Long Syntax: LEC.035 Outbound frame freed, on network *network ID*

Description: Outbound frame freed

LEC.036

Level: UI-ERROR

Short Syntax: LEC.036 Outbound frame queued, on nt *network ID*

Long Syntax: LEC.036 Outbound frame queued, on network *network ID*

Description: Outbound frame queued

LEC.037

Level: UI-ERROR

Short Syntax: LEC.037 Transmit failed, on nt *network ID*, rc= *retcd*

Long Syntax: LEC.037 Transmit failed, on network *network ID*, rc = *retcd*

Description: Transmit failed

LEC.038

Level: UI-ERROR

Short Syntax: LEC.038 Outbound frame discarded, on nt *network ID*, rsn= *reason*,state= *state*,hndl= *conn_handle*

Long Syntax: LEC.038 Outbound frame discarded, on network *network ID*, reason = *reason*, DSM state = *state*, conn handle = *conn_handle*

Description: Outbound frame discarded

LEC.039

Level: UI-ERROR

Short Syntax: LEC.039 LEC inbnd fr dscrd, size *size*, on nt *network ID*

Long Syntax: LEC.039 LEC inbound frame discarded, size *size*, on network *network ID*

Description: LEC inbound data frame was discarded - frame too small

---

LEC.040

Level: UI-ERROR

Short Syntax: LEC.040 LEC inbnd fr dscrd, mcast addr, on nt *network ID*

Long Syntax: LEC.040 LEC inbnd fr dscrd, mcast address, on network *network ID*

Description: LEC inbound data frame was discarded - multicast data rcvd on data direct

---

LEC.041

Level: UI-ERROR

Short Syntax: LEC.041 LEC inbnd fr dscrd, bad mac, on nt *network ID*

Long Syntax: LEC.041 LEC inbnd fr dscrd, bad mac address, on network *network ID*

Description: LEC inbound data frame was discarded - wrong MAC address

---

LEC.042

Level: UI-ERROR

Short Syntax: LEC.042 SRAM nt fnd on dsbl, on nt *network ID*

Long Syntax: LEC.042 SRAM not found after disable, on network *network ID*

Description: Couldn't find the matching SRAM block after user disabled the LEC interface.

LEC.043

Level: UI-ERROR

Short Syntax: LEC.043 cancel alarm, on nt *net_no* rc = *rcode*, num *num*

Long Syntax: LEC.043 Bad return from cancel alarm, on network *net_no*, rc = *rcode*, num = *num*

Description: Stopped timer and got bad return code.

LEC.044

Level: C-TRACE

Short Syntax: LEC.044 nt *network* Rcvd Topology on conn handle *conn_handle* with xid *xid*

Long Syntax: LEC.044 network *network* Received Topology control frame on conn handle *conn_handle* with tran id of *xid*

Description: The LEC received a Topology control frame from the ATM network

---

LEC.045

Level: C-TRACE

Short Syntax: LEC.045 nt *network* Sent Topology on conn handle *conn_handle* with xid *xid*

Long Syntax: LEC.045 network *network* Sent Topology control frame on conn handle *conn_handle* with tran id of *xid*

Description: The LEC sent a Topology control frame over the ATM network

---

LEC.046

Level: UI-ERROR

Short Syntax: LEC.046 nt *net_no* LEC QoS object addresses unavailable

Long Syntax: LEC.046 nt *net_no* LEC QoS object addresses unavailable

Description: The LEC QoS object cannot obtain the object addresses of the LEC.

LEC.047

Level: UI-ERROR

Short Syntax: LEC.047 nt *net_no* LEC QoS invld parms, *entity*:(*maxReservedBW*, *trafficType*, *pcr*, *scr*, *qosClass*, *maxBurstSize*),rc= *rcode*

Long Syntax: LEC.047 nt *net_no* LEC QoS invalid parms, *entity*(max= *maxReservedBW* kbps,type= *trafficType*,pcr= *pcr* kbps,scr= *scr* kbps,class= *qosClass*,busrt= *maxBurstSize*),rc= *rcode*

Description: LEC QoS invalid QoS parameters for an entity.

LEC.048

Level: UI-ERROR

Short Syntax: LEC.048 nt *net_no* LEC QoS invld TLV rcvd, *entity*:type= *tlvType*

Long Syntax: LEC.048 nt *net_no* LEC QoS invalid TLV received, *entity*,type= *tlvType*

Description: LEC QoS invalid TLV received in a control frame (entity) with tlvType

---

LEC.049

Level: UI-ERROR

Short Syntax: LEC.049 nt *net_no* LEC QoS error updating statisticts, invld type = *statisticType*

Long Syntax: LEC.049 nt *net_no* LEC QoS error updating statisticts, invld type = *statisticType*

Description: LEC QoS invalid type specified while updating statistics

---

LEC.050

Level: UI-ERROR

Short Syntax: LEC.050 nt *net_no* LEC QoS error obtaining config parm *configParm = value1*

Long Syntax: LEC.050 nt *net_no* LEC QoS error obtaining configuration parameter *configParm = value1*

Description: LEC QoS error while obtaining configuration paramter from SRAM

LEC.051

Level: U-INFO

Short Syntax: LEC.051 nt *net_no* lec *tableId*: incr tbl sz frm *prevMaxConnEnties* to *newMaxConnEntries* : *statusString*

Long Syntax: LEC.051 nt *net_no* lec *tableId*: increase table size from *prevMaxConnEnties* to *newMaxConnEntries* : Status *statusString*

Description: LEC component increasing the size of a table; operationh status is either SUCCESSFULL or FAILED

LEC.052

Level: U-INFO

Short Syntax: LEC.052 nt *net_no* lec *tableId*: decr tbl sz frm *prevMaxConnEnties* to *newMaxConnEntries* : *statusString*

Long Syntax: LEC.052 nt *net_no* lec *tableId*: decrease table size from *prevMaxConnEnties* to *newMaxConnEntries* : Status *statusString*

Description: LEC component decreasing the size of a table; operationh status is either SUCCESSFULL or FAILED

LEC.053

Level: UI-ERROR

Short Syntax: LEC.053 Outbnd frm dscrd, on nt *net_no*,frm sz (*frame_size*) xcds cnfgd frm sz (*config_frame_size*)

Long Syntax: LEC.053 Outbound frame discarded, on network *net_no*, frame size (*frame_size*) exceeds configured frame size (*config_frame_size*)

Description: An outbound frame was discarded, because the frame's size was larger than the configured frame size.

LEC.054

Level: UI-ERROR

Short Syntax: LEC.054 Inbnd frm dscrd, on nt *net_no*,frm sz (*frame_size*) xcds cnfgd frm sz (*config_frame_size*)

Long Syntax: LEC.054 Inbound frame discarded, on network *net_no*, frame size (*frame_size*) exceeds configured frame size (*config_frame_size*)

Description: An inbound frame was discarded, because the frame's size was larger than the configured frame size.

LEC.055

Level: C-INFO

Short Syntax: LEC.055 FLUSH msg prcssd by Redun IP Gtwy on nt *net_no*

Long Syntax: LEC.055 The LEC received a FLUSH msg that was processed by a Redundant IP Gateway on net *net_no*

Description: The LEC received a FLUSH message that was processed by the Redundant IP Gateway. These messages inform the backup gateways that the primary Gateway is attempting to activate.

LEC.057

Level: DEBUG

Short Syntax: LEC.057 nt *net_no*:ntrng fn: *function_name*: *parameters*

Long Syntax: LEC.057 nt *net_no*:entering function: *function_name*: *parameters*

Description: The named function was entered

LEC.058

Level: DEBUG

Short Syntax: LEC.058 nt *net_no*:xtng fn: *function_name*: *parameters*

Long Syntax: LEC.058 nt *net_no*:exiting function: *function_name*: *parameters*

Description: The named function was exited

LEC.059

Level: UE-ERROR

Short Syntax: LEC.059 nt *net_no*:fn: *function_name*:ntry *entry* unknwn dest st *dest_state*

Long Syntax: LEC.059 nt *net_no*:function *function_name*:entry *entry* unknown destination state *dest_state*

Description: In the named function, the entry being processed is in an invalid state

LEC.060

Level: CE-ERROR

Short Syntax: LEC.060 nt *net_no*:Mx LE_ARP rtry cnt (*retry_cnt*) excd fr *arp_entry*

Long Syntax: LEC.060 nt *net_no*:Max LE_ARP retry count (*retry_cnt*) exceeded from *arp_entry*

Description: Max ARP entry count exceeded. Any queued frames will be discarded and the ARP entry will be deleted

LEC.061

Level: C-INFO

Short Syntax: LEC.061 nt *net_no*:ARP cycl tmr xprd:ntry *arp_entry* st *state*

Long Syntax: LEC.061 nt *net_no*:ARP cycle timer expired:entry *arp_entry* state *state*

Description: The ARP cycle timer has expired for the specified entry. The ARP entry will be released.

LEC.062

Level: C-INFO

Short Syntax: LEC.062 nt *net_no*:Flsh tmr xprd:ntry *arp_entry* st *state*

Long Syntax: LEC.062 nt *net_no*:Flush timer expired:entry *arp_entry* state *state*

Description: The Flush timer has expired for the specified entry. If the specified state is FLUSHING, any queued frames will be discarded, and another Flush request will be sent.

LEC.063

Level: UI-ERROR

Short Syntax: LEC.063 nt *net_no*:Orphnd Flsh tmr xprd:ntry *arp_entry* st *state*

Long Syntax: LEC.063 nt *net_no*:Orphaned Flush timer expired:entry *arp_entry* state *state*

Description: An orphaned Flush timer has expired for the specified entry. In the specified state, a flush timer should not be active.

LEC.064

Level: C-INFO

Short Syntax: LEC.064 nt *net_no*:PSD tmr xprd:ntry *arp_entry* st *state*

Long Syntax: LEC.064 nt *net_no*:PSD timer expired:entry *arp_entry* state *state*

Description: The Path Switch Delay timer has expired for the specified entry. If the specified state is FLUSHING, any queued frames will be forwarded.

LEC.065

Level: UI-ERROR

Short Syntax: LEC.065 nt *net_no*:Orphnd PSD tmr xprd:ntry *arp_entry* st *state*

Long Syntax: LEC.065 nt *net_no*:Orphaned PSD timer expired:entry *arp_entry* state *state*

Description: An orphaned PSD timer has expired for the specified entry. In the specified state, a PSD timer should not be active.

LEC.066

Level: C-INFO

Short Syntax: LEC.066 nt *net_no*:Rdy rtry cnt eqls mx rdy rtries (*max_rdy_retries*), ntry *art_entry*

Long Syntax: LEC.066 nt *net_no*:Ready retry count equals max ready retries(*max_rdy_retries*), entry *art_entry*

Description: The ready retry count equals the defined max ready retries, and the entry's state is CALL PENDING. The call will be hung up.

LEC.067

Level: UI-ERROR

Short Syntax: LEC.067 nt *net_no*:xmt ctrl frm, rdy qry fld,ntry *art_entry*

Long Syntax: LEC.067 nt *net_no*:xmit control frame, ready query failed, entry *art_entry*

Description: An error occured while transmitting a ready query.

LEC.068

Level: UI-ERROR

Short Syntax: LEC.068 nt *net_no*:Orphnd Rdy tmr xprd:ntry *art_entry* st *state*

Long Syntax: LEC.068 nt *net_no*:Orphaned Ready timer expired:entry *art_entry* state *state*

Description: An orphaned Ready timer has expired for the specified entry. In the specified state, a Ready timer should not be active.

LEC.069

Level: UI-ERROR

Short Syntax: LEC.069 nt *net_no*:Plc cll fld:out of rsrc, addr *atm_addr*

Long Syntax: LEC.069 nt *net_no*:Place call failed:out of resource, address *atm_addr*

Description: Place call for outbound data direct connection failed, due to a lack of resources.

LEC.070

Level: UI-ERROR

Short Syntax: LEC.070 nt *net_no*:fn *function_name*:ntry *entry* unknwn cll st *call_state*

Long Syntax: LEC.070 nt *net_no*:function *function_name*:entry *entry* unknown call state *call_state*

Description: In the named function, the entry being processed is in an invalid state

---

LEC.071

Level: UE-ERROR

Short Syntax: LEC.071 nt *net_no*:Mltpl cnnctns exst to cllr addr *caller_addr*

Long Syntax: LEC.071 nt *net_no*:Multiple connections exist to caller address *caller_addr*

Description: An inbound data direct call was received. The call will be rejected because multiple connections exist to the calling party.

LEC.072

Level: UI-ERROR

Short Syntax: LEC.072 nt *net_no*:Rcv cll fld:out of rsrc, conn hndl *conn_handle*

Long Syntax: LEC.072 nt *net_no*:Receive call failed:out of resource,conn handle *conn_handle*

Description: Receive call for inbound data direct connection failed, due to a lack of resources.

LEC.073

Level: C-INFO

Short Syntax: LEC.073 nt *net_no*:Rdy Indct rcvd,ntry *art_entry* st *state*

Long Syntax: LEC.073 nt *net_no*:Ready Indicate received,entry *art_entry* state *state*

Description: Ready Indicate frame has been received

LEC.074

Level: UE-ERROR

Short Syntax: LEC.074 nt *net_no*:Rdy Indct rcvd,cll st err ,ntry *art_entry* st *state*

Long Syntax: LEC.074 nt *net_no*:Ready Indicate received,call state error, entry *art_entry* state *state*

Description: A Ready Indicate frame should not be received on a connection in this state.

LEC.075

Level: UI-ERROR

Short Syntax: LEC.075 nt *net_no*:Rdy Indct rcvd,unknwn conn, conn hndl *conn_handle*

Long Syntax: LEC.075 nt *net_no*:Ready Indicate received,unknown connection, conn handle *conn_handle*

Description: A Ready Indicate frame was received on a connection that is unknown by the LEC.

LEC.076

Level: C-INFO

Short Syntax: LEC.076 nt *net_no*:Plc Cll Ack rcvd,ntry *art_entry* st *state*

Long Syntax: LEC.076 nt *net_no*:Place Call Ack Received,entry *art_entry* state *state*

Description: A Place Call Ack has been received

LEC.077

Level: UI-ERROR

Short Syntax: LEC.077 nt *net_no*:xmt ctrl frm,rdy indct fld,addr *atm_addr*

Long Syntax: LEC.077 nt *net_no*:xmit control frame,ready indicate failed, address *atm_addr*

Description: Unable to send the ready indicate control frame

LEC.078

Level: UE-ERROR

Short Syntax: LEC.078 nt *net_no*:Plc Cll Ack rcvd,cll st err ,ntry *art_entry* st *state*

Long Syntax: LEC.078 nt *net_no*:Place Call Ack received,call state error, entry *art_entry* state *state*

Description: A Place Call Ack should not be received on a connection in this state.

LEC.079

Level: UI-ERROR

Short Syntax: LEC.079 nt *net_no*:Plc Cll Ack rcvd,unknwn conn, conn hndl *conn_handle*

Long Syntax: LEC.079 nt *net_no*:Place Call Ack received,unknown connection, conn handle *conn_handle*

Description: A Place Call Ack was received on a connection that is unknown by the LEC.

LEC.080

Level: C-INFO

Short Syntax: LEC.080 nt *net_no*:Rtrng cll estblshmnt, ntry *art_entry*

Long Syntax: LEC.080 nt *net_no*:Retrying call establishment, entry *art_entry*

Description: Call is being disconnected, attempt to re-establish connection

---

LEC.081

Level: UI-ERROR

Short Syntax: LEC.081 nt *net_no*:Dscnnct rcvd,unknwn conn, conn hndl *conn_handle*

Long Syntax: LEC.081 nt *net_no*:Disconnect received,unknown connection, conn handle *conn_handle*

Description: A disconnect was received for a connection that is unknown by the LEC.

LEC.082

Level: C-INFO

Short Syntax: LEC.082 nt *net_no*:HngUp cll rcvd,ntry *art_entry* st *state*

Long Syntax: LEC.082 nt *net_no*:HangUp call received,entry *art_entry* state *state*

Description: HangUp call has been received

LEC.083

Level: UI-ERROR

Short Syntax: LEC.083 nt *net_no*:HngUp cll rcvd,unknwn conn, conn hndl *conn_handle*

Long Syntax: LEC.083 nt *net_no*:HangUp call received,unknown connection, conn handle *conn_handle*

Description: A HangUp Call was received for a connection that is unknown by the LEC.

LEC.084

Level: UI-ERROR

Short Syntax: LEC.084 nt *net_no*:fn *function_name*:unknwn clnt st *client_state*

Long Syntax: LEC.084 nt *net_no*:function *function_name*:unknown client state *client_state*

Description: In the named function, the LEC is in an invalid state

---

LEC.085

Level: UE-ERROR

Short Syntax: LEC.085 nt *net_no*:Plc Cll Ack for Cfg Drct rcvd,clnt st err ,st *state*

Long Syntax: LEC.085 nt *net_no*:Place Call Ack for Cfg Direct received,client state error, state *state*

Description: A Place Call Ack for a Config Direct VCC should not be received for a LEC in this state.

LEC.086

Level: UI-ERROR

Short Syntax: LEC.086 nt *net_no*:Unbl to strt jn rqst tmr

Long Syntax: LEC.086 nt *net_no*:Unable to start join request timer

Description: The join request timer could not be started for this LEC.

LEC.087

Level: UI-ERROR

Short Syntax: LEC.087 nt *net_no*:xmt ctrl frm,jn rqst fld

Long Syntax: LEC.087 nt *net_no*:xmit control frame,join request failed

Description: Unable to send the join request control frame

LEC.088

Level: UE-ERROR

Short Syntax: LEC.088 nt *net_no*:Plc Cll Ack for Ctrl Drct rcvd,clnt st err ,st *state*

Long Syntax: LEC.088 nt *net_no*:Place Call Ack for Control Direct received,client state error, state *state*

Description: A Place Call Ack for a Control Direct VCC should not be received for a LEC in this state.

LEC.089

Level: UE-ERROR

Short Syntax: LEC.089 nt *net_no*:Plc Cll Ack for Mcst Snd rcvd,clnt st err ,st *state*

Long Syntax: LEC.089 nt *net_no*:Place Call Ack for Mcast Send received,client state error, state *state*

Description: A Place Call Ack for a Multicast Send VCC should not be received for a LEC in this state.

LEC.090

Level: UI-ERROR

Short Syntax: LEC.090 nt *net_no*:PVC stup to LECS fld

Long Syntax: LEC.090 nt *net_no*:PVC setup to LECS failed

Description: Attempt to set up PVC for LECS (VPI 0,VCI 17) failed

LEC.091

Level: UE-ERROR

Short Syntax: LEC.091 nt *net_no*:Dscnnct for Cnfg Drct rcvd,clnt st err ,st *state*

Long Syntax: LEC.091 nt *net_no*:Disconnect for Config Direct received,client state error, state *state*

Description: A disconnect for a Config Direct VCC should not be received for a LEC in this state.

LEC.092

Level: UE-ERROR

Short Syntax: LEC.092 nt *net_no*:Dscnnct for Ctrl Drct rcvd,clnt st err ,st *state*

Long Syntax: LEC.092 nt *net_no*:Disconnect for Control Direct received,client state error, state *state*

Description: A disconnect for a Control Direct VCC should not be received for a LEC in this state.

LEC.093

Level: UE-ERROR

Short Syntax: LEC.093 nt *net_no*:Dscnnct for Ctrl Dstrbt rcvd,clnt st err ,st *state*

Long Syntax: LEC.093 nt *net_no*:Disconnect for Control Distribute received,client state error, state *state*

Description: A disconnect for a Control Distribute VCC should not be received for a LEC in this state.

LEC.094

Level: C-INFO

Short Syntax: LEC.094 nt *net_no*:Rtryng Mcst Snd conn to BUS

Long Syntax: LEC.094 nt *net_no*:Retrying Mcst Send connection to BUS

Description: Try to set up Multicast Send connection to BUS.

LEC.095

Level: UE-ERROR

Short Syntax: LEC.095 nt *net_no*:Dscnnct for Mcst Snd rcvd,clnt st err ,st *state*

Long Syntax: LEC.095 nt *net_no*:Disconnect for Mcast Send received,client state error, state *state*

Description: A disconnect for a Multicst Send VCC should not be received for a LEC in this state.

LEC.096

Level: UE-ERROR

Short Syntax: LEC.096 nt *net_no*:Dscnnct for Mcst Fwd rcvd,clnt st err ,st *state*

Long Syntax: LEC.096 nt *net_no*:Disconnect for Mcast Fwd received,client state error, state *state*

Description: A disconnect for a Multicst Forward VCC should not be received for a LEC in this state.

LEC.097

Level: UE-ERROR

Short Syntax: LEC.097 nt *net_no*:Cnfg Rsp err, trans id (x *trans_id*) not eql rsp trans id (x *rsp_trans_id*)

Long Syntax: LEC.097 nt *net_no*:Config Rsp error,trans id (x *trans_id*) not equal response trans id (x *rsp_trans_id*)

Description: The transaction id in the config request and config response, was not equal. The transaction id should be the same in the request and response.

LEC.098

Level: UE-ERROR

Short Syntax: LEC.098 nt *net_no*:Cnfg Rsp err, st *status*

Long Syntax: LEC.098 nt *net_no*:Config Rsp error, status *status*

Description: The config response returned the stated error status.

LEC.099

Level: UE-ERROR

Short Syntax: LEC.099 nt *net_no*:Vldtn of cnfg parms frm LECS fld

Long Syntax: LEC.099 nt *net_no*:Validation of config parms from LECS failed

Description: The validation of the LEC's config parms from the LECS failed.

LEC.101

Level: UE-ERROR

Short Syntax: LEC.101 nt *net_no*:Jn Rsp err, st *status*

Long Syntax: LEC.101 nt *net_no*:Jn Rsp error, status *status*

Description: The join response returned the stated error status.

LEC.102

Level: UI-ERROR

Short Syntax: LEC.102 nt *net_no*:xmt ctrl frm,arp rqst fld, dst *dest_mac*

Long Syntax: LEC.102 nt *net_no*:xmit control frame,arp request failed, dest *dest_mac*

Description: Unable to send the arp request control frame

LEC.103

Level: UI-ERROR

Short Syntax: LEC.103 nt *net_no*:Unbl to strt arp rqst tmr

Long Syntax: LEC.103 nt *net_no*:Unable to start arp request timer

Description: The arp request timer could not be started for this LEC.

LEC.104

Level: UE-ERROR

Short Syntax: LEC.104 nt *net_no*:Jn rsp rcvd,clnt st err ,st *state*

Long Syntax: LEC.104 nt *net_no*:Join response received,client state error, state *state*

Description: A Join response should not be received by a LEC in this state.

LEC.105

Level: UI-ERROR

Short Syntax: LEC.105 nt *net_no*:Flsh rsp rcvd,no ARP ntry ,st *state*

Long Syntax: LEC.105 nt *net_no*:Flush response received,no ARP entry, state *state*

Description: A Flush response was received, but an associated ARP entry was not found.

LEC.106

Level: UE-ERROR

Short Syntax: LEC.106 nt *net_no*:Flsh rsp rcvd,src addr *src_addr*, LEC addr *lec_addr* msmtch

Long Syntax: LEC.106 nt *net_no*:Flush response received,source addr *src_addr*, LEC addr *lec_addr* mismatch

Description: A Flush response was received, but the frame's source address does not match the LEC's address.

LEC.107

Level: UE-ERROR

Short Syntax: LEC.107 nt *net_no*:LE ARP Rsp err, st *status*, dst *dest_mac*

Long Syntax: LEC.107 nt *net_no*:LE ARP Rsp error, status *status*, dest *dest_mac*

Description: The LE ARP response returned the stated error status.

LEC.108

Level: UE-ERROR

Short Syntax: LEC.108 nt *net_no*:Invld tag(*tag*), LE ARP Rsp, st *state*

Long Syntax: LEC.108 nt *net_no*:Invalid tag (*tag*) LE ARP Rsp, state *state*

Description: The LE ARP response contains an invalid tag.

LEC.109

Level: UI-ERROR

Short Syntax: LEC.109 nt *net_no*:LE ARP Rsp rcvd,no ARP ntry, dest *dest_mac* st *state*

Long Syntax: LEC.109 nt *net_no*:LE ARP Rsp received,no ARP entry, dest *dest_mac* state *state*

Description: A LE ARP response was received, but an associated ARP entry was not found.

LEC.110

Level: UE-ERROR

Short Syntax: LEC.110 nt *net_no*:Invld tag(*tag*), LE NARP Rqst, st *state*

Long Syntax: LEC.110 nt *net_no*:Invalid tag (*tag*) LE ARP Rqst, state *state*

Description: The LE NARP request contains an invalid tag.

LEC.111

Level: UI-ERROR

Short Syntax: LEC.111 nt *net_no*:Jn timeout exceeded

Long Syntax: LEC.111 nt *net_no*:Join timeout exceeded

Description: The join timeout was exceeded.

LEC.112

Level: UI-ERROR

Short Syntax: LEC.112 nt *net_no*:Cnfg rqst tmout (*config_timeout*) excds cntrl tmout(*control_timeout*)

Long Syntax: LEC.112 nt *net_no*:Config request timeout (*config_timeout*) exceeds control timeout (*control_timeout*)

Description: The config request timeout exceeds the control timeout.

LEC.113

Level: UI-ERROR

Short Syntax: LEC.113 nt *net_no*:Cnfg Req tmr xprd,clnt st err ,st *state*

Long Syntax: LEC.113 nt *net_no*:Config Request timer expired,client state error, state *state*

Description: The Config Request timer should not expire for a LEC in this state.

LEC.114

Level: UI-ERROR

Short Syntax: LEC.114 nt *net_no*:LE ARP rtry cnt (*arp_retry_cnt*) fr BUS eqls mx rtries

Long Syntax: LEC.114 nt *net_no*:LE ARP retry count (*arp_retry_cnt*) for BUS equals max retries

Description: The LE ARP retry count equals the max retry count.

LEC.115

Level: UI-ERROR

Short Syntax: LEC.115 nt *net_no*:LE ARP fr BUS tmr xprd,clnt st err ,st *state*

Long Syntax: LEC.115 nt *net_no*:LE ARP for BUS timer expired,client state error, state *state*

Description: The LE ARP Request timer should not expire for a LEC in this state.

LEC.116

Level: UE-ERROR

Short Syntax: LEC.116 nt *net_no*:Cnfg Rsp cntrl frm err,src (*src_dest*) not LEC's (*lec_mac*)

Long Syntax: LEC.116 nt *net_no*:Config Response control frame error, source (*src_dest*) not LEC's (*lec_mac*)

Description: A config response control frame was received. The source MAC address does not equal the LEC's MAC address.

LEC.117

Level: UE-ERROR

Short Syntax: LEC.117 nt *net_no*:Cnfg Rsp cntrl frm err, invld eln nm sz (*name_size*)

Long Syntax: LEC.117 nt *net_no*:Config Response control frame error,invalid ELAN name size (*name_size*)

Description: A config response control frame was received. The ELAN name size is invalid.

LEC.118

Level: UE-ERROR

Short Syntax: LEC.118 nt *net_no*:Cnfg Rsp cntrl frm err, invld frm sz (*frame_size*)

Long Syntax: LEC.118 nt *net_no*:Config Response control frame error,invalid frame size (*frame_size*)

Description: A config response control frame was received. The maximum frame size is invalid.

LEC.119

Level: UE-ERROR

Short Syntax: LEC.119 nt *net_no*:Cnfg Rsp cntrl frm err, invld prmtr *config_parm*

Long Syntax: LEC.119 nt *net_no*:Config Response control frame error,invalid parameter *config_parm*

Description: A config response control frame was received. The specified config parm is invalid, or out of range.

LEC.120

Level: UE-ERROR

Short Syntax: LEC.120 nt *net_no*:Jn Rsp cntrl frm err, invld frm sz (*frame_size*)

Long Syntax: LEC.120 nt *net_no*:Join Response control frame error,invalid frame size (*frame_size*)

Description: A join response control frame was received. The maximum frame size is invalid.

LEC.121

Level: UE-ERROR

Short Syntax: LEC.121 nt *net_no*:Jn Rsp cntrl frm err, lan typ (*lan_type*) not eql to LEC (*lec_lan_type*)

Long Syntax: LEC.121 nt *net_no*:Join Response control frame error,lan type (*lan_type*) not equal to LEC (*lec_lan_type*)

Description: A join response control frame was received. The lan type in the response does not match the LEC's lan type .

LEC.122

Level: UI-ERROR

Short Syntax: LEC.122 nt *net_no*:Unbl to strt cnfg rqst tmr

Long Syntax: LEC.122 nt *net_no*:Unable to start cnfg request timer

Description: The config request timer could not be started for this LEC.

LEC.123

Level: UI-ERROR

Short Syntax: LEC.123 nt *net_no*:xmt cntrl frm,cnfg rqst fld

Long Syntax: LEC.123 nt *net_no*:xmit control frame,cnfg request failed

Description: Unable to send the config request control frame

LEC.124

Level: UI-ERROR

Short Syntax: LEC.124 nt *net_no*:Jn Req tmr xprd,clnt st err ,st *state*

Long Syntax: LEC.124 nt *net_no*:Join Request timer expired,client state error, state *state*

Description: The Join Request timer should not expire for a LEC in this state.

LEC.125

Level: UI-ERROR

Short Syntax: LEC.125 nt *net_no*:set alrm tmr rtrn null pntr, fn *function_name*, alrm type *alarm_type*

Long Syntax: LEC.125 nt *net_no*:Set Alarm timer returned a null pointer, function *function_name*, alarm type *alarm_type*

Description: Unable to allocate an alarm timer.

LEC.126

Level: C-INFO

Short Syntax: LEC.126 nt *net_no*:Regstrd lan dest/RD *lan_dest* w/LES

Long Syntax: LEC.126 nt *net_no*:Registered lan destination/Route Descriptor *lan_dest* with LES

Description: LEC registered a lan destination or route descriptor with the LES.

LEC.127

Level: UI-ERROR

Short Syntax: LEC.127 nt *net_no*:Fld to reg lan dest/RD *lan_dest* w/LES

Long Syntax: LEC.127 nt *net_no*:Failed to register lan destination/Route Descriptor *lan_dest* with LES

Description: Attempt to register a lan destination or route descriptor with the LES failed.

LEC.128

Level: UI-ERROR

Short Syntax: LEC.128 nt *net_no*:fn *function_name*:unknwn RSM st *rsm_state*

Long Syntax: LEC.128 nt *net_no*:function *function_name*:unknown RSM state *rsm_state*

Description: In the named function, the LEC RSM is in an invalid state.

LEC.129

Level: UI-ERROR

Short Syntax: LEC.129 nt *net_no*:Fld to reg lan dest/RD *dest_addr* (atm addr *dest_atm_addr*) w/LES

Long Syntax: LEC.129 nt *net_no*:Failed to register lan destination/Route Descriptor *dest_addr* (atm address *dest_atm_addr*) with LES

Description: Attempt to register a lan destination or route descriptor with the LES failed.

LEC.130

Level: UI-ERROR

Short Syntax: LEC.130 nt *net_no*:fn *function_name*:lec fld to get reg req timr

Long Syntax: LEC.130 nt *net_no*:function *function_name*:lec failed to get register request timer

Description: In the named function, the LEC failed to get a register request timer.

LEC.131

Level: UI-ERROR

Short Syntax: LEC.131 nt *net_no*:fn *function_name*:lec reg xmit join req fld

Long Syntax: LEC.131 nt *net_no*:function *function_name*:lec register transmit join request failed

Description: In the named function, the LEC failed to transmit join request.

LEC.132

Level: UI-ERROR

Short Syntax: LEC.132 nt *net_no*:fn *function_name*: unxpctd rsp rcvd, st *rsm_state*, rsp *rsp_OpCode*

Long Syntax: LEC.132 nt *net_no*:function *function_name*:unexpected response received, state *rsm_state*, response *rsp_OpCode*

Description: In the named function, an unexpected response received for given RSM state.

LEC.133

Level: U-INFO

Short Syntax: LEC.133 nt *net_no*:LEC nt oprtnl, st *state*

Long Syntax: LEC.133 nt *net_no*:LEC not operational, state *state*

Description: The LEC is not operational.

LEC.134

Level: UI-ERROR

Short Syntax: LEC.134 nt *net_no*:Outbnd frm dscrd, dst = *dest_addr* src = *src_addr*, rsn = *reason*

Long Syntax: LEC.134 nt *net_no*:Outbound frame discarded, dest = *dest_addr* source = *src_addr*, reason = *reason*

Description: Outbound frame discarded

LEC.135

Level: DEBUG

Short Syntax: LEC.135 nt *net_no*:Snd frm on Mcast Snd VCC, dst = *dest_addr* src = *src_addr*

Long Syntax: LEC.135 nt *net_no*:Send frame on Multicast Send VCC, dest = *dest_addr* source = *src_addr*

Description: A frame was sent on the Multicast Send VCC.

LEC.136

Level: U-INFO

Short Syntax: LEC.136 nt *net_no*:No cnnctn to BUS

Long Syntax: LEC.136 nt *net_no*:No connection to BUS

Description: The LEC does not have a connection to the BUS.

LEC.137

Level: UI-ERROR

Short Syntax: LEC.137 nt *net_no*:No ARP entrs avlbl,hngup lst usd conn fld

Long Syntax: LEC.137 nt *net_no*:No ARP entires available, hangup least used connection failed

Description: The LEC's ARP table is full.

LEC.138

Level: DEBUG

Short Syntax: LEC.138 nt *net_no*:Snd Drctd frm, dst = *dest_addr* src = *src_addr*

Long Syntax: LEC.138 nt *net_no*:Send Directed frame,

dest = *dest_addr* source = *src_addr*

Description: Directed frame was sent.

LEC.139

Level: UI-ERROR

Short Syntax: LEC.139 nt *net_no*:Inbnd frm dscrd, dst = *dest_addr* src = *src_addr*, rsn = *reason*

Long Syntax: LEC.139 nt *net_no*:Inbound frame discarded, dest = *dest_addr* source = *src_addr*, reason = *reason*

Description: Inbound frame discarded

LEC.140

Level: DEBUG

Short Syntax: LEC.140 nt *net_no*:Rcvd *frame_type* cntrl frm,trans id x *trans_id*, conn hndl *conn_handle*

Long Syntax: LEC.140 nt *net_no*:Received *frame_type* control frame, trans id x *trans_id*, connection handle *conn_handle*

Description: The specified control frame was received by the LEC.

LEC.141

Level: U-INFO

Short Syntax: LEC.141 nt *net_no*:Dscrd LE_ARP Rqst. *str_port* prt nt in fwding st. Prt st *str_state*(0x *port_state*)

Long Syntax: LEC.141 nt *net_no*:Discarded LE_ARP Request. *str_port* port not in forwarding state. Port state is *str_state*(0x *port_state*)

Description: The LEC should not respond to a LE_ARP Request if the local or remote port is not in the forwarding state.

LEC.142

Level: UI_ERROR

Short Syntax: LEC.142 nt *net_no*:Get LEC's cnfg prms fld

Long Syntax: LEC.142 nt *net_no*:Get LEC's config parameters failed

Description: Reading the LEC's configuration parameters failed.

LEC.143

Level: UI_ERROR

Short Syntax: LEC.143 nt *net_no*:Create objct *LEC_object* fld

Long Syntax: LEC.143 nt *net_no*:Create object *LEC_object* failed

Description: The specified LEC object could not be created.

LEC.144

Level: UI_ERROR

Short Syntax: LEC.144 nt *net_no*:Rgstr dest *dest_addr* fld, rsn *reason*

Long Syntax: LEC.144 nt *net_no*:Register destination *dest_addr* failed, rsn *reason*

Description: The specified destination could not be registered with the LES.

LEC.145

Level: UI_ERROR

Short Syntax: LEC.145 nt *net_no*:ATM addr actvtd, invld st *client_state*

Long Syntax: LEC.145 nt *net_no*:ATM address activated, invalid state *client_state*

Description: The ATM addr should not be activated while the LEC is in this state.

LEC.146

Level: UI_ERROR

Short Syntax: LEC.146 nt *net_no*:ATM addr actvtd, ILMI fld

Long Syntax: LEC.146 nt *net_no*:ATM address activated, ILMI failure

Description: An ILMI failure occurred.

LEC.147

Level: UE_ERROR

Short Syntax: LEC.147 nt *net_no*:Get LECS addr fld

Long Syntax: LEC.147 nt *net_no*:Get LECS address failed

Description: The LEC was unable to get the LECS address.

LEC.148

Level: UE_ERROR

Short Syntax: LEC.148 nt *net_no*:Cntrl Drct setup fld

Long Syntax: LEC.148 nt *net_no*:Control Direct setup failed

Description: The LEC was unable to set up the Control Direct VCC to the LES.

LEC.149

Level: UE_ERROR

Short Syntax: LEC.149 nt *net_no*:Cnfg Drct setup fld

Long Syntax: LEC.149 nt *net_no*:Config Direct setup failed

Description: The LEC was unable to set up the Config Direct VCC to the LECS.

LEC.150

Level: U_INFO

Short Syntax: LEC.150 nt *net_no*:Unslctd Cnfg Rsp rcvd

Long Syntax: LEC.150 nt *net_no*:Unsolicited Config Response received

Description: The LEC received an unsolicited Config Response frame.

LEC.151

Level: DEBUG

Short Syntax: LEC.151 nt *net_no*:xmt ctrl frm *frame_type* ,trans id x *trans_id*, conn hndl *conn_handle*

Long Syntax: LEC.151 nt *net_no*:xmit control frame *frame_type*, trans id x *trans_id*, connection handle *conn_handle*

Description: The specified control frame was transmitted.

LEC.152

Level: UE-ERROR

Short Syntax: LEC.152 nt *net_no*:Invld LE ARP Rsp,rsn *status*

Long Syntax: LEC.152 nt *net_no*:Invalid LE ARP Response, reason *status*

Description: The LE ARP response is invalid for the stated reason.

LEC.153

Level: UE-ERROR

Short Syntax: LEC.153 nt *net_no*:Place call ack for unknwn conn, st *client_state*

Long Syntax: LEC.153 nt *net_no*:Place call ack for unknown connection, state *client_state*

Description: A place call ack was received for an unknown connection.

LEC.154

Level: UE-ERROR

Short Syntax: LEC.154 nt *net_no*:lec_cmgr has conn_tbl/freelist inconsistency, index = *index*

Long Syntax: LEC.154 nt *net_no*:lec_mgr has conn_tbl/freelist inconsisency, index = *index*

Description: An inconsistency has been found when acquiring an new conn blk.

LEC.155

Level: UE-ERROR

Short Syntax: LEC.155 nt *net_no*:lec_cmgr:decrTblSz err: tblSz= *max_conn_handles*, connEntries= *non_null_entries*, flstAvl= *freelist_available*

Long Syntax: LEC.155 nt *net_no*:lec_cmgr:decrTblSz error: tblSz= *max_conn_handles*, connEntries= *non_null_entries*, flstAvl= *freelist_available*

Description: An inconsistency has been found when decreasing a LEC connection table.

LEC.156

Level: U-INFO

Short Syntax: LEC.156 nt *net_no*:lec_cmgr:decrTblSz check OK: tblSz= *max_conn_handles*, flstAvl= *freelist_available*

Long Syntax: LEC.156 nt *net_no*:lec_cmgr:decrTblSz check OK: tblSz= *max_conn_handles*, flstAvl= *freelist_available*

Description: Consistency check OK in lec_cmgr:decrease_conn_tbl but table not decreased.

LEC.157

Level: UI-ERROR

Short Syntax: LEC.157 nt *net_no*:Unbl to allct ART entry

Long Syntax: LEC.157 nt *net_no*:Unable to allocate ART entry

Description: An ART entry could not be allocated.

LEC.158

Level: DEBUG

Short Syntax: LEC.158 nt *net_no*:Snt frm to BUS on conn hndl *conn_handle*, frm cnt *frame_cnt*

Long Syntax: LEC.158 nt *net_no*:Sent frame to BUS on conn handle *conn_handle*, frame count *frame_cnt*

Description: The LEC sent a frame to the BUS, because a data direct does not yet exist to the destination.

LEC.159

Level: UE-ERROR

Short Syntax: LEC.159 nt *net_no*:ARP Rsp err, trans id (x *trans_id*) not eql rsp trans id (x *rsp_trans_id*)

Long Syntax: LEC.159 nt *net_no*:ARP Rsp error,trans id (x *trans_id*) not equal response trans id (x *rsp_trans_id*)

Description: The transaction id in the ARP request and ARP response, was not equal. The transaction id should be the same in the request and response.

LEC.160

Level: C-INFO

Short Syntax: LEC.160 nt *net_no*:LEC rcvd mltpl ARP rsp

Long Syntax: LEC.160 nt *net_no*:LEC received multiple ARP responses

Description: The LEC received multiple ARP responses to an ARP request.

LEC.161

Level: UE-ERROR

Short Syntax: LEC.161 nt *net_no*:fn: *function_name*:ntry *entry* invld dest st *dest_state*

Long Syntax: LEC.161 nt *net_no*:function *function_name*:entry *entry* invalid destination state *dest_state*

Description: In the named function, the entry being processed is in an invalid state

---

LEC.162

Level: UI-ERROR

Short Syntax: LEC.162 nt *net_no*:Dscrd *frame_cnt* queued frms, st *dest_state*, conn hndl *conn_hndl*

Long Syntax: LEC.162 nt *net_no*:Discarded *frame_cnt* queued frames, state *dest_state*, conn handle *conn_hndl*

Description: The LEC has discarded the stated number of queued frames

---

LEC.163

Level: UI-ERROR

Short Syntax: LEC.163 nt *net_no*:Purge queued frms fld

Long Syntax: LEC.163 nt *net_no*:Purge queued frames failed

Description: An error occurred while the LEC was attempting to free the queued frames

---

LEC.164

Level: UI-ERROR

Short Syntax: LEC.164 nt *net_no*:Err purging queued frms, queue not empty

Long Syntax: LEC.164 nt *net_no*:Error purging queued frames, queue not empty

Description: All frames should have been removed from the queue.

LEC.165

Level: UI-ERROR

Short Syntax: LEC.165 nt *net_no*:Snd queued frms fld,queue empty

Long Syntax: LEC.165 nt *net_no*:Send queued frames failed, queue empty

Description: An error occurred while the LEC was attempting to send the queued frames

---

LEC.166

Level: UI-ERROR

Short Syntax: LEC.166 nt *net_no*:Err snding queued frms, queue not empty

Long Syntax: LEC.166 nt *net_no*:Error sending queued frames, queue not empty

Description: All frames should have been sent.

LEC.167

Level: UI-ERROR

Short Syntax: LEC.167 nt *net_no*:xmt ctrl frm,flsh rqst fld, dst *dest_mac*

Long Syntax: LEC.167 nt *net_no*:xmit control frame,flush request failed, dest *dest_mac*

Description: Unable to send the flush request control frame

LEC.168

Level: DEBUG

Short Syntax: LEC.168 nt *net_no*:srch tbl *table*, addr *address*

Long Syntax: LEC.168 nt *net_no*:search tbl *table*, address *address*

Description: The specified table was searched for the address.

LEC.169

Level: UI-ERROR

Short Syntax: LEC.169 nt *net_no*:Unbl to add *database_type* dtbs entry, *entry*

Long Syntax: LEC.169 nt *net_no*:Unable to add *database_type* database entry, *entry*

Description: An entry could not be added to the specified database.

LEC.170

Level: UI-ERROR

Short Syntax: LEC.170 nt *net_no*:ARP tbl full, No ARP entrs avlbl

Long Syntax: LEC.170 nt *net_no*:ARP table full, No ARP entries available

Description: The LEC's ARP table is full.

LEC.171

Level: UI-ERROR

Short Syntax: LEC.171 nt *net_no*:No entrs in tbl

Long Syntax: LEC.171 nt *net_no*:No entries in table

Description: The LEC's ARP table is empty.

LEC.172

Level: UI-ERROR

Short Syntax: LEC.172 nt *net_no*:Invld tag(*tag*), rls arp entry

Long Syntax: LEC.172 nt *net_no*:Invalid tag (*tag*),release arp entry

Description: Attempting to release an ARP entry which has an invalid tag.

LEC.173

Level: UI-ERROR

Short Syntax: LEC.173 nt *net_no*:ntry *arp_entry*, xmit queue nt empty (*xmit_queue_count*), rls arp entry

Long Syntax: LEC.173 nt *net_no*:Entry *arp_entry*, xmit queue not empty (*xmit_queue_count*),release arp entry

Description: Attempting to release an ARP entry which has a non-empty transmit queue.

LEC.174

Level: UI-ERROR

Short Syntax: LEC.174 nt *net_no*:fn *function*, unbl to allct memry

Long Syntax: LEC.174 nt *net_no*:Function *function*, unable to allocate memory

Description: The specified function was unable to allocate memory.

LEC.175

Level: UI-ERROR

Short Syntax: LEC.175 nt *net_no*:Invld LEC or ART ptr

Long Syntax: LEC.175 nt *net_no*:Invalid LEC or ART pointer

Description: The LEC or ART pointer is invalid.

LEC.176

Level: UE-ERROR

Short Syntax: LEC.176 nt *net_no*:Invld AAL parms(*AAL_parms*), cll rjctd

Long Syntax: LEC.176 nt *net_no*:Invalid AAL parms(*AAL_parms*), call rejected

Description: Call will be rejected due to invalid AAL parameters.

LEC.177

Level: U-INFO

Short Syntax: LEC.177 nt *net_no*:Invld PID in rcv cll

Long Syntax: LEC.177 nt *net_no*:Invalid PID in receive call

Description: Call was received with an invalid PID.

LEC.178

Level: UI-ERROR

Short Syntax: LEC.178 nt *net_no*:place call fld, rsn *reason*

Long Syntax: LEC.178 nt *net_no*:place call failed, reason *reason*

Description: Place call failed for the following reason.

LEC.179

Level: UI-ERROR

Short Syntax: LEC.179 nt *net_no*:get ART cnfg parms fld

Long Syntax: LEC.179 nt *net_no*:get ART config parms failed

Description: Unable to get ART config parameters.

LEC.180

Level: UI-ERROR

Short Syntax: LEC.180 nt *net_no*:ART tbl full, No ART entrs avlbl

Long Syntax: LEC.180 nt *net_no*:ART table full, No ART entries available

Description: The LEC's ART table is full.

LEC.181

Level: UI-ERROR

Short Syntax: LEC.181 nt *net_no*:ART entry alrdy freed

Long Syntax: LEC.181 nt *net_no*:ART entry already freed

Description: The ART has already been freed.

LEC.182

Level: UI-ERROR

Short Syntax: LEC.182 nt *net_no*:ART entry aging suspnd

Long Syntax: LEC.182 nt *net_no*:ART entry aging suspended

Description: The ART entries will not be aged out.

LEC.183

Level: U-INFO

Short Syntax: LEC.183 nt *network ID*:Old llh in func *name* dscrding: arp_ptr=0x *arp_entry_ptr* arp_ts= *arp_time_stamp* vcc_ptr=0x *vcc_handle* vcc_ts= *vcc_time_created*

Long Syntax: LEC.183 nt *network ID*:Old llh in function *name* discarding:arp_ptr=0x *arp_entry_ptr* arp_ts= *arp_time_stamp* vcc_ptr=0x *vcc_handle* vcc_ts= *vcc_time_created*

Description: Old llh used in LEC fastpath. A new llh is built.

LEC.186

Level: UE-ERROR

Short Syntax: LEC.186 nt *net_no*:Jn Rsp cntrl frm err, invld prmtr *config_parm*

Long Syntax: LEC.186 nt *net_no*:Join Response control frame error,invalid parameter *config_parm*

Description: A join response control frame was received. The specified join parm is invalid, or out of range.

LEC.187

Level: C-INFO

Short Syntax: LEC.187 nt *net_no*:MUF tmr xprd:ntry *arp_entry* st *state*

Long Syntax: LEC.187 nt *net_no*:MUF timer expired:entry *arp_entry* state *state*

Description: The Maximum Unknown Frame timer has expired for the specified entry. If the specified state is not CONNECTED or FLUSHING, up to the Maximum Unknown Frame Count of queued frames will be forwarded.

LEC.188

Level: UI-ERROR

Short Syntax: LEC.188 nt *net_no*:Orphnd MUF tmr xprd:ntry *arp_entry* st *state*

Long Syntax: LEC.188 nt *net_no*:Orphaned MUF timer expired:entry *arp_entry* state *state*

Description: An orphaned MUF timer has expired for the specified entry. In the specified state, a MUF timer should not be active.

LEC.189

Level: UI-ERROR

Short Syntax: LEC.189 nt *net_no*:Fwd Dscn tmr xprd,clnt st err ,st *state*

Long Syntax: LEC.189 nt *net_no*:Forward disconnect timer expired,client state error, state *state*

Description: The forward disconnect timer should not expire for a LEC in this state.

LEC.190

Level: UI-ERROR

Short Syntax: LEC.190 nt *net_no*:Unbl to strt fwd dscn tmr

Long Syntax: LEC.190 nt *net_no*:Unable to start forward disconnect timer

Description: The forward disconnect timer could not be started for this LEC.

LEC.191

Level: UE-ERROR

Short Syntax: LEC.191 nt *net_no*:Reg Rsp err, trans id (x *trans_id*) not eql rsp trans id (x *rsp_trans_id*)

Long Syntax: LEC.191 nt *net_no*:Register Rsp error,trans id (x *trans_id*) not equal response trans id (x *rsp_trans_id*)

Description: The transaction id in the register request

and register response, was not equal. The transaction id should be the same in the request and response.

LEC.192

Level: UE-ERROR

Short Syntax: LEC.192 nt *net_no*:Unsuccsfl reg rsp rcvd, LEC trmntd

Long Syntax: LEC.192 nt *net_no*:Unsuccessful register response received, LEC will be terminated

Description: An unsuccessful register response was received. The LEC's ELAN membership will be terminated.

LEC.193

Level: UI-ERROR

Short Syntax: LEC.193 nt *net_no*:Regstrtn tmr exceeded, LEC trmntd

Long Syntax: LEC.193 nt *net_no*:Registration timer exceeded, LEC will be terminated

Description: The registration timer was exceeded. The LEC's ELAN membership will be terminated.

LEC.194

Level: UI-ERROR

Short Syntax: LEC.194 nt *net_no*:Unbl to xmit reg req, LEC trmntd

Long Syntax: LEC.194 nt *net_no*:Unable to xmit register request, LEC will be terminated

Description: The LEC was unable to send a register request. The LEC's ELAN membership will be terminated.

LEC.195

Level: U-INFO

Short Syntax: LEC.195 nt *net_no*:in_use_flag set to TRUE for ARP entry *arp_entry*

Long Syntax: LEC.195 nt *net_no*:in_use_flag set to TRUE for ARP entry *arp_entry*

Description: Based on the domain member response message from the LAN Switch, the in_use_flag for this route descriptor is set to TRUE.

LEC.196

Level: U-INFO

Short Syntax: LEC.196 nt *net_no*:switch domain member *arp_entry*, not found in ARP table

Long Syntax: LEC.196 nt *net_no*:switch domain member *arp_entry*, not found in ARP table

Description: The switch domain member was not found in the ARP table.

LEC.197

Level: U-INFO

Short Syntax: LEC.197 nt *net_no* xmit queue hgh wtr mrk, Strt ARP Sweep Timer *timer_status*

Long Syntax: LEC.197 nt *net_no* xmit queue high water mark, Start ARP Sweep Timer *timer_status*

Description: LEC reached high water mark on the transmit buffer queue while trying to transmit an LE_ARP request during the verification or topology sweep timer. The ARP sweep timer is started to finish sending the rest of the outstanding LE_ARPs.

LEC.198

Level: UI-ERROR

Short Syntax: LEC.198 nt *network ID*:LEC at hgh wtr mrk on xmit buf queue *iorbs opfair*

Long Syntax: LEC.198 nt *network ID*:LEC at high water mark on xmit buffer queue *iorbs opfair*

Description: LEC reached high water mark on the ATM transmit buffer queue while trying to transmit a control frame.

LEC.199

Level: C-INFO

Short Syntax: LEC.199 nt *net_no*:LE ARP Rqst snt, for LES/BUS tmout, cnt *dest_mac*, dst *count* st *state*

Long Syntax: LEC.199 nt *net_no*:LE ARP Request sent, for LES/BUS timeout, count *dest_mac*, dest *count* state *state*

Description: An LE ARP request was sent for LES/BUS activity timeout.

LEC.200

Level: C-INFO

Short Syntax: LEC.200 nt *net_no*:LE ARP Rsp rcvd, for LES/BUS tmout, dest *dest_mac* st *state*

Long Syntax: LEC.200 nt *net_no*:LE ARP Rsp received, for LES/BUS timeout, dest *dest_mac* state *state*

Description: An LE ARP response was received for LES/BUS activity timeout.

LEC.201

Level: C_INFO

Short Syntax: LEC.201 nt *net_no*:updtd cnfgrtn for fld ' *field_name*'

Long Syntax: LEC.201 nt *net_no*:updated configuration for field ′ *field_name*′

Description: During initialization, an outdated configuration record was discovered. Certain parameters in the configuration of the LEC were updated to reflect new functional abilities. This event is common after moving to a new code release.

LEC.202

Level: DEBUG

Short Syntax: LEC.202 nt *net_no*:Inbnd frm dscrd, dst = *dest_addr* src = *src_addr*, rsn = *reason*

Long Syntax: LEC.202 nt *net_no*:Inbound frame discarded, dest = *dest_addr* source = *src_addr*, reason = *reason*

Description: Inbound frame discarded

LEC.203

Level: UI-ERROR

Short Syntax: LEC.203 nt *net_no*:LES/BUS tmout, cnt *dest_mac*, dst *count* st *state*

Long Syntax: LEC.203 nt *net_no*:LES/BUS timeout, count *dest_mac*, dest *count* state *state*

Description: Did not receive a response when sending an LE ARP request. Assuming the LES/BUS are down and resetting the LEC. By resetting, the LEC can find a new LES/BUS.

Chapter 64. LAN Emulation Configuration Server (LECS)

This chapter describes LAN Emulation Configuration Server (LECS) messages. For information on message content and how to use the message, refer to the Introduction.

LECS.001

Level: UE_ERROR

Short Syntax: LECS.001 LECS: crt fld: dplct LECS

Long Syntax: LECS.001 LECS: create failed: duplicate LECS

Description: LECS already exists so another cannot be created.

LECS.002

Level: UI_ERROR

Short Syntax: LECS.002 LECS: crt fld: mem alloc err

Long Syntax: LECS.002 LECS: create failed: memory allocation error

Description: A memory allocation error occurred while attempting to create the LECS.

Action: Contact your customer service representative.

LECS.003

Level: U_INFO

Short Syntax: LECS.003 LECS: starting operation

Long Syntax: LECS.003 LECS: starting operation

Description: The LECS initialization procedures are starting.

LECS.004

Level: UE_ERROR

Short Syntax: LECS.004 LECS: doesn't exist: *descrip_string*

Long Syntax: LECS.004 LECS: does not exist: *descrip_string*

Description: The user is attempting to add, delete, or modify resources of the LECS when the LECS has yet to be created. The offending action is given by the parameter.

LECS.005

Level: UE_ERROR

Short Syntax: LECS.005 LECS: inactv state: *descrip_string*

Long Syntax: LECS.005 LECS: inactive state: *descrip_string*

Description: The user is attempting to add, delete, or modify resources of the LECS when the LECS is in a state which does not permit this action. The offending action is given by the parameter.

LECS.006

Level: UE_ERROR

Short Syntax: LECS.006 LECS: dlt fld: no LECS

Long Syntax: LECS.006 LECS: delete failed: no LECS

Description: An attempt was made to delete the LECS when the LECS does not exist.

LECS.007

Level: U_INFO

Short Syntax: LECS.007 LECS: dltd

Long Syntax: LECS.007 LECS: deleted

Description: The LECS has been deleted.

LECS.008

Level: UE_ERROR

Short Syntax: LECS.008 LECS: stp fld: no LECS

Long Syntax: LECS.008 LECS: stop failed: no LECS

Description: An attempt was made to stop the LECS when the LECS does not exist.

LECS.009

Level: UI_ERROR

Short Syntax: LECS.009 LECS: stp fld: invld ctl blk

Long Syntax: LECS.009 LECS: stop failed: invalid control block

Description: An attempt was made to stop the LECS using an invalid pointer to its control block.

Action: Contact your customer service representative.

LECS.010

Level: U_INFO

Short Syntax: LECS.010 LECS: stopped

Long Syntax: LECS.010 LECS: stopped

Description: The LECS operation has been stopped.

LECS.011

Level: U_INFO

Short Syntax: LECS.011 LECS: restarting

Long Syntax: LECS.011 LECS: restarting

Description: The LECS operation is being restarted.

LECS.012

Level: UE_ERROR

Short Syntax: LECS.012 LECS: set fld: no LECS

Long Syntax: LECS.012 LECS: set failed: no LECS

Description: An attempt was made to set a parameter of the LECS when the LECS does not exist.

LECS.013

Level: UE_ERROR

Short Syntax: LECS.013 LECS: set fld: invld parm

Long Syntax: LECS.013 LECS: set failed: invalid parameter

Description: An attempt was made to set a parameter of the LECS using an invalid parameter identifier.

Action: Contact your customer service representative.

LECS.014

Level: UE_ERROR

Short Syntax: LECS.014 LECS: crt ELAN ' *elan_name*' fld: dplct ELAN nm

Long Syntax: LECS.014 LECS: create ELAN ' *elan_name*' failed: duplicate ELAN name

Description: The user is attempting to create an ELAN at the LECS using an ELAN name which already exists at the LECS.

LECS.015

Level: UI_ERROR

Short Syntax: LECS.015 LECS: crt ELAN ' *elan_name*' fld: mem alloc err

Long Syntax: LECS.015 LECS: create ELAN ' *elan_name*' failed: memory allocation error

Description: A memory allocation error occured while attempting to create an ELAN at the LECS.

Action: Contact your customer service representative.

LECS.016

Level: UI_ERROR

Short Syntax: LECS.016 LECS: invld crrltr on upcall ' *upcall_descriptor_string*'

Long Syntax: LECS.016 LECS: invalid correlator on upcall ' *upcall_descriptor_string*'

Description: The ATM interface has issued an upcall to the LECS using an invalid user correlator.

Action: Contact your customer service representative.

LECS.017

Level: U_INFO

Short Syntax: LECS.017 LECS: ELAN ' *elan_name*' crtd

Long Syntax: LECS.017 LECS: ELAN ' *elan_name*' created

Description: The specified ELAN was created at the LECS.

LECS.018

Level: U_INFO

Short Syntax: LECS.018 LECS: ELAN ' *elan_name*' dltd

Long Syntax: LECS.018 LECS: ELAN ' *elan_name*' deleted

Description: The specified ELAN was deleted at the LECS.

LECS.019

Level: UE_ERROR

Short Syntax: LECS.019 LECS: crt plcy fld: invld plcy type x *policy_type*

Long Syntax: LECS.019 LECS: create policy failed: invalid policy type x *policy_type*

Description: The user attempted to create a policy using an invalid policy type.

LECS.020

Level: UE_ERROR

Short Syntax: LECS.020 LECS: crt plcy fld: invld plcy prrty *policy_priority*

Long Syntax: LECS.020 LECS: create policy failed: invalid policy priority *policy_priority*

Description: The user attempted to create a policy using an invalid policy priority.

LECS.021

Level: UI_ERROR

Short Syntax: LECS.021 LECS: crt plcy fld: mem alloc err: tp x *policy_type* prrty *policy_priority*

Long Syntax: LECS.021 LECS: create policy failed: memory allocation error: type x *policy_type* priority *policy_priority*

Description: The LECS was unable to allocate the memory required to create the policy.

Action: Contact your customer service representative.

LECS.022

Level: C_INFO

Short Syntax: LECS.022 LECS: plcy x *policy_type* crtd at prrty *policy_priority*

Long Syntax: LECS.022 LECS: policy x *policy_type* created at priority *policy_priority*

Description: The specified policy was created at the specified priority at the LECS

LECS.023

Level: C_INFO

Short Syntax: LECS.023 LECS: plcy x *policy_type* dltd at prrty *policy_priority*

Long Syntax: LECS.023 LECS: policy x *policy_type* deleted at priority *policy_priority*

Description: The specified policy was deleted at the specified priority from the LECS

LECS.024

Level: UI_ERROR

Short Syntax: LECS.024 LECS: crt plcy val fld: mem alloc err: *pol_value_type_description pol_value*

Long Syntax: LECS.024 LECS: create policy value failed: memory allocation error: *pol_value_type_description pol_value*

Description: The LECS was unable to allocate the memory required to create the specified policy value.

Action: Contact your customer service representative.

LECS.025

Level: UE_ERROR

Short Syntax: LECS.025 LECS: crt plcy val fld: val exsts: *pol_value_type_description pol_value*

Long Syntax: LECS.025 LECS: create policy value failed: value already exists: *pol_value_type_description pol_value*

Description: The specified policy value already exists at the LECS.

LECS.026

Level: UI_ERROR

Short Syntax: LECS.026 LECS: crt plcy val fld: dbase err: *pol_value_type_description pol_value*

Long Syntax: LECS.026 LECS: create policy value failed: database error: *pol_value_type_description pol_value*

Description: The LECS was unable to create the policy value because of an internal database error.

Action: Contact your customer service representative.

LECS.027

Level: UE_ERROR

Short Syntax: LECS.027 LECS: crt plcy val fld: incompat val: *pol_val_type_description pol_value*

Long Syntax: LECS.027 LECS: create policy value failed: incompatible value: *pol_val_type_description pol_value*

Description: The LECS was unable to create the policy value because the value specified was incompatible with the given ELAN. Either there was a conflict between the ELAN type given and the type of the specified ELAN, or there was a conflict between the frame size given and the maximum frame size of the specified ELAN.

LECS.028

Level: C_INFO

Short Syntax: LECS.028 LECS: crtd ATM pref pol val: x *atm_prefix_pv* => x *les_atm_addr*

Long Syntax: LECS.028 LECS: created ATM prefix policy value: x *atm_prefix_pv* => x *les_atm_addr*

Description: The LECS successfully created the specified policy value, binding it to the specified LES.

LECS.029

Level: C_INFO

Short Syntax: LECS.029 LECS: crtd MAC addr pol val: x *mac_address_pv* => x *les_atm_addr*

Long Syntax: LECS.029 LECS: created MAC address policy value: x *mac_address_pv* => *les_atm_addr*

Description: The LECS successfully created the specified policy value, binding it to the specified LES.

LECS.030

Level: C_INFO

Short Syntax: LECS.030 LECS: crtd rte desc pol val: x *rte_descriptor_pv* => x *les_atm_addr*

Long Syntax: LECS.030 LECS: created route descriptor policy value: x *rte_descriptor_pv* => x *les_atm_addr*

Description: The LECS successfully created the specified policy value, binding it to the specified LES.

LECS.031

Level: C_INFO

Short Syntax: LECS.031 LECS: crtd LAN type pol val: *lan_type_pv* => x *les_atm_addr*

Long Syntax: LECS.031 LECS: created LAN type policy value: *lan_type_pv* => x *les_atm_addr*

Description: The LECS successfully created the specified policy value, binding it to the specified LES.

LECS.032

Level: C_INFO

Short Syntax: LECS.032 LECS: crtd max frm sz pol val: *frame_size_pv* => x *les_atm_addr*

Long Syntax: LECS.032 LECS: created maximum frame size policy value: *frame_size_pv* => x *les_atm_addr*

Description: The LECS successfully created the specified policy value, binding it to the specified LES.

LECS.033

Level: C_INFO

Short Syntax: LECS.033 LECS: crtd ELAN nm pol val: ' *elan_name_pv*' => x *les_atm_addr*

Long Syntax: LECS.033 LECS: created ELAN name policy value: ' *elan_name_pv*' => x *les_atm_addr*

Description: The LECS successfully created the specified policy value, binding it to the specified LES.

LECS.034

Level: UE_ERROR

Short Syntax: LECS.034 LECS: dltd ATM pref pol val: val not exst x *atm_prefix_pv* => x *les_atm_addr*

Long Syntax: LECS.034 LECS: deleted ATM prefix policy value: value did not exist x *atm_prefix_pv* => x *les_atm_addr*

Description: The LECS successfully deleted the binding between the policy value and the LES because the value did not exist at the LECS.

LECS.035

Level: UE_ERROR

Short Syntax: LECS.035 LECS: dltd ATM pref pol val: bad LES addr x *atm_prefix_pv* => x *les_atm_addr*

Long Syntax: LECS.035 LECS: deleted ATM prefix policy value: bad LES address x *atm_prefix_pv* => x *les_atm_addr*

Description: The LECS successfully deleted the binding between the policy value and the LES because the specified value is bound to a LES other than the specified LES.

LECS.036

Level: C_INFO

Short Syntax: LECS.036 LECS: dltd ATM pref pol val: x *atm_prefix_pv* => x *les_atm_addr*

Long Syntax: LECS.036 LECS: deleted ATM prefix policy value: x *atm_prefix_pv* => x *les_atm_addr*

Description: The LECS successfully deleted the binding between the policy value and the specified LES.

LECS.037

Level: UE_ERROR

Short Syntax: LECS.037 LECS: dltd MAC addr pol val: val not exst x *mac_address_pv* => x *les_atm_addr*

Long Syntax: LECS.037 LECS: deleted MAC address policy value: value did not exist x *mac_address_pv* => x *les_atm_addr*

Description: The LECS successfully deleted the binding between the policy value and the LES because the value did not exist at the LECS.

LECS.038

Level: UE_ERROR

Short Syntax: LECS.038 LECS: dltd MAC addr pol val: bad LES addr x *mac_address_pv* => x *les_atm_addr*

Long Syntax: LECS.038 LECS: deleted MAC address policy value: bad LES address x *mac_address_pv* => x *les_atm_addr*

Description: The LECS successfully deleted the binding between the policy value and the LES because the specified value is bound to a LES other than the specified LES.

LECS.039

Level: C_INFO

Short Syntax: LECS.039 LECS: dltd MAC addr pol val: x *mac_address_pv* => x *les_atm_addr*

Long Syntax: LECS.039 LECS: deleted MAC address policy value: x *mac_address_pv* => x *les_atm_addr*

Description: The LECS successfully deleted the binding between the policy value and the specified LES.

LECS.040

Level: UE_ERROR

Short Syntax: LECS.040 LECS: dltd rte desc pol val: val not exst x *rte_descriptor_pv* => x *les_atm_addr*

Long Syntax: LECS.040 LECS: deleted route descriptor policy value: bad LES addr x *rte_descriptor_pv* => x *les_atm_addr*

Description: The LECS successfully deleted the binding between the policy value and the LES because the value did not exist at the LECS.

LECS.041

Level: UE_ERROR

Short Syntax: LECS.041 LECS: dltd rte desc pol val: bad LES addr x *rte_descriptor_pv* => x *les_atm_addr*

Long Syntax: LECS.041 LECS: deleted route descriptor policy value: bad LES address x *rte_descriptor_pv* => x *les_atm_addr*

Description: The LECS successfully deleted the binding between the policy value and the LES because the specified value is bound to a LES other than the specified LES.

LECS.042

Level: C_INFO

Short Syntax: LECS.042 LECS: dltd rte desc pol val: x *rte_descriptor_pv* => x *les_atm_addr*

Long Syntax: LECS.042 LECS: deleted route descriptor policy value: x *rte_descriptor_pv* => x *les_atm_addr*

Description: The LECS successfully deleted the binding between the policy value and the specified LES.

LECS.043

Level: C_INFO

Short Syntax: LECS.043 LECS: dltd LAN type pol val: *lan_type_pv* => x *les_atm_addr*

Long Syntax: LECS.043 LECS: deleted LAN type policy value: *lan_type_pv* => x *les_atm_addr*

Description: The LECS successfully deleted the binding between the policy value and the LES.

LECS.044

Level: C_INFO

Short Syntax: LECS.044 LECS: dltd max frm sz pol val: *frame_size_pv* => x *les_atm_addr*

Long Syntax: LECS.044 LECS: deleted maximum frame size policy value: *frame_size_pv* => x *les_atm_addr*

Description: The LECS successfully deleted the binding between the policy value and the LES.

LECS.045

Level: UE_ERROR

Short Syntax: LECS.045 LECS: dltd ELAN nm pol val: val not exst ′ *elan_name_pv*′ => x *les_atm_addr*

Long Syntax: LECS.045 LECS: deleted ELAN name policy value: value did not exist ′ *elan_name_pv*′ => x *les_atm_addr*

Description: The LECS successfully deleted the binding between the policy value and the LES because the value did not exist at the LECS.

LECS.046

Level: UE_ERROR

Short Syntax: LECS.046 LECS: dltd ELAN nm pol val: bad LES addr ′ *elan_name_pv*′ => x *les_atm_addr*

Long Syntax: LECS.046 LECS: deleted ELAN name policy value: bad LES address ′ *elan_name_pv*′ => x *les_atm_addr*

Description: The LECS successfully deleted the binding between the policy value and the LES because the specified value is bound to a LES other than the specified LES.

LECS.047

Level: C_INFO

Short Syntax: LECS.047 LECS: dltd ELAN nm pol val: ′ *elan_name_pv*′ => x *les_atm_addr*

Long Syntax: LECS.047 LECS: deleted ELAN name policy value: ′ *elan_name_pv*′ => x *les_atm_addr*

Description: The LECS successfully deleted the binding between the policy value and the specified LES.

LECS.048

Level: UE_ERROR

Short Syntax: LECS.048 LECS: *operation_descrip_string*: ELAN nm ′ *elan_name*′ not exst

Long Syntax: LECS.048 LECS: *operation_descrip_string*: ELAN name ′ *elan_name*′ does not exist

Description: A lookup for the specified ELAN name failed during the given operation.

LECS.049

Level: UI_ERROR

Short Syntax: LECS.049 LECS: crt TLV failed: tp x *tlv_type* len *tlv_length* val *tlv_value*: for ELAN ′ *elan_name*′: mem alloc err

Long Syntax: LECS.049 LECS: create TLV failed: type x *tlv_type* length *tlv_length* value *tlv_value*: for ELAN ′ *elan_name*′: memory allocation error

Description: The LECS failed in attempting to allocate memory for the TLV.

Action: Contact your customer service representative.

LECS.050

Level: C_INFO

Short Syntax: LECS.050 LECS: crtd TLV: tp x *tlv_type* len *tlv_length* val x *tlv_value*: for ELAN ′ *elan_name*′

Long Syntax: LECS.050 LECS: created TLV: type x *tlv_type* length *tlv_length* value x *tlv_value*: for ELAN ′ *elan_name*′

Description: The LECS created the specified TLV for the given ELAN.

LECS.051

Level: C_INFO

Short Syntax: LECS.051 LECS: dltd TLV: tp x *tlv_type* len *tlv_length* val *tlv_value*: for ELAN ′ *elan_name*′

Long Syntax: LECS.051 LECS: deleted TLV: type x *tlv_type* length *tlv_length* value *tlv_value*: for ELAN ′ *elan_name*′

Description: The LECS deleted the specified TLV from the ELAN.

LECS.052

Level: UI_ERROR

Short Syntax: LECS.052 LECS: trmntng *error_string* (*error_code*)

Long Syntax: LECS.052 LECS: terminating *error_string* (*error_code*)

Description: LECS is being terminated because of the specified reason.

Action: Contact your customer service representative.

LECS.053

Level: U_INFO

Short Syntax: LECS.053 LECS: rlsng idle vccs

Long Syntax: LECS.053 LECS: releasing idle vccs

Description: LECS has exceeded its maximum number of VCCs and is attempting to release VCCs which have not been recently used.

LECS.054

Level: UI_ERROR

Short Syntax: LECS.054 LECS: rls idle vccs fld: *error_string* (*error_code*)

Long Syntax: LECS.054 LECS: release idle vccs failed: *error_string* (*error_code*)

Description: LECS failed in attempting to get the current time from the ATM device. Idle VCCs were not able to be released.

Action: Contact your customer service representative.

LECS.055

Level: C_INFO

Short Syntax: LECS.055 LECS: rlsd idle vcc to x *atm_address*

Long Syntax: LECS.055 LECS: released idle vcc to x *atm_address*

Description: LECS determined that the connection to the given ATM address was idle and released it.

LECS.056

Level: UI_ERROR

Short Syntax: LECS.056 LECS: ATM user reg fld: *error_string* (*error_code*)

Long Syntax: LECS.056 LECS: ATM user registration failed: *error_string* (*error_code*)

Description: LECS was unable to register as a user of ATM.

Action: Contact your customer service representative.

LECS.057

Level: U_INFO

Short Syntax: LECS.057 LECS: wtng for ATM net up

Long Syntax: LECS.057 LECS: waiting for ATM net up

Description: ATM interface is down, waiting for net up

LECS.058

Level: U_INFO

Short Syntax: LECS.058 LECS: wtng for ATM addr act

Long Syntax: LECS.058 LECS: waiting for ATM address activation

Description: ATM address activation has not yet completed

LECS.059

Level: UI_ERROR

Short Syntax: LECS.059 LECS: ATM addr act fld: *error_string* (*error_code*)

Long Syntax: LECS.059 LECS: ATM address activation failed: *error_string* (*error_code*)

Description: ATM address activation has failed for the LECS.

Action: Contact your customer service representative.

LECS.060

Level: UI_ERROR

Short Syntax: LECS.060 LECS: unbl to get ATM addr: *error_string* (*error_code*)

Long Syntax: LECS.060 LECS: unable to get ATM address: *error_string* (*error_code*)

Description: The LECS was unable to get its ATM address.

Action: Contact your customer service representative.

LECS.061

Level: U_INFO

Short Syntax: LECS.061 LECS: ATM addr: x *atm_address*

Long Syntax: LECS.061 LECS: ATM address: x *atm_address*

Description: The LECS has retrieved its ATM address.

LECS.062

Level: U_INFO

Short Syntax: LECS.062 LECS: wtng for UNI vrsn rpt

Long Syntax: LECS.062 LECS: waiting for UNI version report

Description: The LECS is waiting to be informed of the UNI version.

LECS.063

Level: U_INFO

Short Syntax: LECS.063 LECS: UNI vrsn *uni_version* rptd

Long Syntax: LECS.063 LECS: UNI version *uni_version* reported

Description: The LECS is operating under the specified UNI version.

LECS.064

Level: UI_ERROR

Short Syntax: LECS.064 LECS: unbl to open frame SAP: *error_string* (*error_code*)

Long Syntax: LECS.064 LECS: unable to open frame SAP: *error_string* (*error_code*)

Description: The LECS failed to open a frame SAP.

Action: Contact your customer service representative.

LECS.065

Level: UI_ERROR

Short Syntax: LECS.065 LECS: unbl to open call SAP: *error_string* (*error_code*)

Long Syntax: LECS.065 LECS: unable to open call SAP: *error_string* (*error_code*)

Description: The LECS failed to open a call SAP.

Action: Contact your customer service representative.

LECS.066

Level: U_INFO

Short Syntax: LECS.066 LECS: cmpltd intzltn

Long Syntax: LECS.066 LECS: completed initialization

Description: The LECS has completed initialization and is completely operational.

LECS.067

Level: UE_ERROR

Short Syntax: LECS.067 LECS: frm dscrdd: *discard_reason*

Long Syntax: LECS.067 LECS: frame discarded: *discard_reason*

Description: The LECS has discarded an incoming frame for the specified reason.

LECS.068

Level: UE_ERROR

Short Syntax: LECS.068 LECS: invld lecid: *lecid*

Long Syntax: LECS.068 LECS: invalid lecid: *lecid*

Description: The incoming frame had an invalid LEC-ID value.

LECS.069

Level: UE_ERROR

Short Syntax: LECS.069 LECS: invld src LAN dest: x *source_lan_dest_string*

Long Syntax: LECS.069 LECS: invalid source LAN destination: x *source_lan_dest_string*

Description: The incoming frame had an invalid source LAN destination.

LECS.070

Level: UE_ERROR

Short Syntax: LECS.070 LECS: invld src ATM addr: x *source_atm_addr_string*

Long Syntax: LECS.070 LECS: invalid source ATM address: x *source_atm_addr_string*

Description: The incoming frame had an invalid source ATM address.

LECS.071

Level: UE_ERROR

Short Syntax: LECS.071 LECS: invld ELAN typ: *requested_lan_type*

Long Syntax: LECS.071 LECS: invalid ELAN type: *requested_lan_type*

Description: The incoming frame had an invalid ELAN type field.

LECS.072

Level: UE_ERROR

Short Syntax: LECS.072 LECS: invld max frm sz: *requested_mfs*

Long Syntax: LECS.072 LECS: invalid maximum frame size: *requested_mfs*

Description: The incoming frame had an invalid maximum frame size field.

LECS.073

Level: C_INFO

Short Syntax: LECS.073 LECS: frm pssd vldtn chcks

Long Syntax: LECS.073 LECS: frame passed validation checks

Description: The incoming frame passed all frame validation tests.

LECS.074

Level: C_INFO

Short Syntax: LECS.074 LECS: LEC x *lec_atm_addr* assgnd to LES x *les_atm_addr* at *priority* usng *policy_descript_string*

Long Syntax: LECS.074 LECS: LEC x *lec_atm_addr* assigned to LES x *les_atm_addr* at priority *priority* using policy *policy_descript_string*

Description: The LEC was assigned to the specified LES based on the given policy.

LECS.075

Level: C_INFO

Short Syntax: LECS.075 LECS: unbl to assign rqst frm x *source_atm_address*

Long Syntax: LECS.075 LECS: unable to assign request from x *source_atm_address*

Description: The LECS was unable to assign the configuration request from the given source ATM address to a LES.

LECS.076

Level: UE_ERROR

Short Syntax: LECS.076 LECS: ATM net down

Long Syntax: LECS.076 LECS: ATM net down

Description: The ATM network is now down.

LECS.077

Level: U_INFO

Short Syntax: LECS.077 LECS: ATM net up

Long Syntax: LECS.077 LECS: ATM net up

Description: The ATM network is now up.

LECS.078

Level: U_INFO

Short Syntax: LECS.078 LECS: ATM addr actvtd

Long Syntax: LECS.078 LECS: ATM address activated

Description: The ATM address of the LECS has been activated.

LECS.079

Level: UE_ERROR

Short Syntax: LECS.079 LECS: ATM addr actvtn tmd out: retrying

Long Syntax: LECS.079 LECS: ATM address activation timed out: retrying

Description: The activation of the LECS' ATM address has timed out. Activation will be retried.

LECS.080

Level: UE_ERROR

Short Syntax: LECS.080 LECS: ATM addr rjctd

Long Syntax: LECS.080 LECS: ATM address rejected

Description: The ATM address of the LECS has been rejected. Another attempt will be made to activate the ATM address.

LECS.081

Level: UE_ERROR

Short Syntax: LECS.081 LECS: ATM addr dctvted: reactivating

Long Syntax: LECS.081 LECS: ATM address deactivated: reactivating

Description: The ATM address of the LECS has been deactivated. The LECS is attempting to reactivate it.

LECS.082

Level: U_INFO

Short Syntax: LECS.082 LECS: UNI vrsn rptd

Long Syntax: LECS.082 LECS: UNI version reported

Description: The UNI version was reported.

LECS.083

Level: UI_ERROR

Short Syntax: LECS.083 LECS: invld upcall: *upcall_descriptor_string*

Long Syntax: LECS.083 LECS: invalid upcall from ATM: *upcall_descriptor_string*

Description: ATM has called an invalid or unexpected LECS procedure.

Action: Contact your customer service representative.

LECS.084

Level: UE_ERROR

Short Syntax: LECS.084 LECS: rfsd cfgtn drct: invld parms

Long Syntax: LECS.084 LECS: refused configuration direct: invalid parameters

Description: The LECS has refused an incoming configuration direct VCC due to invalid parameters.

LECS.085

Level: UE_ERROR

Short Syntax: LECS.085 LECS: rfsd cfgtn drct frm x *calling_atm_address*: invld parms

Long Syntax: LECS.085 LECS: refused configuration direct from x *calling_atm_address*: invalid parameters

Description: The LECS has refused an incoming configuration direct VCC from the given ATM address due to invalid parameters. To gain more information on why the VCC was rejected, LES ELS messages must be displayed. The relevant LES ELS messages are LES.002 through LES.040.

LECS.086

Level: UE_ERROR

Short Syntax: LECS.086 LECS: rfsd cfgtn drct frm x *calling_atm_address*: at max VCCs

Long Syntax: LECS.086 LECS: refused configuration direct from x *calling_atm_address*: at maximum VCCs

Description: The LECS has refused an incoming configuration direct VCC from the given ATM address because it is already at its maximum number of VCCs. The LECS attempted, and failed, to locate and release idle VCCs.

LECS.087

Level: UI_ERROR

Short Syntax: LECS.087 LECS: rfsd cfgtn drct frm x *calling_atm_address*: mem alloc err

Long Syntax: LECS.087 LECS: refused configuration direct from x *calling_atm_address*: memory allocation error

Description: The LECS has refused an incoming configuration direct VCC from the given ATM address due to a memory allocation error.

Action: Contact your customer service representative.

LECS.088

Level: UI_ERROR

Short Syntax: LECS.088 LECS: rfsd cfgtn drct frm x *calling_atm_address*: unble to get time

Long Syntax: LECS.088 LECS: refused configuration direct from x *calling_atm_address*: unable to get time

Description: The LECS has refused an incoming configuration direct VCC from the given ATM address due to an inability to determine the time.

Action: Contact your customer service representative.

LECS.089

Level: UI_ERROR

Short Syntax: LECS.089 LECS: rfsd cfgtn drct frm x *calling_atm_address*: opn data path err: no ATM mem

Long Syntax: LECS.089 LECS: refused configuration direct from x *calling_atm_address*: open data path error: no ATM memory

Description: The LECS has refused an incoming configuration direct VCC from the given ATM address due to an inability to open a data path to the caller. The data path failure was due to a lack of memory in ATM.

Action: Contact your customer service representative.

LECS.090

Level: UI_ERROR

Short Syntax: LECS.090 LECS: rfsd cfgtn drct frm x *calling_atm_address*: opn data path err: *error_code_string* (*error_code*)

Long Syntax: LECS.090 LECS: refused configuration direct from x *calling_atm_address*: open data path error: *error_code_string* (*error_code*)

Description: The LECS has refused an incoming configuration direct VCC from the given ATM address due to an inability to open a data path to the caller.

Action: Contact your customer service representative.

LECS.091

Level: UI_ERROR

Short Syntax: LECS.091 LECS: rfsd cfgtn drct frm x *calling_atm_address*: rcv ack err: no SVC mem

Long Syntax: LECS.091 LECS: refused configuration direct from x *calling_atm_address*: receive ack error: no SVC memory

Description: The LECS has refused an incoming configuration direct VCC from the given ATM address due to an inability to acknowledge the call. The acknowledgment failure was due to a lack of memory in SVC.

Action: Contact your customer service representative.

LECS.092

Level: UI_ERROR

Short Syntax: LECS.092 LECS: rfsd cfgtn drct frm x *calling_atm_address*: rcv ack err: *error_code_string* (*error_code*)

Long Syntax: LECS.092 LECS: refused configuration direct from x *calling_atm_address*: rcv ack error: *error_code_string* (*error_code*)

Description: The LECS has refused an incoming configuration direct VCC from the given ATM address due to an inability to acknowledge the call.

Action: Contact your customer service representative.

LECS.093

Level: C_INFO

Short Syntax: LECS.093 LECS: cfgtn drct frm x *calling_atm_address* estblshd

Long Syntax: LECS.093 LECS: configuration direct from x *calling_atm_address* established

Description: The LECS has established a configuration direct from the given caller.

LECS.094

Level: C_INFO

Short Syntax: LECS.094 LECS: cfgtn drct frm x *calling_atm_address* dscnnctd

Long Syntax: LECS.094 LECS: configuration direct from x *calling_atm_address* disconnected

Description: The configuration direct VCC from the specified ATM address has been disconnected.

LECS.095

Level: C_INFO

Short Syntax: LECS.095 LECS: sent config response to x *source_atm_address*

Long Syntax: LECS.095 LECS: sent configuration response to x *source_atm_address*

Description: The LECS has transmitted a configuration response using the specified source ATM address.

LECS.096

Level: UI_ERROR

Short Syntax: LECS.096 LECS: invld state (*state*) for upcall ′ *upcall_descriptor_string*′

Long Syntax: LECS.096 LECS: invalid state (*state*) for upcall ′ *upcall_descriptor_string*′

Description: The ATM interface has issued an upcall to the LECS using an invalid user correlator.

Action: Contact your customer service representative.

LECS.097

Level: UE_ERROR

Short Syntax: LECS.097 LECS: *operation_descrip_string*: LES addr *les_atm_addr* not exst

Long Syntax: LECS.097 LECS: *operation_descrip_string*: LES ATM address *les_atm_addr* does not exist

Description: A lookup for the specified LES ATM address failed during the given operation.

LECS.098

Level: UI_ERROR

Short Syntax: LECS.098 LECS: crt LES *les_atm_addr* fld: mem alloc err

Long Syntax: LECS.098 LECS: create LES *les_atm_addr* failed: memory allocation error

Description: A memory allocation error occured while attempting to create a LES at the LECS.

Action: Contact your customer service representative.

LECS.099

Level: UE_ERROR

Short Syntax: LECS.099 LECS: crt LES *les_atm_addr* fld: dplct LES addr

Long Syntax: LECS.099 LECS: create LES *les_atm_addr* failed: duplicate LES ATM address

Description: The user is attempting to create a LES using a LES ATM address which already exists.

LECS.100

Level: UI_ERROR

Short Syntax: LECS.100 LECS: crt LES *les_atm_addr* fld: dbase err

Long Syntax: LECS.100 LECS: create LES *les_atm_addr* failed: database error

Description: A database error occured while attempting to add the LES address to the LECS databases.

Action: Contact your customer service representative.

LECS.101

Level: U_INFO

Short Syntax: LECS.101 LECS: LES crtd x *les_atm_addr* => ′ *elan_name*′

Long Syntax: LECS.101 LECS: LES created x *les_atm_addr* => ′ *elan_name*′

Description: The specified LES was created and bound to the given ELAN.

LECS.102

Level: UE_ERROR

Short Syntax: LECS.102 LECS: dltd LES: val not exst x *les_atm_addr* => ′ *elan_name*′

Long Syntax: LECS.102 LECS: deleted LES: value did not exist x *les_atm_addr* => ′ *elan_name*′

Description: The LECS successfully deleted the binding between LES and the ELAN because the specified LES ATM address did not exist at the LECS.

LECS.103

Level: UE_ERROR

Short Syntax: LECS.103 LECS: dlt LES *les_atm_addr* fld: bad ELAN nm ′ *elan_name*′

Long Syntax: LECS.103 LECS: delete of LES *les_atm_addr* failed: bad ELAN name ′ *elan_name*′

Description: The user attempted to delete a LES from an ELAN when that LES was not associated with that ELAN.

LECS.104

Level: U_INFO

Short Syntax: LECS.104 LECS: LES dltd: x *les_atm_addr* => ′ *elan_name*′

Long Syntax: LECS.104 LECS: LES deleted: x *les_atm_addr* => ′ *elan_name*′

Description: The specified LES was deleted from the given ELAN at the LECS.

LECS.105

Level: UI_ERROR

Short Syntax: LECS.105 LECS: crt ELAN ′ *elan_name*′ fld: dbase err

Long Syntax: LECS.105 LECS: create ELAN ′

elan_name' failed: database error

Description: A database error occured while attempting to add the ELAN to the the LECS databases.

Action: Contact your customer service representative.

LECS.106

Level: C_INFO

Short Syntax: LECS.106 LECS: incmng call: *local_or_wk_address*

Long Syntax: LECS.106 LECS: incoming call: *local_or_wk_address*

Description: The LECS received an incoming call for either the local address, or for the LECS well-known address as specified in the LAN emulation specification.

LECS.107

Level: C_INFO

Short Syntax: LECS.107 LECS: addng LEC addr to mem: *lec_atm_addr*: LES *les_atm_addr* time *current_time*

Long Syntax: LECS.107 LECS: adding LEC ATM address to memory: *lec_atm_addr*: LES *les_atm_addr* current time *current_time*

Description: The LEC ATM address was added to the LECS short-term memory. The LEC was assigned to the specified LES at the given time.

LECS.108

Level: C_INFO

Short Syntax: LECS.108 LECS: dltng LEC addr frm mem: *lec_atm_addr* time *current_time*

Long Syntax: LECS.108 LECS: deleting LEC ATM address from memory: *lec_atm_addr* current time *current_time*

Description: The LEC ATM address was deleted from the LECS short-term memory at the specified time.

LECS.109

Level: C_INFO

Short Syntax: LECS.109 LECS: updtng LEC addr in mem: *lec_atm_addr* LES *les_atm_addr* time *current_time*

Long Syntax: LECS.109 LECS: updating LEC ATM address in memory: *lec_atm_addr*: LES *les_atm_addr* current time *current_time*

Description: The LEC ATM address was updated in the LECS short-term memory. The LES was last assigned to the specified LES at the given time.

LECS.110

Level: UI_ERROR

Short Syntax: LECS.110 LECS: mem add fld: unbl to get time: *lec_atm_addr*

Long Syntax: LECS.110 LECS: memory add failed: unable to get time: *lec_atm_addr*

Description: The LECS was unable to add the LEC ATM address to its memory because it was unable to get the current time.

Action: Contact your customer service representative.

LECS.111

Level: UI_ERROR

Short Syntax: LECS.111 LECS: mem updt fld: unbl to get time: *lec_atm_addr*

Long Syntax: LECS.111 LECS: memory update failed: unable to get time: *lec_atm_addr*

Description: The LECS was unable to update the LEC ATM address in its memory because it was unable to get the current time.

Action: Contact your customer service representative.

LECS.112

Level: UI_ERROR

Short Syntax: LECS.112 LECS: mem lkup fld: unbl to get time: *lec_atm_addr*

Long Syntax: LECS.112 LECS: memory lookup failed: unable to get time: *lec_atm_addr*

Description: The LECS was unable to search for a LEC ATM address in its memory because it was unable to get the current time.

Action: Contact your customer service representative.

LECS.113

Level: UI_ERROR

Short Syntax: LECS.113 LECS: mem add fld: mem alloc err: *lec_atm_addr*

Long Syntax: LECS.113 LECS: memory add failed: memory allocation error: *lec_atm_addr*

Description: The LECS was unable to add the LEC ATM address to its memory because of a memory allocation error.

Action: Contact your customer service representative.

LECS.114

Level: C_INFO

Short Syntax: LECS.114 LECS: mem lkup success: *lec_atm_addr* prim LES *primary_les_atm_addr*: last LES *last_les_atm_addr*

Long Syntax: LECS.114 LECS: memory lookup success: *lec_atm_addr* primary LES *primary_les_atm_addr*: last LES *last_les_atm_addr*

Description: The LECS found the LEC ATM address in its short term memory. The LEC is associated with the specified primary LES in its databases, and the last time the LEC contacted the LECS it was given the LES specified LES address.

LECS.115

Level: C_INFO

Short Syntax: LECS.115 LECS: mem lkup fld: *lec_atm_addr*

Long Syntax: LECS.115 LECS: memory lookup failed: *lec_atm_addr*

Description: The LECS did not find the LEC ATM address in memory. The primary LES ATM address is used in the configuration response.

LECS.116

Level: U_INFO

Short Syntax: LECS.116 LECS: mvd to nrml state

Long Syntax: LECS.116 LECS: moved to normal state

Description: The LECS has moved from the state where it rejects all incoming calls to its normal operating state.

LECS.117

Level: U_INFO

Short Syntax: LECS.117 LECS: mvd to rjct calls state

Long Syntax: LECS.117 LECS: moved to reject calls state

Description: The LECS has moved to a state where it will reject all incoming calls.

LECS.118

Level: C_INFO

Short Syntax: LECS.118 LECS: rfsd cfgtn drct frm x *calling_atm_address* rject calls state

Long Syntax: LECS.118 LECS: refused configuration direct from x *calling_atm_address* in reject call state

Description: The LECS has refused an incoming configuration direct VCC from the given ATM address because it is in a state which dictates that all VCCs are rejected.

LECS.119

Level: UI_ERROR

Short Syntax: LECS.119 LECS: ELAN ' *elan_name*': set fld: dbase err: *set_fail_reason*

Long Syntax: LECS.119 LECS: ELAN ' *elan_name*': set failed: database error: *set_fail_reason*

Description: The attempt to set the parameter of the ELAN failed due to a database error. The resources of the failed ELAN were released at the LECS.

Action: Contact your customer service representative.

LECS.120

Level: UI_ERROR

Short Syntax: LECS.120 LECS: LES x *les_atm_addr*: set fld: dbase err: *set_fail_reason*

Long Syntax: LECS.120 LECS: LES x *les_atm_addr*: set failed: database error: *set_fail_reason*

Description: The attempt to set the parameter of the LES failed due to a database error. The failed LES was released at the LECS.

Action: Contact your customer service representative.

LECS.121

Level: C_INFO

Short Syntax: LECS.121 LECS: lcl LES addr for ELAN ' *elan_name*' mapped to LES: *actual_les_atm_addr*

Long Syntax: LECS.121 LECS: local LES address for ELAN: ' *elan_name*' mapped to LES: *actual_les_atm_addr*

Description: The local LES ATM address was mapped to specified actual ATM address. The actual ATM address was obtained from a LES/BUS located on this router which serves the specified ELAN.

LECS.122

Level: U_INFO

Short Syntax: LECS.122 LECS: unbl to find local LES for ELAN ' *elan_name*' for LEC: *lec_atm_adddr*

Long Syntax: LECS.122 LECS: unable to find local LES for ELAN ' *elan_name*' for LEC: *lec_atm_adddr*

Description: The specified LEC was to be assigned to a local LES, but that local LES does not exist on the router. The configuration request for this LEC is rejected.

LECS.123

Level: U_INFO

Short Syntax: LECS.123 LECS: *wka_anycast* rgstrtn: success

Long Syntax: LECS.123 LECS: *wka_anycast* address registration: success

Description: The attempt by ILMI to register the LECS well-known address or the LECS anycast addreess with the ATM switch has succeeded.

LECS.124

Level: U_INFO

Short Syntax: LECS.124 LECS: *wka_anycast* rgstrtn: no success

Long Syntax: LECS.124 LECS: *wka_anycast* address registration: no success

Description: The attempt by ILMI to register the LECS well-known address or the LECS anycast address with the ATM switch has either failed, or has not yet succeeded. The LECS will poll the status of the well-known or anycast address again.

LECS.125

Level: U_INFO

Short Syntax: LECS.125 LECS: *wka_anycast* rgstrtn: gvng up

Long Syntax: LECS.125 LECS: *wka_anycast* address registration: giving up

Description: The attempt by ILMI to register the LECS well-known address or the LECS anycast address with the ATM switch has either failed, or has not yet succeeded. The LECS will not poll the status of the well-known or anycast address again.

LECS.126

Level: UI_ERROR

Short Syntax: LECS.126 LECS: *wka_anycast* rgstrtn err: *error_string* (*error_code*)

Long Syntax: LECS.126 LECS: *wka_anycast* address registration error: *error_string* (*error_code*)

Description: The LECS attempt to poll the status of the LECS well-known address or the LECS anycast address registration resulted in an error.

Action: Contact your customer service representative.

LECS.127

Level: UI_ERROR

Short Syntax: LECS.127 LECS: *wka_anycast* actvtn err: *error_string* (*error_code*)

Long Syntax: LECS.127 LECS: *wka_anycast* address activation error: *error_string* (*error_code*)

Description: The LECS attempt to activate the LECS well-known address or the LECS anycast address resulted in an error.

Action: Contact your customer service representative.

LECS.128

Level: P_TRACE

Short Syntax: LECS.128 Trace LECS control frames

Long Syntax: LECS.128 Trace LAN Emulation Configuration Server control frames

Description: Packet tracing for control frames to and from the LECS.

LECS.129

Level: C_INFO

Short Syntax: LECS.129 LECS: secrty req rejected for LEC *lec_atm_addr* and LES *les_atm_addr*: *rejection_reason*

Long Syntax: LECS.129 LECS: security request rejected for LEC *lec_atm_addr* and LES *les_atm_addr*: *rejection_reason*

Description: The LECS processed a security request concerning the specified LEC and LES, and this request was rejected. Reasons for rejecting a security request are: 1) ″reqstng LES not last assgned LES″ - the LEC was found in the short-term memory of the LECS, and the last LES to which it was assigned is not the requesting LES. 2) ″LES assgnmnt fld″ - the LECS has no knowledge of the LEC contacting it, and the LECS is unable to find a LES for the LEC based on the supplied information. 3) ″requstng LES not assgnd LES″ - the LECS has no knowledge of the LEC contacting it, the LECS was able to assign the LEC to a LES, but the requesting LES is not the LES that would be assigned by the LECS.

LECS.130

Level: C_INFO

Short Syntax: LECS.130 LECS: secrty req apprvd for LEC *lec_atm_addr* and LES *les_atm_addr*

Long Syntax: LECS.130 LECS: security request approved for LEC *lec_atm_addr* and LES *les_atm_addr*

Description: The LECS processed a security request concerning the specified LEC and LES, and the request was approved.

LECS.132

Level: C_INFO

Short Syntax: LECS.132 LECS: snding LEC: *lec_atm_addr* to *primary_or_backup* LES: *les_atm_addr*

Long Syntax: LECS.132 LECS: sending LEC: *lec_atm_addr* to *primary_or_backup* LES: *les_atm_addr*

Description: The specified client is being sent to the specified primary or backup LES.

LECS.133

Level: UE_ERROR

Short Syntax: LECS.133 LECS: err *error_location*: invld ATM addr mask *atm_addr_mask*

Long Syntax: LECS.133 LECS: error *error_location*: invalid ATM address mask *atm_addr_mask*

Description: The given ATM address mask is invalid. The only type of address mask currently allowed specifies a prefix of the ATM address. Thus, the mask must be a non-zero number of 0xff octets, followed by all 0x00 octets. The error occured when the user attempted to create or delete an ATM address policy value.

LECS.134

Level: C_INFO

Short Syntax: LECS.134 LECS: crtd ESI/Sel pol val: x *esi_selector* => x *les_atm_addr*

Long Syntax: LECS.134 LECS: created ESI/Selector policy value: x *esi_selector* => x *les_atm_addr*

Description: The LECS successfully created the specified policy value, binding it to the specified LES.

LECS.135

Level: UE_ERROR

Short Syntax: LECS.135 LECS: dltd ESI/Sel pol val: val not exst x *esi_selector_pv* => x *les_atm_addr*

Long Syntax: LECS.135 LECS: deleted ESI/Selector policy value: value did not exist x *esi_selector_pv* => x *les_atm_addr*

Description: The LECS successfully deleted the binding between the policy value and the LES because the value did not exist at the LECS.

LECS.136

Level: UE_ERROR

Short Syntax: LECS.136 LECS: dltd ESI/Sel pol val: bad LES addr x *esi_selector_pv* => x *les_atm_addr*

Long Syntax: LECS.136 LECS: deleted ESI/Selector policy value: bad LES address x *esi_selector_pv* => x *les_atm_addr*

Description: The LECS successfully deleted the binding between the policy value and the LES because the specified value is bound to a LES other than the specified LES.

LECS.137

Level: C_INFO

Short Syntax: LECS.137 LECS: dltd ESI/Sel pol val: x *esi_selector_pv* => x *les_atm_addr*

Long Syntax: LECS.137 LECS: deleted ESI/Selector policy value: x *esi_selector_pv* => x *les_atm_addr*

Description: The LECS successfully deleted the binding between the policy value and the specified LES.

LECS.138

Level: UI_ERROR

Short Syntax: LECS.138 LECS: crt sus ATM Addr fld: mem alloc err: *suspect_atm_address_description_string*

Long Syntax: LECS.138 LECS: create suspect ATM Addr failed: memory allocation error: *suspect_atm_address_description_string*

Description: The LECS was unable to allocate the memory required to create the specified suspect ATM address.

Action: Contact your customer service representative.

LECS.139

Level: UE_ERROR

Short Syntax: LECS.139 LECS: crt sus ATM Addr fld: val exsts: *suspect_ATM_Address_description*

Long Syntax: LECS.139 LECS: create suspect ATM Address failed: value already exists: *suspect_ATM_Address_description*

Description: The specified suspect ATM Address already exists at the LECS.

LECS.140

Level: UI_ERROR

Short Syntax: LECS.140 LECS: crt sus ATM Addr fld: dbase err: *suspect_ATM_Address_description*

Long Syntax: LECS.140 LECS: create suspect ATM Address failed: database error: *suspect_ATM_Address_description*

Description: The LECS was unable to create the suspect ATM Address because of an internal database error.

Action: Contact your customer service representative.

LECS.141

Level: U_INFO

Short Syntax: LECS.141 LECS: sus ATM Addr dltd: *suspect_ATM_Address_description*

Long Syntax: LECS.141 LECS: suspect ATM Address deleted : *suspect_ATM_Address_description*

Description: The specified suspect ATM address was deleted from the LECS.

LECS.142

Level: C_INFO

Short Syntax: LECS.142 LECS: suspect src ATM addr: x *source_atm_addr_string*

Long Syntax: LECS.142 LECS: suspect source ATM address: x *source_atm_addr_string*

Description: The incoming frame had a suspect source ATM address. This configuration or security request is rejected with cause "Access Denied."

LECS.143

Level: C_INFO

Short Syntax: LECS.143 LECS: updtd cnfgrtn for fld: ' *field_name*'

Long Syntax: LECS.143 LECS: updated configuration for field: ' *field_name*'

Description: During initialization, an outdated configuration record was discovered. Certain configuration parameters of the LECS were updated to reflect a new code release. This event is common and expected after updating to a new release of operation code.

LECS.144

Level: C_INFO

Short Syntax: LECS.144 LECS: rfsd cfgtn drct frm x *calling_atm_address* suspect ATM address

Long Syntax: LECS.144 LECS: refused configuration direct from x *calling_atm_address* in reject call state

Description: The LECS has refused an incoming configuration direct VCC from the given ATM address because the address is configured to be a suspect ATM address in the access-control database.

LECS.145

Level: UE_ERROR

Short Syntax: LECS.145 LECS: frm contains bad TLV info

Long Syntax: LECS.145 LECS: frame contains incorrect no bytes in TLV data.

Description: The incoming frame contains incorrect no bytes in TLV data.

LECS.146

Level: U_INFO

Short Syntax: LECS.146 LECS: LEC x *lec_atm_addr* last assgnd to unkwn LES, occrrd *count* times

Long Syntax: LECS.146 LECS: LEC x *lec_atm_addr* last assigned to unknown LES, has occurred *count* times

Description: The specified LEC is in the LECS' memory, but has most recently been assigned to a LES which is neither the primary nor the backup. This may be the result of a configuration change at the LEC or the LECS, or it may indicate that an edge device is incorrectly using the same source ATM address for configuring multiple LECs. If this message occurs multiple times for the same LEC with an increasing count, then the latter explanation is most likely.

LECS.147

Level: C_INFO

Short Syntax: LECS.147 LECS: crtd LEC TLV: tp x *tlv_type* len *tlv_length* val x *tlv_value*: for *policy_value_type policy_value*

Long Syntax: LECS.147 LECS: created LEC TLV: type x *tlv_type* length *tlv_length* value x *tlv_value*: for *policy_value_type policy_value*

Description: The LECS created the specified TLV for the given policy value.

LECS.148

Level: C_INFO

Short Syntax: LECS.148 LECS: dltd LEC TLV: tp x *tlv_type* len *tlv_length* val *tlv_value*: for *policy_value_type policy_value*

Long Syntax: LECS.148 LECS: deleted LEC TLV: type x *tlv_type* length *tlv_length* value *tlv_value*: for *policy_value_type policy_value*

Description: The LECS deleted the specified TLV from the specified policy value.

LECS.149

Level: UI_ERROR

Short Syntax: LECS.149 LECS: crt LEC TLV failed: tp x *tlv_type* len *tlv_length* val *tlv_value*

Long Syntax: LECS.149 LECS: create LEC TLV failed: type x *tlv_type* length *tlv_length* value *tlv_value*

Description: The LECS failed in attempting to allocate memory for the TLV.

Action: Contact your customer service representative.

LECS.150

Level: C_INFO

Short Syntax: LECS.150 LECS: crt sus ATM Addr *suspect_ATM_Address_*

Long Syntax: LECS.150 LECS: create suspect ATM Address *suspect_ATM_Address_*

Description: A suspect ATM address was created at the LECS.

LECS.151

Level: UI_ERROR

Short Syntax: LECS.151 LECS: crt TLV failed: type x *tlv_type* len *tlv_length*

Long Syntax: LECS.151 LECS: create TLV failed: type x *tlv_type* length *tlv_length*

Description: The LECS failed in attempting to allocate memory for a TLV that contains an encoded LECS SRAM record to be sent to a remote LECS as part of the LECS Database Synchronization

Action: Contact your customer service representative.

LECS.152

Level: UI_ERROR

Short Syntax: LECS.152 LECS: error alloc mem for Dbase Sync ATM addr: *dbase_sync_atm_address_string*

Long Syntax: LECS.152 LECS: error allocating memory for Dbase Sync ATM addr: *dbase_sync_atm_address_string*

Description: The LECS was unable to allocate the memory required to create the specified remote LECS ATM address for Database Sync.

Action: Contact your customer service representative.

LECS.153

Level: UE_ERROR

Short Syntax: LECS.153 LECS: remote LECS ATM addr already exists: *dbase_sync_ATM_Address_string*

Long Syntax: LECS.153 LECS: remote LECS ATM addr already exists: *dbase_sync_ATM_Address_string*

Description: The specified remote LECS ATM Address for LECS Database Synchronization already exists at the LECS.

LECS.154

Level: UI_ERROR

Short Syntax: LECS.154 LECS: dbase err for Dbase Sync ATM address: *dbase_sync_ATM_Address_string*

Long Syntax: LECS.154 LECS: dbase err for Dbase Sync ATM address: *dbase_sync_ATM_Address_string*

Description: The LECS was unable to create the Dbase Sync ATM Address because of an internal database error.

Action: Contact your customer service representative.

LECS.155

Level: C_INFO

Short Syntax: LECS.155 LECS: create Dbase Sync Addr: *dbase_sync_ATM_Address_string*

Long Syntax: LECS.155 LECS: create Dbase Sync Addr: *dbase_sync_ATM_Address_string*

Description: An LECS Database Synchronization ATM address was created at the LECS.

LECS.156

Level: U_INFO

Short Syntax: LECS.156 LECS: dbase sync ATM Addr deleted: *dbase_sync_ATM_Address_string*

Long Syntax: LECS.156 LECS: dbase sync ATM Addr deleted: *dbase_sync_ATM_Address_string*

Description: The specified LECS Database Sync ATM address was deleted from the LECS.

LECS.157

Level: C_INFO

Short Syntax: LECS.157 LECS: attempting Dbase Sync VCC to: *dbase_sync_ATM_Address_string*

Long Syntax: LECS.157 LECS: attempting setup of Dbase Sync VCC to: *dbase_sync_ATM_Address_string*

Description: An LECS Database Synchronization VCC is being attempted to the indicated remote LECS ATM address.

LECS.158

Level: UE_ERROR

Short Syntax: LECS.158 LECS: *error_message*, rc=x *error_code*: err in Dbase Sync VCC to: *dbase_sync_ATM_Address_string*

Long Syntax: LECS.158 LECS: *error_message*, rc=x *error_code*: err in Dbase Sync VCC to: *dbase_sync_ATM_Address_string*

Description: An LECS Database Synchronization VCC cannot be setup to the indicated remote LECS ATM address due to indicated error.

LECS.159

Level: UI_ERROR

Short Syntax: LECS.159 LECS: error alloc mem for Dbase Sync frame, rc=x *error_code*: *dbase_sync_atm_address_string*

Long Syntax: LECS.159 LECS: error alloc mem for Dbase Sync frame, rc=x *error_code*: *dbase_sync_atm_address_string*

Description: The LECS was unable to allocate the memory required to create the specified remote LECS ATM address for Database Sync.

Action: Contact your customer service representative.

LECS.160

Level: UE_ERROR

Short Syntax: LECS.160 LECS: reject Dbase Sync VCC (not allowed) from: *calling_atm_address*

Long Syntax: LECS.160 LECS: reject Dbase Sync VCC (not allowed) from: *calling_atm_address*

Description: The LECS has refused an incoming LECS Database Sync VCC from the given LECS ATM address because this LECS has been configured to not allow remote configuration.

LECS.161

Level: UE_ERROR

Short Syntax: LECS.161 LECS: reject Dbase Sync VCC (already exists) from: *calling_atm_address*

Long Syntax: LECS.161 LECS: reject Dbase Sync VCC (already exists) from: *calling_atm_address*

Description: The LECS has refused an incoming LECS Database Sync VCC from the given LECS ATM address because another Dbase Sync VCC already exists, and only one is allowed at a time.

LECS.162

Level: UI_ERROR

Short Syntax: LECS.162 LECS: rfsd dbase sync frm x *calling_atm_address*: rcv ack err: *error_code_string* (*error_code*)

Long Syntax: LECS.162 LECS: refused database sync vcc from x *calling_atm_address*: rcv ack error: *error_code_string* (*error_code*)

Description: The LECS has refused an incoming LECS Database Sync VCC from the given ATM address due to an inability to acknowledge the call.

Action: Contact your customer service representative.

LECS.163

Level: C_INFO

Short Syntax: LECS.163 LECS: dbase sync vcc setup from remote LECS: *calling_atm_address*

Long Syntax: LECS.163 LECS: dbase sync vcc setup from remote LECS: *calling_atm_address*

Description: The LECS Database Synchronization VCC has been successfully established from a remote LECS with the indicated ATM address.

LECS.164

Level: UI_ERROR

Short Syntax: LECS.164 LECS: rfsd dbase sync frm x *calling_atm_address*: open data path err: *error_code_string* (*error_code*)

Long Syntax: LECS.164 LECS: refused database sync vcc from x *calling_atm_address*: open data path error: *error_code_string* (*error_code*)

Description: The LECS has refused an incoming LECS Database Sync VCC from the given ATM address due to an inability to open a data path to the caller.

Action: Contact your customer service representative.

LECS.165

Level: UI_ERROR

Short Syntax: LECS.165 LECS: dbase sync call fail: open data path err: *error_code_string* (*error_code*)

Long Syntax: LECS.165 LECS: database sync call failed: open data path err: *error_code_string* (*error_code*)

Description: The LECS was unable to setup an LECS Database Sync VCC due to an inability to open a data path to the caller.

Action: Contact your customer service representative.

LECS.166

Level: C_INFO

Short Syntax: LECS.166 LECS: dbase sync vcc setup to remote LECS

Long Syntax: LECS.166 LECS: dbase sync vcc setup to remote LECS

Description: The LECS Database Synchronization VCC has been successfully established to a remote LECS.

LECS.167

Level: CE_ERROR

Short Syntax: LECS.167 LECS: Dbase Sync call failed: cause *cause_code*

Long Syntax: LECS.167 LECS: Dbase Sync call failed: cause code = *cause_code*

Description: The LECS Database Synchronization VCC failed to be established due to the indicated cause code.

LECS.168

Level: UE_ERROR

Short Syntax: LECS.168 LECS: Dbase Sync call failed: net down

Long Syntax: LECS.168 LECS: Dbase Sync call failed: net down

Description: The LECS Database Synchronization VCC failed to be established because the connection to the network is down.

LECS.169

Level: UE_ERROR

Short Syntax: LECS.169 LECS: Dbase Sync VCC released: cause *cause_code*

Long Syntax: LECS.169 LECS: Dbase Sync VCC released: cause *cause_code*

Description: The LECS Database Synchronization VCC was released with the indicated cause code.

LECS.170

Level: UE_ERROR

Short Syntax: LECS.170 LECS: Dbase Sync VCC released: net down

Long Syntax: LECS.170 LECS: Dbase Sync VCC released: net down

Description: The LECS Database Synchronization VCC was released because the connection to the network is down.

LECS.171

Level: UI_ERROR

Short Syntax: LECS.171 LECS: SRAM error: *reason_string*

Long Syntax: LECS.171 LECS: SRAM error: *reason_string*

Description: The LECS SRAM configuration could not be read correctly during the LECS Database Sync procedure. See the error message for details.

LECS.172

Level: UE_ERROR

Short Syntax: LECS.172 LECS: Dbase Sync VCC rx bad frame: *frame_opcode_string*

Long Syntax: LECS.172 LECS: Dbase Sync VCC received bad frame type: *frame_opcode_string*

Description: The indicated illegal frame type was received on an LECS Database Synchronization VCC.

LECS.173

Level: U_INFO

Short Syntax: LECS.173 LECS: Dbase Sync retry timer expiration num= *num_call_failures*

Long Syntax: LECS.173 LECS: Dbase Sync retry timer expiration num= *num_call_failures*

Description: The LECS Database Synchronization VCC retry timer has expired indicating another Database Sync VCC should be attempted.

LECS.174

Level: C_INFO

Short Syntax: LECS.174 LECS: Config Req sent on Dbase Sync VCC

Long Syntax: LECS.174 LECS: Config Req sent on Dbase Sync VCC

Description: The Configure Request frame was successfully sent on the LECS Database Synchronization VCC with the encoded LECS SRAM data.

LECS.175

Level: UI_ERROR

Short Syntax: LECS.175 LECS: error writing config from Dbase Sync VCC

Long Syntax: LECS.175 LECS: error writing config from Dbase Sync VCC

Description: An error occurred when storing the

LECS SRAM records received on the LECS Database Synchronization VCC.

LECS.176

Level: C_INFO

Short Syntax: LECS.176 LECS: Config Request received on Dbase Sync VCC

Long Syntax: LECS.176 LECS: Config Request received on Dbase Sync VCC

Description: A Configure Request frame was successfully received on the LECS Database Synchronization VCC.

LECS.177

Level: C_INFO

Short Syntax: LECS.177 LECS: Config Response received on Dbase Sync VCC: status= *status*

Long Syntax: LECS.177 LECS: Config Response received on Dbase Sync VCC: status= *status*

Description: A Configure Response frame was successfully received on the LECS Database Synchronization VCC.

LECS.178

Level: UI_ERROR

Short Syntax: LECS.178 LECS: not ready for Config Response on Dbase Sync VCC

Long Syntax: LECS.178 LECS: not ready for Config Response on Dbase Sync VCC

Description: An error occurred while receiving a Config Response on the LECS Database Synchronization VCC. The pointer to the current Dbase Sync control block is invalid (NULL).

Chapter 65. LAN Emulation Server and Broadcast Unknown Server (LES/BUS)

This chapter describes LAN Emulation Server and Broadcast Unknown Server (LES/BUS) messages. For information on message content and how to use the message, refer to the Introduction.

LES.001

Level: UI_ERROR

Short Syntax: LES.001 LES/BUS:' *ELAN_name*':trmntng: *error_string* (*error_code*)

Long Syntax: LES.001 LES/BUS:' *ELAN_name*':terminating: *error_string* (*error_code*)

Description: ELAN is being terminated

LES.002

Level: CE_ERROR

Short Syntax: LES.002 LE:Cell Rate IE:Fwd PCR(CLP=0+1) excds ln rt *fwd_peak_rate*

Long Syntax: LES.002 LE:Cell Rate IE:Forward Peak Cell Rate(CLP=0+1) exceeds line rate *fwd_peak_rate*

Description: Forward Peak Cell Rate for low priority data, exceeds line rate

LES.003

Level: CE_ERROR

Short Syntax: LES.003 LE:Cell Rate IE:Fwd SCR(CLP=0+1) excds max *fwd_sustainable_rate*

Long Syntax: LES.003 LE:Cell Rate IE:Forward Sustainable Cell Rate(CLP=0+1) exceeds maximum *fwd_sustainable_rate*

Description: Forward Sustainable Cell Rate for low priority data exceeds maximum reserved cell rate

LES.004

Level: CE_ERROR

Short Syntax: LES.004 LE:Cell Rate IE:Fwd SCR(CLP=0) excds max *fwd_sustainable_rate*

Long Syntax: LES.004 LE:Cell Rate IE:Forward Sustainable Cell Rate(CLP=0) exceeds maximum *fwd_sustainable_rate*

Description: Forward Sustainable Cell Rate for high priority data exceeds maximum reserved cell rate

LES.005

Level: CE_ERROR

Short Syntax: LES.005 LE:Cell Rate IE:Fwd PCR(CLP=0+1) excds max *fwd_peak_rate*

Long Syntax: LES.005 LE:Cell Rate IE:Forward Peak Cell Rate(CLP=0+1) exceeds maximum *fwd_peak_rate*

Description: Forward Peak Cell Rate for low priority data exceeds maximum reserved cell rate

LES.006

Level: CE_ERROR

Short Syntax: LES.006 LE:Cell Rate IE:Bak SCR(CLP=0+1) excds max *bak_sustainable_rate*

Long Syntax: LES.006 LE:Cell Rate IE:Backward Sustainable Cell Rate(CLP=0+1) exceeds maximum *bak_sustainable_rate*

Description: Backward Sustainable Cell Rate for low priority data exceeds maximum reserved cell rate

LES.007

Level: CE_ERROR

Short Syntax: LES.007 LE:Cell Rate IE:Bak SCR(CLP=0) excds max *bak_sustainable_rate*

Long Syntax: LES.007 LE:Cell Rate IE:Backward Sustainable Cell Rate(CLP=0) exceeds maximum *bak_sustainable_rate*

Description: Backward Sustainable Cell Rate for high priority data exceeds maximum reserved cell rate

LES.008

Level: CE_ERROR

Short Syntax: LES.008 LE:Cell Rate IE:Bak PCR(CLP=0+1) excds max *bak_peak_rate*

Long Syntax: LES.008 LE:Cell Rate IE:Backward Peak Cell Rate(CLP=0+1) exceeds maximum *bak_peak_rate*

Description: Backward Peak Cell Rate for low priority data exceeds maximum reserved cell rate

LES.009

Level: CE_ERROR

Short Syntax: LES.009 LE:Bearer IE:Invld class (x *bearer_class*)

Long Syntax: LES.009 LE:Bearer IE:Invalid class (x *bearer_class*)

Description: Invalid bearer class, bearer class should be class C or class X

LES.010

Level: CE_ERROR

Short Syntax: LES.010 LE:Bearer IE:Invld conn type (x *conn_type*)

Long Syntax: LES.010 LE:Bearer IE:Invalid connection type (x *conn_type*)

Description: Invalid connection type, connection type should be point-to-point

LES.011

Level: CE_ERROR

Short Syntax: LES.011 LE:QOS IE:Invld fwd QOS class (x *fwd_QOS*)

Long Syntax: LES.011 LE:QOS IE:Invalid forward QOS class (x *fwd_QOS*)

Description: Connection is best effort service, and forward Quality Of Service should be QOS class 0

LES.012

Level: CE_ERROR

Short Syntax: LES.012 LE:QOS IE:Invld bak QOS class (x *bak_QOS*)

Long Syntax: LES.012 LE:QOS IE:Invalid backward QOS class (x *bak_QOS*)

Description: Connection is best effort, and backward Quality Of Service should be QOS class 0

LES.013

Level: CE_ERROR

Short Syntax: LES.013 LE:Calling Party addr IE not prsnt

Long Syntax: LES.013 LE:Calling Party address IE not present

Description: Calling Party address IE is not present

LES.014

Level: CE_ERROR

Short Syntax: LES.014 LE:Calling Party Addr IE:Invld ATM addr lngth (*remote_addr_length*)

Long Syntax: LES.014 LE:Calling Party Addr IE:Invalid ATM address length (*remote_addr_length*)

Description: Calling Party Address IE has invalid ATM address length

LES.015

Level: CE_ERROR

Short Syntax: LES.015 LE:Calling Party Addr IE:ATM addr fld scrn

Long Syntax: LES.015 LE:Calling Party Addr IE:ATM address failed screening

Description: ATM address was verified and failed screening

LES.016

Level: CE_ERROR

Short Syntax: LES.016 LE:Calling Party Addr IE:Invld ATM addr

Long Syntax: LES.016 LE:Calling Party Address IE:Invalid ATM address

Description: Format of ATM address is incorrect, only private ATM address format is supported

LES.017

Level: CE_ERROR

Short Syntax: LES.017 LE:AAL IE:Not prsnt, or Invld AAL type (x *AAL_type*)

Long Syntax: LES.017 LE:AAL IE:Not present, or Invalid AAL type (x *AAL_type*)

Description: Invalid AAL type, AAL type should be AAL5

LES.018

Level: CE_ERROR

Short Syntax: LES.018 LE:AAL IE:Invld fwd max SDU sz (*fwd_max_SDU_size*)

Long Syntax: LES.018 LE:AAL IE:Invalid forward maximum SDU size (*fwd_max_SDU_size*)

Description: Forward maximum SDU size is not valid

LES.019

Level: CE_ERROR

Short Syntax: LES.019 LE:AAL IE:Invld bak max SDU sz for P2P call (*bak_max_SDU_size*)

Long Syntax: LES.019 LE:AAL IE:Invalid backward maximum SDU size for Point-to-Point Call (*bak_max_SDU_size*)

Description: For a point-to-point call, the backward maximum SDU size is invalid

LES.020

Level: CE_ERROR

Short Syntax: LES.020 LE:AAL IE:Invld bak max SDU sz for P2MP call (*bak_max_SDU_size*)

Long Syntax: LES.020 LE:AAL IE:Invalid backward maximum SDU size for Point-to-MultiPoint Call (*bak_max_SDU_size*)

Description: For a point-to-multipoint call, the backward maximum SDU size is invalid, should be zero or one

LES.022

Level: CE_ERROR

Short Syntax: LES.022 LE:AAL IE:Invld mode (x *data_transport_mode*)

Long Syntax: LES.022 LE:AAL IE:Invalid mode (x *data_transport_mode*)

Description: For UNI Version 3.0, the data transport mode is invalid, data transport mode should be message mode

LES.023

Level: CE_ERROR

Short Syntax: LES.023 LE:AAL IE:Mode spcfd in UNI 3.1 x *data_transport_mode*

Long Syntax: LES.023 LE:AAL IE:Mode specified in UNI 3.1 x *data_transport_mode*

Description: For UNI 3.1, the data transport mode should not be specified

LES.024

Level: CE_ERROR

Short Syntax: LES.024 LE:AAL IE:Invld SSCS type (x *SSCS_type*)

Long Syntax: LES.024 LE:AAL IE:Invalid SSCS type (x *SSCS_type*)

Description: Invalid SSCS type, SSCS type should be null. This check is no longer performed by LE Services.

LES.025

Level: CE_ERROR

Short Syntax: LES.025 LE:BLLI IE:Invld L2 prtcl (x *l2prot*)

Long Syntax: LES.025 LE:BLLI IE:Invalid Layer 2 protocol (x *l2prot*)

Description: BLLI IE contains an invalid Layer 2 protocol, Layer 2 protocol should be not specified

LES.026

Level: CE_ERROR

Short Syntax: LES.026 LE:BLLI IE:Invld L2 mode (x *l2mode*)

Long Syntax: LES.026 LE:BLLI IE:Invalid Layer 2 mode (x *l2mode*)

Description: Invalid Layer 2 mode, Layer 2 mode should be not specified

LES.027

Level: CE_ERROR

Short Syntax: LES.027 LE:BLLI IE:Invld L2 wndw sz (*l2wndw_size*)

Long Syntax: LES.027 LE:BLLI IE:Invalid Layer 2 window size (*l2wndw_size*)

Description: BLLI IE contains invalid Layer 2 window size, Layer 2 window size should be not specified

LES.028

Level: CE_ERROR

Short Syntax: LES.028 LE:BLLI IE:Invld L2 prtcl info (x *l2info*)

Long Syntax: LES.028 LE:BLLI IE:Invalid Layer 2 protocol info (x *l2info*)

Description: Invalid Layer 2 protocol info, Layer 2 protocol info should be not specified

LES.029

Level: CE_ERROR

Short Syntax: LES.029 LE:BLLI IE:L3 prtcl not spcfd

Long Syntax: LES.029 LE:BLLI IE:Layer 3 protocol not specified

Description: Layer 3 protocol not specified, should be ISO/IEC TR 9577 (x0B)

LES.030

Level: CE_ERROR

Short Syntax: LES.030 LE:BLLI IE:Invld L3 prtcl (x *l3prtcl*)

Long Syntax: LES.030 LE:BLLI IE:Invalid Layer 3 protocol (x *l3prtcl*)

Description: Invalid Layer 3 protocol,should be ISO/IEC TR9577 (x0B)

LES.031

Level: CE_ERROR

Short Syntax: LES.031 LE:BLLI IE:Invld L3 mode (x *l3mode*)

Long Syntax: LES.031 LE:BLLI IE:Invalid Layer 3 mode (x *l3mode*)

Description: Invalid Layer 3 mode, Layer 3 mode should be not specified

---

LES.032

Level: CE_ERROR

Short Syntax: LES.032 LE:BLLI IE:Invld L3 dflt pkt sz (x *l3dflt_pkt_sz*)

Long Syntax: LES.032 LE:BLLI IE:Invalid Layer 3 default packet size (x *l3dflt_pkt_sz*)

Description: Invalid Layer 3 default packet size, Layer 3 packet size should be not specified

---

LES.033

Level: CE_ERROR

Short Syntax: LES.033 LE:BLLI IE:Invld L3 pkt wndw sz x *l3pkt_wndw_sz*

Long Syntax: LES.033 LE:BLLI IE:Invalid Layer 3 packet window size x *l3pkt_wndw_sz*

Description: Invalid Layer 3 packet window size, Layer 3 packet window size should be not specified

---

LES.034

Level: CE_ERROR

Short Syntax: LES.034 LE:BLLI IE:Invld L3 prtcl info (x *l3info*)

Long Syntax: LES.034 LE:BLLI IE:Invalid Layer 3 protocol info (x *l3info*)

Description: Invalid Layer 3 protocol info, Layer 3 protocol info should be not specified

---

LES.035

Level: CE_ERROR

Short Syntax: LES.035 LE:BLLI IE:L3 IPI not spcfd

Long Syntax: LES.035 LE:BLLI IE:Layer 3 Initial Protocol Identifier not specified

Description: Layer 3 Initial Protocol Identifier not specified, Layer 3 IPI should be x80

LES.036

Level: CE_ERROR

Short Syntax: LES.036 LE:BLLI IE:Invld L3 IPI (x *IPI*)

Long Syntax: LES.036 LE:BLLI IE:Invalid Layer 3 Initial Protocol Identifier (x *IPI*)

Description: Invalid Layer 3 Initial Protocol Identifier, Layer 3 IPI should be x80

LES.037

Level: CE_ERROR

Short Syntax: LES.037 LE:BLLI IE:SNAP OUI not spcfd

Long Syntax: LES.037 LE:BLLI IE:SNAP OUI not specified

Description: BLLI IE, SNAP OUI not specified

LES.038

Level: CE_ERROR

Short Syntax: LES.038 LE:BLLI IE:invld SNAP OUI x *SNAP_OUI*

Long Syntax: LES.038 LE:BLLI IE:Invalid SNAP OUI x *SNAP_OUI*

Description: Invalid SNAP OUI, SNAP OUI should be x00 xA0 x3E

LES.039

Level: CE_ERROR

Short Syntax: LES.039 LE:BLLI IE:SNAP PID not spcfd

Long Syntax: LES.039 LE:BLLI IE:SNAP PID not specified

Description: BLLI IE, SNAP PID not specified

LES.040

Level: CE_ERROR

Short Syntax: LES.040 LE:BLLI IE:Invld SNAP PID (x *SNAP_PID*)

Long Syntax: LES.040 LE:BLLI IE:Invalid SNAP PID (x *SNAP_PID*)

Description: BLLI IE, invalid SNAP PID

LES.041

Level: CE_ERROR

Short Syntax: LES.041 LES/BUS:' *ELAN_name*':crt fld:dplct ELAN name

Long Syntax: LES.041 LES/BUS:' *ELAN_name*':create failed: duplicate ELAN name

Description: LES/BUS cannot be created, because a LES/BUS already exists with the given ELAN name

---

LES.042

Level: UI_ERROR

Short Syntax: LES.042 LES/BUS:' *ELAN_name*':crt fld:mem alloc err

Long Syntax: LES.042 LES/BUS:' *ELAN_name*':create failed memory allocation error

Description: When trying to create a LES/BUS, a memory allocation error occurred

Action: Contact your customer service representative

LES.043

Level: UI_ERROR

Short Syntax: LES.043 LES/BUS:' *ELAN_name*':crt fld:dtbs err

Long Syntax: LES.043 LES/BUS:' *ELAN_name*':create failed:database error

Description: Unable to add this LES/BUS to the ELAN database

Action: Contact your customer service representative

LES.044

Level: U_INFO

Short Syntax: LES.044 LES/BUS:' *ELAN_name*':STARTING

Long Syntax: LES.044 LES/BUS:' *ELAN_name*':STARTING

Description: The LES/BUS was started

LES.045

Level: U_INFO

Short Syntax: LES.045 LES/BUS:' *ELAN_name*':releasing *redun_type*Redundancy VCC

Long Syntax: LES.045 LES/BUS:' *ELAN_name*':releasing *redun_type*Redundancy VCC

Description: The Redundancy VCC was released

LES.046

Level: U_INFO

Short Syntax: LES.046 LES/BUS:' *ELAN_name*':DELETED

Long Syntax: LES.046 LES/BUS:' *ELAN_name*':DELETED

Description: The LES/BUS was deleted

LES.047

Level: CE_ERROR

Short Syntax: LES.047 LES/BUS:' *ELAN_name*':rfsd *redun_type*Rdndncy Call Calling ATM addr = x *calling_address*

Long Syntax: LES.047 LES/BUS:' *ELAN_name*':refused *redun_type*Redundancy Call, Calling ATM address = x *calling_address*

Description: Redundancy call was refused

LES.048

Level: U_INFO

Short Syntax: LES.048 LES/BUS:' *ELAN_name*':RESTARTING

Long Syntax: LES.048 LES/BUS:' *ELAN_name*':RESTARTING

Description: LES/BUS was restarted

LES.049

Level: UI_ERROR

Short Syntax: LES.049 LES/BUS:' *ELAN_name*':rfsd *redun_type*Rdndncy call:ack fld:no mem Calling ATM addr = x *calling_address*

Long Syntax: LES.049 LES/BUS:' *ELAN_name*':refused *redun_type*Redundancy Call:ack failed:no memory, Calling ATM address = x *calling_address*

Description: Redundancy call was refused due to insufficient resources

Action: Contact your customer service representative

LES.050

Level: U_INFO

Short Syntax: LES.050 LES/BUS:' *ELAN_name*':STOPPED

Long Syntax: LES.050 LES/BUS:' *ELAN_name*':STOPPED

Description: The LES/BUS was stopped

LES.051

Level: UI_ERROR

Short Syntax: LES.051 LES/BUS:' *ELAN_name*':=>DOWN:err acking *redun_type*Rdndncy call: *error_string* (*error_code*)

Long Syntax: LES.051 LES/BUS:' *ELAN_name*':=>DOWN:error acking *redun_type*Redundancy call: *error_string* (*error_code*)

Description: An error occured when accepting Redundancy call, ELAN will be terminated

---

LES.052

Level: U_INFO

Short Syntax: LES.052 LES/BUS:' *ELAN_name*': *redun_type*Redundancy VCC estblshd Calling Atm addr = x *calling_address*

Long Syntax: LES.052 LES/BUS:' *ELAN_name*': *redun_type*Redundancy VCC established, Calling ATM address = x *calling_address*

Description: Redundancy VCC was established

LES.053

Level: CE_ERROR

Short Syntax: LES.053 LES/BUS:' *ELAN_name*':rfsd Ctrl Dir call to redundant LES Calling ATM addr = x *calling_address*

Long Syntax: LES.053 LES/BUS:' *ELAN_name*':refused Control Direct call to redundant LES, Calling ATM address = x *calling_address*

Description: Control Direct call rejected because LES is redundant, yielding to the partner LES/BUS.

LES.054

Level: U_INFO

Short Syntax: LES.054 LES/BUS:' *ELAN_name*': *redun_type*Redundancy VCC estblshd Called ATM addr = x *called_address*

Long Syntax: LES.054 LES/BUS:' *ELAN_name*': *redun_type*Redundancy VCC established, Called ATM address = x *called_address*

Description: Redundancy VCC established

LES.055

Level: C_INFO

Short Syntax: LES.055 LES/BUS:' *ELAN_name*': *VCC_type* rlsd:nrml

Long Syntax: LES.055 LES/BUS:' *ELAN_name*': *VCC_type* released:normal

Description: A VCC was released for normal reasons

LES.056

Level: CE_ERROR

Short Syntax: LES.056 LES/BUS:' *ELAN_name*': *redun_type*Rdndncy call fld:cause *cause_code* Called ATM addr = x *called_address*

Long Syntax: LES.056 LES/BUS:' *ELAN_name*': *redun_type*Redundancy call failed:cause *cause_code*, Called ATM address = x *called_address*

Description: Redundancy call failed

LES.057

Level: UI_ERROR

Short Syntax: LES.057 LES/BUS:' *ELAN_name*':=>DOWN:ATM user reg fld: *error_string* (*error_code*)

Long Syntax: LES.057 LES/BUS:' *ELAN_name*':=>DOWN:ATM user registration failed: *error_string* (*error_code*)

Description: ATM user registration failed

Action: Contact your customer service representative

LES.058

Level: U_INFO

Short Syntax: LES.058 LES/BUS:' *ELAN_name*':waiting for ATM Net Up

Long Syntax: LES.058 LES/BUS;' *ELAN_name*':waiting for ATM Net Up

Description: ATM interface is down, waiting for a Net Up

LES.059

Level: U_INFO

Short Syntax: LES.059 LES/BUS:' *ELAN_name*':waiting for ATM addr actvn

Long Syntax: LES.059 LES/BUS:' *ELAN_name*':waiting for ATM address activation

Description: ATM address activation has not yet completed

LES.060

Level: UI_ERROR

Short Syntax: LES.060 LES/BUS:'

ELAN_name':=>DOWN:ATM addr actvn fld: *error_string* (*error_code*)

Long Syntax: LES.060 LES/BUS:' *ELAN_name*':=>DOWN:ATM address activation failed: *error_string* (*error_code*)

Description: ATM address activation failed

Action: Contact your customer service representative

LES.061

Level: UI_ERROR

Short Syntax: LES.061 LES/BUS:' *ELAN_name*':=>DOWN:err reading ATM addr: *error_string* (*error_code*)

Long Syntax: LES.061 LES/BUS:' *ELAN_name*':=>DOWN:err reading ATM address: *error_string* (*error_code*)

Description: Error reading ATM address

Action: Contact your customer service representative

LES.062

Level: U_INFO

Short Syntax: LES.062 LES/BUS:' *ELAN_name*':waiting for UNI Vrsn rpt

Long Syntax: LES.062 LES/BUS:' *ELAN_name*':waiting for UNI Version report

Description: UNI Version Report has not yet completed

LES.063

Level: UI_ERROR

Short Syntax: LES.063 LES/BUS:' *ELAN_name*':=>DOWN:err reading UNI Vrsn: *error_string* (*error_code*)

Long Syntax: LES.063 LES/BUS:' *ELAN_name*':=>DOWN:error reading UNI Version: *error_string* (*error_code*)

Description: Error reading UNI version

LES.064

Level: UI_ERROR

Short Syntax: LES.064 LES/BUS:' *ELAN_name*':=>DOWN:err opening ATM Adptr Frame SAP: *error_string* (*error_code*)

Long Syntax: LES.064 LES/BUS:' *ELAN_name*':=>DOWN:error opening ATM Adapter Frame SAP: *error_string* (*error_code*)

Description: Error opening ATM Adapter Frame SAP

Action: Contact your customer service representative

LES.065

Level: UI_ERROR

Short Syntax: LES.065 LES/BUS:' *ELAN_name*':=>DOWN:err opening Call SAP: *error_string* (*error_code*)

Long Syntax: LES.065 LES/BUS:' *ELAN_name*':=>DOWN:error opening Call SAP: *error_string* (*error_code*)

Description: Error opening Call SAP

Action: Contact your customer service representative

LES.066

Level: UI_ERROR

Short Syntax: LES.066 LES/BUS:' *ELAN_name*':=>DOWN:err opening Ctrl Dist Grp: *error_string* (*error_code*)

Long Syntax: LES.066 LES/BUS:' *ELAN_name*':=>DOWN:error opening Control Distribute Group: *error_string* (*error_code*)

Description: Error opening Control Distribute Group

Action: Contact your customer service representative

LES.067

Level: UI_ERROR

Short Syntax: LES.067 LES/BUS:' *ELAN_name*':=>DOWN:err opening Mcast Fwd Grp: *error_string* (*error_code*)

Long Syntax: LES.067 LES/BUS:' *ELAN_name*':=>DOWN:error opening Multicast Forward Group: *error_string* (*error_code*)

Description: Error opening Multicast Forward Group

Action: Contact your customer service representative

LES.068

Level: UI_ERROR

Short Syntax: LES.068 LES/BUS:' *ELAN_name*':BCM init fld

Long Syntax: LES.068 LES/BUS:' *ELAN_name*':BCM initialization failed

Description: BroadCast Manager initialization failed

LES.069

Level: UI_ERROR

Short Syntax: LES.069 LES/BUS:' *ELAN_name*':err starting IP BCM

Long Syntax: LES.069 LES/BUS:' *ELAN_name*':error starting IP BCM

Description: An error occurred while attempting to start IP BroadCast Manager

---

LES.070

Level: UI_ERROR

Short Syntax: LES.070 LES/BUS:' *ELAN_name*':err starting IPX BCM

Long Syntax: LES.070 LES/BUS:' *ELAN_name*':error starting IPX BCM

Description: An error occurred while attempting to start IPX BroadCast Manager

---

LES.071

Level: UI_ERROR

Short Syntax: LES.071 LES/BUS:' *ELAN_name*':err starting NetBIOS BCM

Long Syntax: LES.071 LES/BUS:' *ELAN_name*':error starting NetBIOS BCM

Description: An error occurred while trying to start NetBIOS BroadCast Manager

---

LES.072

Level: UE_ERROR

Short Syntax: LES.072 LES/BUS:' *ELAN_name*':ATM Net DOWN

Long Syntax: LES.072 LES/BUS:' *ELAN_name*':ATM Net DOWN

Description: ATM interface is in an inoperable state

LES.073

Level: U_INFO

Short Syntax: LES.073 LES/BUS:' *ELAN_name*':ATM Net UP

Long Syntax: LES.073 LES/BUS:' *ELAN_name*':ATM Net UP

Description: ATM interface is in an operable state

LES.074

Level: U_INFO

Short Syntax: LES.074 LES/BUS:' *ELAN_name*':ATM addr actvted

Long Syntax: LES.074 LES/BUS:' *ELAN_name*':ATM address activated

Description: ATM address was activated successfully

LES.075

Level: UE_ERROR

Short Syntax: LES.075 LES/BUS:' *ELAN_name*':ATM addr actvtn tmd out: retrying

Long Syntax: LES.075 LES/BUS:' *ELAN_name*':ATM address activation timed out:retrying

Description: ATM address activation request timed out, activation will be retried

---

LES.076

Level: UE_ERROR

Short Syntax: LES.076 LES/BUS:' *ELAN_name*':ATM addr rjctd by switch

Long Syntax: LES.076 LES/BUS:' *ELAN_name*':ATM address rejected by switch

Description: ATM address was rejected by switch. Another attempt will be made to activate the ATM address.

LES.077

Level: UE_ERROR

Short Syntax: LES.077 LES/BUS:' *ELAN_name*':ATM Addr deactvtd: reactvtng

Long Syntax: LES.077 LES/BUS:' *ELAN_name*':ATM address deactivated: reactivating

Description: ATM address has been deactivated by switch, address will be reactivated

---

LES.078

Level: U_INFO

Short Syntax: LES.078 LES/BUS:' *ELAN_name*':UNI Vrsn rprtd

Long Syntax: LES.078 LES/BUS:' *ELAN_name*':UNI Version reported

Description: The UNI version was reported

LES.079

Level: UI_ERROR

Short Syntax: LES.079 Unexpected LECS addr lst rprtd

Long Syntax: LES.079 Unexpected LECS address list reported

Description: An unexpected LECS ATM address list was reported

---

LES.080

Level: CE_ERROR

Short Syntax: LES.080 LES/BUS:' *ELAN_name*':rfsd Ctrl Dir call

Long Syntax: LES.080 LES/BUS:' *ELAN_name*':refused Control Direct call

Description: Validation of request for Control direct VCC failed

LES.081

Level: CE_ERROR

Short Syntax: LES.081 LES/BUS:' *ELAN_name*':rfsd Ctrl Dir Call, Calling ATM addr = x *calling_address*

Long Syntax: LES.081 LES/BUS:' *ELAN_name*':refused Control Direct Call, Calling ATM address = x *calling_address*

Description: Validation of request for Control Direct VCC failed

LES.082

Level: UI_ERROR

Short Syntax: LES.082 LES/BUS:' *ELAN_name*':rfsd Ctrl Dir Call:mem alloc err, Calling ATM addr = x *calling_address*

Long Syntax: LES.082 LES/BUS:' *ELAN_name*':refused Control Direct Call:memory allocation error, Calling ATM address = x *calling_address*

Description: Request for Control Direct VCC failed, unable to allocate memory

Action: Contact your customer service representative

LES.083

Level: UI_ERROR

Short Syntax: LES.083 LES/BUS:' *ELAN_name*':rfsd Ctrl Dir Call:dt pth opn err:no mem, Calling ATM addr = x *calling_address*

Long Syntax: LES.083 LES/BUS:' *ELAN_name*':refused Control Direct Call:data path open error:no memory, Calling ATM address = x *calling_address*

Description: Insufficient resources to open data path for Control Direct VCC

Action: Contact your customer service representative

LES.084

Level: UI_ERROR

Short Syntax: LES.084 LES/BUS:' *ELAN_name*':=>DOWN:Ctrl Dir dt pth opn err: *error_string* (*error_code*)

Long Syntax: LES.084 LES/BUS:' *ELAN_name*':=>DOWN:Control Direct data path open error: *error_string* (*error_code*)

Description: An error occurred when trying to open data path for Control Direct VCC, the ELAN will be terminated

LES.085

Level: UI_ERROR

Short Syntax: LES.085 LES/BUS:' *ELAN_name*':rfsd Ctrl Dir Call:ack fld:no mem, Calling ATM addr = x *calling_address*

Long Syntax: LES.085 LES/BUS:' *ELAN_name*':refused Control Direct Call:ack failed:no memory, Calling ATM address = x *calling_address*

Description: Unable to accept Control Direct Call due to insufficient resources

Action: Contact your customer service representative

LES.086

Level: UI_ERROR

Short Syntax: LES.086 LES/BUS:' *ELAN_name*':=>DOWN:err acking Ctrl Dir call: *error_string* (*error_code*)

Long Syntax: LES.086 LES/BUS:' *ELAN_name*':=>DOWN:error acking Control Direct call: *error_string* (*error_code*)

Description: An error occurred while accepting Control Direct Call, ELAN will be terminated

LES.087

Level: C_INFO

Short Syntax: LES.087 LES/BUS:' *ELAN_name*':Ctrl Dir estblshd, Calling ATM addr = x *calling_address*

Long Syntax: LES.087 LES/BUS:' *ELAN_name*':=>Control Direct established, Calling ATM address = x *calling_address*

Description: Control Direct VCC was established

LES.088

Level: CE_ERROR

Short Syntax: LES.088 LES/BUS:' *ELAN_name*':rfsd Mcast Send call

Long Syntax: LES.088 LES/BUS:' *ELAN_name*':refused Multicast Send call

Description: Validation of request for Multicast Send VCC failed

LES.089

Level: CE_ERROR

Short Syntax: LES.089 LES/BUS:' *ELAN_name*':rfsd Mcast Send call, Calling ATM addr = x *calling_address*

Long Syntax: LES.089 LES/BUS:' *ELAN_name*':refused Multicast Send call, Calling ATM addr = x *calling_address*

Description: Validation of request for Multicast Send VCC failed

---

LES.090

Level: CE_ERROR

Short Syntax: LES.090 LES/BUS:' *ELAN_name*':rfsd Mcast Send call:unkwn ATM addr, calling ATM addr = x *calling_address*

Long Syntax: LES.090 LES/BUS:' *ELAN_name*':refused Multicast Send Call:unknown ATM address,calling ATM address = x *calling_address*

Description: Multicast Send Call refused, ATM address is unknown

---

LES.091

Level: CE_ERROR

Short Syntax: LES.091 LES/BUS:' *ELAN_name*':rfsd Mcast Send call:JOIN incmplt, LEC ATM addr = x *LEC_address*

Long Syntax: LES.091 LES/BUS:' *ELAN_name*':refused Multicast Send call:join incomplete, LEC ATM address = x *LEC_address*

Description: Multicast Send Call refused, JOIN phase has not completed

---

LES.092

Level: CE_ERROR

Short Syntax: LES.092 LES/BUS:' *ELAN_name*':rfsd Mcast Send call:VCC alrdy actv, LEC ATM addr = x *LEC_address*

Long Syntax: LES.092 LES/BUS:' *ELAN_name*':refused Multicast Send call:VCC already active, LEC ATM address = x *LEC_address*

Description: LEC already has a connection to the BUS

LES.093

Level: UI_ERROR

Short Syntax: LES.093 LES/BUS:' *ELAN_name*':rfsd Mcast Send call:dt pth opn err:no mem, LEC ATM addr = x *LEC_address*

Long Syntax: LES.093 LES/BUS:' *ELAN_name*':refused Multicast Send call:data path open error:no memory, LEC ATM addres = x *LEC_address*

Description: Insufficient resources to open data path for Multicast Send VCC

Action: Contact your customer service representative

LES.094

Level: UI_ERROR

Short Syntax: LES.094 LES/BUS:' *ELAN_name*':=>DOWN:Mcast Send dt pth opn err: *error_string* (*error_code*)

Long Syntax: LES.094 LES/BUS:' *ELAN_name*':=>DOWN:Multicast Send data path open error: *error_string* (*error_code*)

Description: An error occurred when trying to open data path for Multicast Send VCC, ELAN will be terminated

---

LES.095

Level: UI_ERROR

Short Syntax: LES.095 LES/BUS:' *ELAN_name*':rfsd Mcast Send call:ack fld:no mem, LEC ATM addr = x *LEC_address*

Long Syntax: LES.095 LES/BUS:' *ELAN_name*':refused Multicast Send call:ack failed:no memory, LEC ATM address = x *LEC_address*

Description: Unable to accept Multicast Send Call, due to insufficient resources

Action: Contact your customer service representative

LES.096

Level: UI_ERROR

Short Syntax: LES.096 LES/BUS:' *ELAN_name*':=>DOWN:err ackng Mcast Send call: *error_string* (*error_code*)

Long Syntax: LES.096 LES/BUS:' *ELAN_name*':=>DOWN:error acknowledging Multicast Send call: *error_string* (*error_code*)

Description: An error occurred while accepting Multicast Send Call, ELAN will be terminated

---

LES.097

Level: C_INFO

Short Syntax: LES.097 LES/BUS:' *ELAN_name*':Mcast Send estblshd, LEC ATM addr = x *LEC_address*

Long Syntax: LES.097 LES/BUS:' *ELAN_name*':Multicast Send established, LEC ATM address = x *LEC_address*

Description: Multicast Send VCC was established

LES.098

Level: CE_ERROR

Short Syntax: LES.098 LES/BUS:' *ELAN_name*':trmntng LEC:ngttd *VCC_type* parms, LEC ATM addr = x *LEC_address*

Long Syntax: LES.098 LES/BUS:' *ELAN_name*':terminating LEC:negotiated *VCC_type* parms, LEC ATM addr = x *LEC_address*

Description: AAL and BLLI parameters are not negotiable

LES.099

Level: UI_ERROR

Short Syntax: LES.099 LES/BUS:' *ELAN_name*':=>DOWN: *VCC_type* dt pth opn err: *error_string* (*error_code*)

Long Syntax: LES.099 LES/BUS:' *ELAN_name*':=>DOWN: *VCC_type* data path open error: *error_string* (*error_code*)

Description: An error occurred when trying to open data path for VCC, ELAN will be terminated

---

LES.100

Level: UI_ERROR

Short Syntax: LES.100 LES/BUS:' *ELAN_name*':trmntng LEC: *VCC_type* dt pth opn err:no mem, LEC ATM addr = x *LEC_address*

Long Syntax: LES.100 LES/BUS:' *ELAN_name*':terminating LEC: *VCC_type* data path open error:no memory, LEC ATM address = x *LEC_address*

Description: Insufficient resources to open data path for VCC

Action: Contact your customer service representative

LES.101

Level: C_INFO

Short Syntax: LES.101 LES/BUS:' *ELAN_name*': *VCC_type* estblshd, LEC ATM addr = x *LEC_address*

Long Syntax: LES.101 LES/BUS:' *ELAN_name*': *VCC_type* established, LEC ATM address = x *LEC_address*

Description: VCC of the given type was established

LES.102

Level: UI_ERROR

Short Syntax: LES.102 LES/BUS:' *ELAN_name*':=>DOWN:err adding to Ctrl Dist Grp: *error_string* (*error_code*)

Long Syntax: LES.102 LES/BUS:' *ELAN_name*':=>DOWN:error adding to Control Distribute Group: *error_string* (*error_code*)

Description: An error occurred when trying to add VCC to Control Distribute Group

---

LES.103

Level: UI_ERROR

Short Syntax: LES.103 LES/BUS:' *ELAN_name*':=>DOWN:err adding to Mcast Fwd Grp: *error_string* (*error_code*)

Long Syntax: LES.103 LES/BUS:' *ELAN_name*':=>DOWN:error adding to Multicast Forward Group: *error_string* (*error_code*)

Description: An error occurred while trying to add VCC to Multicast Forward Group, ELAN will be terminated

---

LES.104

Level: C_INFO

Short Syntax: LES.104 LES/BUS:' *ELAN_name*': *VCC_type* leaf estblshd, LEC ATM addr = x *LEC_address*

Long Syntax: LES.104 LES/BUS:' *ELAN_name*': *VCC_type* leaf established, LEC ATM address = x *LEC_address*

Description: For the given VCC type, a party was added to a point-to-multipoint call

---

LES.105

Level: C_INFO

Short Syntax: LES.105 LES/BUS:' *ELAN_name*':trmntng LEC:Ctrl Dir rlsd:nrml, LEC ATM addr = x *LEC_address*

Long Syntax: LES.105 LES/BUS:' *ELAN_name*':terminating LEC:Control Direct released:normal, LEC ATM address = x *LEC_address*

Description: A Control Direct Call was released for normal reasons, the LEC's ELAN membership will be terminated

---

LES.106

Level: CE_ERROR

Short Syntax: LES.106 LES/BUS:' *ELAN_name*':trmntng LEC:Ctrl Dir rlsd:cause *cause_code*, LEC ATM addr = x *LEC_address*

Long Syntax: LES.106 LES/BUS:' *ELAN_name*':terminating LEC:Control Direct

released:cause *cause_code*, LEC ATM address = x *LEC_address*

Description: A Control Direct Call was released due to the given cause, the LEC's ELAN membership will be terminated

---

LES.107

Level: UE_ERROR

Short Syntax: LES.107 LES/BUS:' *ELAN_name*':trmntng LEC:Ctrl Dir rlsd:nt dwn, LEC ATM addr = x *LEC_address*

Long Syntax: LES.107 LES/BUS:' *ELAN_name*':terminating LEC:Control Direct released:net down, LEC ATM address = x *LEC_address*

Description: A Control Direct Call was released, because the connection to the network was down. The LEC's ELAN membership will be terminated

---

LES.108

Level: C_INFO

Short Syntax: LES.108 LES/BUS:' *ELAN_name*': *VCC_type* call fld:retrying wth Bearer Class C, LEC ATM addr = x *LEC_address*

Long Syntax: LES.108 LES/BUS:' *ELAN_name*': *VCC_type* call failed:retrying with Bearer Class C, LEC ATM address = x *LEC_address*

Description: A call failed of the given type, the call will be retried with Bearer Class C

LES.109

Level: CE_ERROR

Short Syntax: LES.109 LES/BUS:' *ELAN_name*':trmntng LEC: *VCC_type* call fld:cause *cause_code*, LEC ATM addr = x *LEC_address*

Long Syntax: LES.109 LES/BUS:' *ELAN_name*':terminating LEC: *VCC_type* call failed:cause *cause_code*, LEC ATM address = x *LEC_address*

Description: A called failed due to the given cause, the LEC's ELAN membership will be terminated

---

LES.110

Level: UE_ERROR

Short Syntax: LES.110 LES/BUS:' *ELAN_name*':trmntng LEC: *VCC_type* call fld:net dwn, LEC ATM addr = x *LEC_address*

Long Syntax: LES.110 LES/BUS:' *ELAN_name*':terminating LEC: *VCC_type* call failed:net down, LEC ATM address = x *LEC_address*

Description: A call failed because the connection to the network was down. The LEC's ELAN membership will be terminated

---

LES.111

Level: UE_ERROR

Short Syntax: LES.111 LES/BUS:' *ELAN_name*': *VCC_type* rlsd:cause *cause_code*

Long Syntax: LES.111 LES/BUS:' *ELAN_name*': *VCC_type* released:cause *cause_code*

Description: A VCC was released, due to the given cause

LES.112

Level: UE_ERROR

Short Syntax: LES.112 LES/BUS:' *ELAN_name*': *VCC_type* rlsd:net dwn

Long Syntax: LES.112 LES/BUS:' *ELAN_name*': *VCC_type* released:net down

Description: A VCC was released, because the connection to the network was down

---

LES.113

Level: C_INFO

Short Syntax: LES.113 LES/BUS:' *ELAN_name*':Mcast Send rlsd:nrml, LEC ATM addr = x *LEC_address*

Long Syntax: LES.113 LES/BUS:' *ELAN_name*':Multicast Send released:normal, LEC ATM address = x *LEC_address*

Description: A Multicast Send Call was released for normal reasons

---

LES.114

Level: CE_ERROR

Short Syntax: LES.114 LES/BUS:' *ELAN_name*':Mcast Send rlsd:cause *cause_code*, LEC ATM addr = x *LEC_address*

Long Syntax: LES.114 LES/BUS:' *ELAN_name*':Multicast Send released:cause *cause_code*, LEC ATM address = x *LEC_address*

Description: A Multicast Send Call was released, due to the given cause

---

LES.115

Level: UE_ERROR

Short Syntax: LES.115 LES/BUS:' *ELAN_name*':Mcast Send rlsd:net dwn, LEC ATM addr = x *LEC_address*

Long Syntax: LES.115 LES/BUS:' *ELAN_name*':Multicast Send released:net down, LEC

ATM address = x *LEC_address*

Description: A Multicast Send Call was released, becuase the connection to the network is currently down

---

LES.116

Level: CE_ERROR

Short Syntax: LES.116 LES/BUS:' *ELAN_name*':trmntng LEC:err adding *VCC_type* leaf:cause *cause_code*, LEC ATM addr = x *LEC_address*

Long Syntax: LES.116 LES/BUS:' *ELAN_name*':terminating LEC:error adding *VCC_type* leaf:cause *cause_code*, LEC ATM address = x *LEC_address*

Description: An error occurred when adding a leaf, the LEC's ELAN membership will be terminated

---

LES.118

Level: C_INFO

Short Syntax: LES.118 LES/BUS:' *ELAN_name*':trmntng LEC: *VCC_type* leaf rlsd:nrml, LEC ATM addr = x *LEC_address*

Long Syntax: LES.118 LES/BUS:' *ELAN_name*':terminating LEC: *VCC_type* leaf released:normal, LEC ATM address = x *LEC_address*

Description: A leaf was released for normal reasons, the LEC's ELAN membership will be terminated

---

LES.119

Level: CE_ERROR

Short Syntax: LES.119 LES/BUS:' *ELAN_name*':trmntng LEC: *VCC_type* leaf rlsd:cause *cause_code*, LEC ATM addr = x *LEC_address*

Long Syntax: LES.119 LES/BUS:' *ELAN_name*':terminating LEC: *VCC_type* leaf released:cause *cause_code*, LEC ATM address = x *LEC_address*

Description: A leaf was released due to the given cause, the LEC's ELAN membership will be terminated

---

LES.120

Level: UE_ERROR

Short Syntax: LES.120 LES/BUS:' *ELAN_name*':trmntng LEC: *VCC_type* leaf rlsd:net dwn, LEC ATM addr = x *LEC_address*

Long Syntax: LES.120 LES/BUS:' *ELAN_name*':terminating LEC: *VCC_type* leaf released:net down, LEC ATM address = x *LEC_address*

Description: A leaf was released because the connection to the network was down. The LEC's ELAN membership will be terminated

---

LES.121

Level: C_INFO

Short Syntax: LES.121 LES/BUS:' *ELAN_name*': *VCC_type* leaf rlsd:normal, LEC ATM addr = x *LEC_address*

Long Syntax: LES.121 LES/BUS:' *ELAN_name*': *VCC_type* leaf released:normal, LEC ATM address = x *LEC_address*

Description: A leaf was released for normal reasons

LES.122

Level: CE_ERROR

Short Syntax: LES.122 LES/BUS:' *ELAN_name*': *VCC_type* leaf rlsd:cause *cause_code*, LEC ATM addr = x *LEC_address*

Long Syntax: LES.122 LES/BUS:' *ELAN_name*': *VCC_type* leaf released:cause *cause_code*, LEC ATM address = x *LEC_address*

Description: A leaf was released due to the given cause

LES.123

Level: UE_ERROR

Short Syntax: LES.123 LES/BUS:' *ELAN_name*': *VCC_type* leaf rlsd:net dwn, LEC ATM addr = x *LEC_address*

Long Syntax: LES.123 LES/BUS:' *ELAN_name*': *VCC_type* leaf released:net down, LEC ATM address = x *LEC_address*

Description: A leaf was released because the connection to the network was down

---

LES.124

Level: C_INFO

Short Syntax: LES.124 LES/BUS:' *ELAN_name*':dscrd OAM frm, PTI (x *pti*)

Long Syntax: LES.124 LES/BUS:' *ELAN_name*':discarded OAM frame, PTI (x *pti*)

Description: An OAM frame was discarded

LES.125

Level: CE_ERROR

Short Syntax: LES.125 LES/BUS:' *ELAN_name*':dscrd cntrl frm:invld mrkr (x *marker*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.125 LES/BUS:' *ELAN_name*':discarded control frame:invalid Marker (x *marker*), LEC ATM addr = x *LEC_address*

Description: A control frame was discarded, because the Marker was invalid. The Marker should be xFF00

LES.126

Level: CE_ERROR

Short Syntax: LES.126 LES/BUS:' *ELAN_name*':dscrd cntrl frm:invld prtcl (x *protocol*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.126 LES/BUS:' *ELAN_name*':discarded control frame:invalid prtcl (x *protocol*), LEC ATM addr = x *LEC_address*

Description: A control frame was discarded, because the protocol was invalid. The protocol should be x01

LES.127

Level: CE_ERROR

Short Syntax: LES.127 LES/BUS:' *ELAN_name*':dscrd cntrl frm:invld Vrsn (x *version*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.127 LES/BUS:' *ELAN_name*':discarded control frame:invalid Version (x *version*), LEC ATM addr = x *LEC_address*

Description: A control frame was discarded, because the Version is invalid. The version should be x01

LES.128

Level: CE_ERROR

Short Syntax: LES.128 LES/BUS:' *ELAN_name*':dscrd ARP RSP:src JOIN incmplt, Src LEC ATM addr = x *source_LEC_address*

Long Syntax: LES.128 LES/BUS:' *ELAN_name*':discarded ARP Response:source JOIN incomplete, Source LEC ATM address = x *source_LEC_address*

Description: An ARP Response was discarded, because the JOIN phase has not completed for the source LEC

LES.129

Level: CE_ERROR

Short Syntax: LES.129 LES/BUS:' *ELAN_name*':dscrd ARP RSP:unkwn LECID (x *LECID*), Src LEC ATM addr = x *source_LEC_address*

Long Syntax: LES.129 LES/BUS:' *ELAN_name*':discarded ARP Response:unknown LECID (x *LECID*), Source LEC ATM address = x *source_LEC_address*

Description: An ARP Response was discarded, because the LECID is unknown

---

LES.130

Level: CE_ERROR

Short Syntax: LES.130 LES/BUS:' *ELAN_name*':dscrd ARP RSP:trgt JOIN incmplt, Src LEC ATM addr = x *source_LEC_address*, Trgt LEC ATM addr = x *target_LEC_address*

Long Syntax: LES.130 LES/BUS:' *ELAN_name*':discarded ARP Response:target JOIN incomplete, Source LEC ATM address = x *source_LEC_address*, Target ATM address = x *target_LEC_address*

Description: An ARP Response was discarded, because the JOIN phase has not completed for the target LEC

LES.131

Level: CE_ERROR

Short Syntax: LES.131 LES/BUS:' *ELAN_name*':dscrd FLUSH RSP:src JOIN incmplt, Src LEC ATM addr = x *source_LEC_address*

Long Syntax: LES.131 LES/BUS:' *ELAN_name*':discarded FLUSH Response:source JOIN incomplete, Source LEC ATM address =x *source_LEC_address*

Description: A FLUSH Response was discarded,because the JOIN phase has not completed for the source LEC

LES.132

Level: C_INFO

Short Syntax: LES.132 LES/BUS:' *ELAN_name*':flooded FLUSH RSP:unkwn LECID (x *LECID*), Src LEC ATM addr = x *source_LEC_address*

Long Syntax: LES.132 LES/BUS:' *ELAN_name*':flooded FLUSH Response:unknown LECID (x *LECID*), Source LEC ATM address =x *source_LEC_address*

Description: A FLUSH Response was flooded to all clients because its LECID was unknown to the LES. A common reason for this event is the use of short cut bridging.

LES.133

Level: CE_ERROR

Short Syntax: LES.133 LES/BUS:' *ELAN_name*':dscrd FLUSH RSP:trgt JOIN incmplt, Src LEC ATM addr = x *source_LEC_address*, Trgt LEC ATM addr = x *target_LEC_address*

Long Syntax: LES.133 LES/BUS:' *ELAN_name*':discarded FLUSH Response:target JOIN incomplete, Source LEC ATM address = x

source_LEC_address, Target ATM address = x *target_LEC_address*

Description: A FLUSH Response was discarded, because the JOIN phase has not completed for target LEC

LES.134

Level: CE_ERROR

Short Syntax: LES.134 LES/BUS:' *ELAN_name*':dscrd NARP REQ:JOIN incmlpt, LEC ATM address = x *LEC_address*

Long Syntax: LES.134 LES/BUS:' *ELAN_name*':discarded NARP Request:JOIN incomplete, LEC ATM address = x *LEC_address*

Description: A NARP Request was discarded, because the JOIN phase has not completed

---

LES.135

Level: CE_ERROR

Short Syntax: LES.135 LES/BUS:' *ELAN_name*':dscrd NARP REQ:invld LECID (x *LECID*), LEC ATM address = x *LEC_address*

Long Syntax: LES.135 LES/BUS:' *ELAN_name*':discarded NARP Request:invalid LECID (x *LECID*), LEC ATM address = x *LEC_address*

Description: An NARP Request was discarded, because the LECID is unknown

---

LES.136

Level: CE_ERROR

Short Syntax: LES.136 LES/BUS:' *ELAN_name*':dscrd TPLGY REQ:JOIN incmplt, LEC ATM addr = x *LEC_address*

Long Syntax: LES.136 LES/BUS:' *ELAN_name*':discarded TOPOLOGY Request:JOIN incomplete, LEC ATM address = x *LEC_address*

Description: A TOPOLOGY Request was discarded, because the JOIN phase has not completed

---

LES.137

Level: CE_ERROR

Short Syntax: LES.137 LES/BUS:' *ELAN_name*':dscrd TPLGY REQ:invld LECID (x *LECID*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.137 LES/BUS:' *ELAN_name*':discarded TOPOLOGY Request:invalid LECID (x *LECID*), LEC ATM address = x *LEC_address*

Description: A TOPOLOGY Request was discarded, because the LECID is unknown

---

LES.138

Level: CE_ERROR

Short Syntax: LES.138 LES/BUS:' *ELAN_name*':dscrd cntrl frm:invld Opcode (x *opcode*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.138 LES/BUS:' *ELAN_name*':discarded control frame:invalid Opcode (x *opcode*), LEC ATM address = x *LEC_address*

Description: A control frame was discarded, because the Opcode was invalid.

LES.139

Level: CE_ERROR

Short Syntax: LES.139 LES/BUS:' *ELAN_name*':dscrd Mcast Snd frm:Bus cnnct incmplt, LEC ATM addr = x *LEC_address*

Long Syntax: LES.139 LES/BUS:' *ELAN_name*':discarded Multicast Send frame:Bus connect incomplete, LEC ATM address = x *LEC_address*

Description: A Multicast Send frame was discarded, because the source LEC has not completed the BUS connect phase

---

LES.140

Level: CE_ERROR

Short Syntax: LES.140 LES/BUS:' *ELAN_name*':dscrd Mcast Snd frm:invld prtcl (x *protocol*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.140 LES/BUS:' *ELAN_name*':discarded Multicast Send Frame:invalid protocol (x *protocol*), LEC ATM address = x *LEC_address*

Description: A Multicast Send frame was discarded, because the protocol is invalid. The protocol should be x01

LES.141

Level: CE_ERROR

Short Syntax: LES.141 LES/BUS:' *ELAN_name*':dscrd Mcast Snd frm:invld Vrsn (x *version*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.141 LES/BUS:' *ELAN_name*':discarded Multicast Send frame:invalid Version (x *version*), LEC ATM address = x *LEC_address*

Description: A Multicast Send frame was discarded, because the Version is invalid. The Version should be x01

LES.142

Level: CE_ERROR

Short Syntax: LES.142 LES/BUS:' *ELAN_name*':dscrd Mcast Snd frm:invld Opcode (x *opcode*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.142 LES/BUS:' *ELAN_name*':discarded Multicast Send frame:invalid Opcode (x *opcode*), LEC ATM address = x *LEC_address*

Description: A Multicast Send frame was discarded, because the Opcode is invalid.

LES.144

Level: CE_ERROR

Short Syntax: LES.144 LES/BUS:' *ELAN_name*':dscrd FLUSH REQ:trgt Bus Cnnct incmplt, Src LEC ATM addr = x *source_LEC_address*, Trgt LEC ATM addr = x *target_LEC_address*

Long Syntax: LES.144 LES/BUS:' *ELAN_name*':discarded FLUSH Request:target Bus Connect incomplete, Source LEC ATM address = x *source_LEC_address*, Target LEC ATM address = x *target_LEC_address*

Description: A FLUSH Request was discarded, because the target LEC has not completed the BUS Connect phase.

LES.146

Level: CE_ERROR

Short Syntax: LES.146 LES/BUS:' *ELAN_name*':dscrd dt frm:invld sz (x *frame_size*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.146 LES/BUS:' *ELAN_name*':discarded data frame:invalid size (x *frame_size*), LEC ATM address =x *LEC_address*

Description: A data frame was discarded, because the frame size is invalid.

LES.147

Level: CE_ERROR

Short Syntax: LES.147 LES/BUS:' *ELAN_name*':dscrd dt frm:trgt Bus cnnct incmplt, Src LEC ATM addr = x *source_LEC_address*, Trgt LEC ATM addr = x *target_LEC_address*

Long Syntax: LES.147 LES/BUS:' *ELAN_name*':discarded data frame:target Bus connect incomplete, Source LEC ATM address = x *source_LEC_address*, Target LEC ATM address = x *target_LEC_address*

Description: A data frame was discarded, because the target LEC has not completed the BUS Connect phase.

LES.148

Level: UI_ERROR

Short Syntax: LES.148 LES/BUS:' *ELAN_name*':=>BUS tx err: *error_string* (*error_code*)

Long Syntax: LES.148 LES/BUS:' *ELAN_name*':=>BUS transmit error: *error_string* (*error_code*)

Description: A BUS transmit error occurred. Depending on the severity of the error, the ELAN may be terminated.

LES.149

Level: CE_ERROR

Short Syntax: LES.149 LES/BUS:' *ELAN_name*':trmntng LEC:JOIN parms chngd, LEC ATM addr = x *LEC_address*

Long Syntax: LES.149 LES/BUS:' *ELAN_name*':terminating LEC:JOIN parms changed, LEC ATM address = x *LEC_address*

Description: JOIN parameters have changed, LEC's ELAN membership will be terminated

---

LES.150

Level: C_INFO

Short Syntax: LES.150 LES/BUS:' *ELAN_name*':dscrd dplct JOIN REQ, LEC ATM addr = x *LEC_address*

Long Syntax: LES.150 LES/BUS:' *ELAN_name*':discard duplicate JOIN Request, LEC ATM address = x *LEC_address*

Description: A duplicate JOIN Request was received and discarded

---

LES.151

Level: C_INFO

Short Syntax: LES.151 LES/BUS:' *ELAN_name*':resndng JOIN RSP, LEC ATM addr = x *LEC_address*

Long Syntax: LES.151 LES/BUS:' *ELAN_name*':resending JOIN Response,LEC ATM address = x *LEC_address*

Description: A JOIN Response was resent

LES.152

Level: CE_ERROR

Short Syntax: LES.152 LES/BUS:' *ELAN_name*':JOIN fld:invld LECID (x *LECID*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.152 LES/BUS:' *ELAN_name*':JOIN

failed:invalid LECID (x *LECID*), LEC ATM address = x *LEC_address*

Description: JOIN failed due to invalid LECID. The LECID should be x00

LES.153

Level: CE_ERROR

Short Syntax: LES.153 LES/BUS:' *ELAN_name*':JOIN fld:invld MAC addr (x *MAC_address*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.153 LES/BUS:' *ELAN_name*':JOIN failed:invalid MAC address (x *MAC_address*), LEC ATM address = x *LEC_address*

Description: JOIN failed, because MAC address is invalid

LES.154

Level: CE_ERROR

Short Syntax: LES.154 LES/BUS:' *ELAN_name*':JOIN fld:dplct MAC addr (x *MAC_address*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.154 LES/BUS:' *ELAN_name*':JOIN failed:duplicate MAC address (x *MAC_address*), LEC ATM address = x *LEC_address*

Description: JOIN failed, because MAC address was not unique

LES.156

Level: CE_ERROR

Short Syntax: LES.156 LES/BUS:' *ELAN_name*':JOIN fld:invld LAN Dest Tag (x *LAN_dest_tag*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.156 LES/BUS:' *ELAN_name*':JOIN failed:invalid LAN Destination Tag (x *LAN_dest_tag*), LEC ATM address = x *LEC_address*

Description: JOIN failed, because LAN Dest Tag is invalid

LES.157

Level: CE_ERROR

Short Syntax: LES.157 LES/BUS:' *ELAN_name*':JOIN fld:ATM addr msmtch, Calling ATM addr = x *calling_address*, Src ATM addr = x *source_address*

Long Syntax: LES.157 LES/BUS:' *ELAN_name*':Join failed:ATM address mismatch, Calling ATM address = x *calling_address*, Source ATM address = x *source_address*

Description: JOIN failed, because Source ATM address does not match the Calling Party address

---

LES.158

Level: CE_ERROR

Short Syntax: LES.158 LES/BUS:' *ELAN_name*':JOIN fld:invld Src ATM addr frmt, LEC ATM addr = x *LEC_address*, Src ATM addr = x *source_address*

Long Syntax: LES.158 LES/BUS:' *ELAN_name*':Join failed:invalid Source ATM address format, LEC ATM address = x *LEC_address*, Source ATM address = x *source_address*

Description: JOIN failed,because the Source ATM address format is invalid

---

LES.159

Level: CE_ERROR

Short Syntax: LES.159 LES/BUS:' *ELAN_name*':JOIN fld:dplct ATM addr, LEC ATM addr = x *LEC_address*

Long Syntax: LES.159 LES/BUS:' *ELAN_name*':JOIN failed:duplicate ATM address, LEC ATM address = x *LEC_address*

Description: JOIN failed, because ATM address is not unique

LES.160

Level: CE_ERROR

Short Syntax: LES.160 LES/BUS:' *ELAN_name*':JOIN fld:invld LAN Type (x *LAN_type*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.160 LES/BUS:' *ELAN_name*':JOIN failed:invalid LAN Type (x *LAN_type*), LEC ATM address = x *LEC_address*

Description: JOIN failed, because LAN type is invalid

LES.161

Level: CE_ERROR

Short Syntax: LES.161 LES/BUS:' *ELAN_name*':JOIN fld:invld frm sz (x *frame_size*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.161 LES/BUS:' *ELAN_name*':JOIN failed:invalid frame size (x *frame_size*), LEC ATM address =x *LEC_address*

Description: JOIN failed, because frame size is invalid

LES.162

Level: UI_ERROR

Short Syntax: LES.162 LES/BUS:' *ELAN_name*':JOIN fld:ATM addr CB alloc err, LEC ATM addr = x *LEC_address*

Long Syntax: LES.162 LES/BUS:' *ELAN_name*':JOIN

failed:ATM address Control Block allocation error, LEC ATM address =x *LEC_address*

Description: JOIN failed, because an error occurred while trying to allocate memory for the ATM address Control Block.

Action: Contact your customer service representative

LES.163

Level: UI_ERROR

Short Syntax: LES.163 LES/BUS:' *ELAN_name*':JOIN fld:MAC addr CB alloc err, LEC ATM addr = x *LEC_address*

Long Syntax: LES.163 LES/BUS:' *ELAN_name*':JOIN failed:MAC address Control Block allocation error, LEC ATM address =x *LEC_address*

Description: JOIN failed, because an error occurred while trying to allocate memory for the MAC address Control Block.

Action: Contact your customer service representative

LES.164

Level: UI_ERROR

Short Syntax: LES.164 LES/BUS:' *ELAN_name*':JOIN fld:LECID CB alloc err, LEC ATM addr = x *LEC_address*

Long Syntax: LES.164 LES/BUS:' *ELAN_name*':JOIN failed:LECID Control Block allocation error, LEC ATM address =x *LEC_address*

Description: JOIN failed, because an error occurred while trying to allocate memory for the LECID Control Block.

Action: Contact your customer service representative

LES.165

Level: U_INFO

Short Syntax: LES.165 LES/BUS:' *ELAN_name*':JOIN fld:all LECIDs in use, LEC ATM addr = x *LEC_address*

Long Syntax: LES.165 LES/BUS:' *ELAN_name*':JOIN failed:all LECIDs in use , LEC ATM address =x *LEC_address*

Description: JOIN failed, because all LECIDs are in use

LES.166

Level: CE_ERROR

Short Syntax: LES.166 LES/BUS:' *ELAN_name*':trmntng LEC:JOIN time-out, LEC ATM addr = x *LEC_address*

Long Syntax: LES.166 LES/BUS:' *ELAN_name*':terminating LEC:JOIN time-out, LEC ATM address = x *LEC_address*

Description: JOIN phase has not completed before timer expired, LEC's ELAN membership will be terminated

---

LES.167

Level: UI_ERROR

Short Syntax: LES.167 LES/BUS:' *ELAN_name*':=>DOWN:LECID DB add err: *error_string* (*error_code*)

Long Syntax: LES.167 LES/BUS:' *ELAN_name*':DOWN:LECID DataBase add error: *error_string* (*error_code*)

Description: An error occurred while trying to add an entry to the LECID DataBase. The ELAN will be terminated

Action: Contact your customer service representative

LES.168

Level: C_INFO

Short Syntax: LES.168 LES/BUS:' *ELAN_name*':plcng *VCC_type* call, LEC ATM addr = x *LEC_address*

Long Syntax: LES.168 LES/BUS:' *ELAN_name*':placing *VCC_type* call, LEC ATM address = x *LEC_address*

Description: A call is being placed for the given VCC type

LES.169

Level: UI_ERROR

Short Syntax: LES.169 LES/BUS:' *ELAN_name*':JOIN fld:err plcng *VCC_type* call: *error_string* (*error_code*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.169 LES/BUS:' *ELAN_name*':JOIN failed:error placing *VCC_type* call: *error_string* (*error_code*), LEC ATM address = x *LEC_address*

Description: JOIN failed, unable to place call due to lack of memory

Action: Contact your customer service representative

LES.170

Level: UI_ERROR

Short Syntax: LES.170 LES/BUS:' *ELAN_name*':=>DOWN:err plcng *VCC_type* call: *error_string* (*error_code*)

Long Syntax: LES.170 LES/BUS:' *ELAN_name*':=>DOWN:error placing *VCC_type* call: *error_string* (*error_code*)

Description: An error occurred while trying to place a call

LES.171

Level: C_INFO

Short Syntax: LES.171 LES/BUS:' *ELAN_name*':wtng to add *VCC_type* leaf, LEC ATM addr = x *LEC_address*

Long Syntax: LES.171 LES/BUS:' *ELAN_name*':waiting to add *VCC_type* leaf,LEC ATM address = x *LEC_address*

Description: Call signaling in progress, waiting for completion

---

LES.172

Level: C_INFO

Short Syntax: LES.172 LES/BUS:' *ELAN_name*':adding *VCC_type* leaf, LEC ATM addr = x *LEC_address*

Long Syntax: LES.172 LES/BUS:' *ELAN_name*':adding *VCC_type* leaf,LEC ATM address = x *LEC_address*

Description: A leaf is being added

LES.173

Level: UI_ERROR

Short Syntax: LES.173 LES/BUS:' *ELAN_name*':JOIN fld:err adding *VCC_type* leaf: *error_string* (*error_code*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.173 LES/BUS:' *ELAN_name*':JOIN failed:error adding *VCC_type* leaf: *error_string* (*error_code*),LEC ATM address = x *LEC_address*

Description: JOIN failed, because an error occurred when adding a leaf

---

LES.174

Level: UI_ERROR

Short Syntax: LES.174 LES/BUS:' *ELAN_name*':=>DOWN:err adding *VCC_type* leaf: *error_string* (*error_code*)

Long Syntax: LES.174 LES/BUS:' *ELAN_name*':=>DOWN:error adding *VCC_type* leaf: *error_string* (*error_code*)

Description: An error occurred when adding a leaf, the ELAN will be terminated

---

LES.175

Level: C_INFO

Short Syntax: LES.175 LES/BUS:' *ELAN_name*':dscrd FLUSH REQ:trgtd for BUS, LEC ATM addr = x *LEC_address*

Long Syntax: LES.175 LES/BUS:' *ELAN_name*':discarded FLUSH Request:targeted for BUS, LEC ATM address = x *LEC_address*

Description: FLUSH Request was discarded, because it was targeted for BUS

LES.176

Level: UI_ERROR

Short Syntax: LES.176 LES/BUS:' *ELAN_name*':trmntng LEC:err plcng *VCC_type* call: *error_string* (*error_code*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.176 LES/BUS:' *ELAN_name*':terminating LEC:error placing *VCC_type* call: *error_string* (*error_code*), LEC ATM address = x *LEC_address*

Description: Unable to place call due to lack of memory

Action: Contact your customer service representative

LES.177

Level: UI_ERROR

Short Syntax: LES.177 LES/BUS:' *ELAN_name*':trmntng LEC:err adding *VCC_type* leaf: *error_string* (*error_code*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.177 LES/BUS:' *ELAN_name*':terminating LEC:error adding *VCC_type* leaf: *error_string* (*error_code*), LEC ATM address = x *LEC_address*

Description: An error occured when adding a leaf

LES.178

Level: CE_ERROR

Short Syntax: LES.178 LES/BUS:' *ELAN_name*':dscrd REG REQ:JOIN incmplt, LEC ATM addr = x *LEC_address*

Long Syntax: LES.178 LES/BUS:' *ELAN_name*':discarded Register Request:JOIN incomplete, LEC ATM addr = x *LEC_address*

Description: Register Request was discarded, because the JOIN phase has not completed

---

LES.179

Level: CE_ERROR

Short Syntax: LES.179 LES/BUS:' *ELAN_name*':REG fld:invld LECID (*LECID*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.179 LES/BUS:' *ELAN_name*':Registration failed:invalid LECID (

LECID), LEC ATM addr = x *LEC_address*

Description: Registration failed, because the LECID is invalid

---

LES.180

Level: CE_ERROR

Short Syntax: LES.180 LES/BUS:' *ELAN_name*':REG fld:invld Src ATM addr frmt, LEC ATM addr = x *LEC_address*, Source ATM addr = x *source_address*

Long Syntax: LES.180 LES/BUS:' *ELAN_name*':Registration failed:invalid Source ATM address format, LEC ATM addr = x *LEC_address*, Source ATM address = x *source_address*

Description: Registration failed, because the source ATM address format is invalid

---

LES.181

Level: CE_ERROR

Short Syntax: LES.181 LES/BUS:' *ELAN_name*':REG fld:invld MAC addr (x *MAC_address*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.181 LES/BUS:' *ELAN_name*':Registration failed:invalid MAC address (x *MAC_address*), LEC ATM addr = x *LEC_address*

Description: Registration failed, because the MAC address is invalid

---

LES.182

Level: CE_ERROR

Short Syntax: LES.182 LES/BUS:' *ELAN_name*':REG fld:dplct MAC addr (x *MAC_address*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.182 LES/BUS:' *ELAN_name*':Registration failed:duplicate MAC address (x *MAC_address*), LEC ATM addr = x *LEC_address*

Description: Registration failed, MAC address is not unique

LES.183

Level: CE_ERROR

Short Syntax: LES.183 LES/BUS:' *ELAN_name*':REG fld:dplct Src ATM addr, LEC ATM addr = x *LEC_address*, Src ATM addr = x *source_address*

Long Syntax: LES.183 LES/BUS:' *ELAN_name*':Registration failed:duplicate Source ATM address, LEC ATM addr = x *LEC_address*, Source ATM address = x *source_address*

Description: Registration failed, Source ATM address is not unique

---

LES.184

Level: UI_ERROR

Short Syntax: LES.184 LES/BUS:' *ELAN_name*':REG fld:ATM addr CB alloc err, LEC ATM addr = x *LEC_address*

Long Syntax: LES.184 LES/BUS:' *ELAN_name*':Registration failed:ATM address Control Block allocation error, LEC ATM address =x *LEC_address*

Description: Registration failed, because an error occurred while trying to allocate memory for the ATM address Control Block

Action: Contact your customer service representative

LES.185

Level: UI_ERROR

Short Syntax: LES.185 LES/BUS:' *ELAN_name*':REG fld:MAC addr CB alloc err, LEC ATM addr = x *LEC_address*

Long Syntax: LES.185 LES/BUS:' *ELAN_name*':Registration failed:MAC address Control Block allocation error, LEC ATM address =x *LEC_address*

Description: Registration failed, because an error occurred while trying to allocate memory for the MAC address Control Block

Action: Contact your customer service representative

LES.186

Level: CE_ERROR

Short Syntax: LES.186 LES/BUS:' *ELAN_name*':REG fld:RD on Eth ELAN, LEC ATM addr = x *LEC_address*

Long Syntax: LES.186 LES/BUS:' *ELAN_name*':Registration failed:Route Descriptor on Ethernet ELAN, LEC ATM address = x *LEC_address*

Description: Registration failed, Route Descriptors are not allowed on Ethernet ELANs

---

LES.187

Level: CE_ERROR

Short Syntax: LES.187 LES/BUS:' *ELAN_name*':REG fld:dplct RD (x *route_descriptor*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.187 LES/BUS:' *ELAN_name*':Registration failed:duplicate Route Descriptor (x *route_descriptor*), LEC ATM address = x *LEC_address*

Description: Registration failed, Route Descriptor is not unique

LES.188

Level: UI_ERROR

Short Syntax: LES.188 LES/BUS:' *ELAN_name*':REG fld:RD CB alloc err, LEC ATM addr = x *LEC_address*

Long Syntax: LES.188 LES/BUS:' *ELAN_name*':Registration failed:Route Descriptor Control Block allocation error, LEC ATM address =x *LEC_address*

Description: Registration failed, because an error occured while trying to allocate memory for the Route Descriptor Control Block

Action: Contact your customer service representative

LES.189

Level: CE_ERROR

Short Syntax: LES.189 LES/BUS:' *ELAN_name*':REG fld:invld LAN Dest Tag (x *LAN_dest_tag*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.189 LES/BUS:' *ELAN_name*':Registration failed:invalid LAN Destination Tag (x *LAN_dest_tag*), LEC ATM address = x *LEC_address*

Description: Registration failed, because the LAN Dest Tag is invalid

---

LES.190

Level: CE_ERROR

Short Syntax: LES.190 LES/BUS:' *ELAN_name*':dscrd UNREG REQ:JOIN incmplt, LEC ATM addr = x *LEC_address*

Long Syntax: LES.190 LES/BUS:' *ELAN_name*':discarded Unregister Request:JOIN incomplete, LEC ATM address = x *LEC_address*

Description: Unregister Request discarded, because JOIN phase has not completed

---

LES.191

Level: CE_ERROR

Short Syntax: LES.191 LES/BUS:' *ELAN_name*':UNREG fld:invld LECID (*LECID*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.191 LES/BUS:' *ELAN_name*':Unregister failed:invalid LECID (*LECID*), LEC ATM addr = x *LEC_address*

Description: Unregister failed, because the LECID is invalid

LES.192

Level: CE_ERROR

Short Syntax: LES.192 LES/BUS:' *ELAN_name*':UNREG fld:invld Src ATM addr frmt, LEC ATM addr = x *LEC_address*, Source ATM addr = x *source_address*

Long Syntax: LES.192 LES/BUS:' *ELAN_name*':Unregister failed:invalid Source ATM address format, LEC ATM addr = x *LEC_address*, Source ATM address = x *source_address*

Description: Unregister failed, because the Source ATM address format is invalid

---

LES.193

Level: CE_ERROR

Short Syntax: LES.193 LES/BUS:' *ELAN_name*':UNREG fld:invld MAC addr (x *MAC_address*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.193 LES/BUS:' *ELAN_name*':Unregister failed:invalid MAC address (x *MAC_address*), LEC ATM addr = x *LEC_address*

Description: Unregister failed, because the MAC address is invalid

---

LES.194

Level: CE_ERROR

Short Syntax: LES.194 LES/BUS:' *ELAN_name*':UNREG fld:RD on Eth ELAN, LEC ATM addr = x *LEC_address*

Long Syntax: LES.194 LES/BUS:' *ELAN_name*':Unregister failed:Route Descriptor on Ethernet ELAN, LEC ATM address = x *LEC_address*

Description: Unregister failed, Route Descriptors are not allowed on Ethernet ELANs

---

LES.195

Level: CE_ERROR

Short Syntax: LES.195 LES/BUS:' *ELAN_name*':UNREG fld:invld LAN Dest Tag (x *LAN_dest_tag*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.195 LES/BUS:' *ELAN_name*':Unregister failed:invalid LAN Destination Tag (x *LAN_dest_tag*), LEC ATM address = x *LEC_address*

Description: Unregister failed, because LAN Dest Tag is invalid

---

LES.196

Level: CE_ERROR

Short Syntax: LES.196 LES/BUS:' *ELAN_name*':dscrd ARP REQ:JOIN incmlpt, LEC ATM addr = x *LEC_address*

Long Syntax: LES.196 LES/BUS:' *ELAN_name*':discarded ARP Request:JOIN incomplete, LEC ATM address = x *LEC_address*

Description: ARP Request was discarded, because the JOIN phase has not completed

---

LES.197

Level: CE_ERROR

Short Syntax: LES.197 LES/BUS:' *ELAN_name*':ARP fld:invld LECID (*LECID*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.197 LES/BUS:' *ELAN_name*':ARP failed:invalid LECID (*LECID*), LEC ATM address = x *LEC_address*

Description: ARP failed, because LECID is invalid

LES.198

Level: CE_ERROR

Short Syntax: LES.198 LES/BUS:' *ELAN_name*':ARP fld:mcast MAC addr (x *MAC_address*), LANEv1 LEC ATM addr = x *LEC_address*

Long Syntax: LES.198 LES/BUS:' *ELAN_name*':ARP failed:multicast MAC address (x *MAC_address*), LANEv1 LEC ATM address = x *LEC_address*

Description: ARP failed, because MAC address is multicast and LEC is LANEv1.

LES.199

Level: CE_ERROR

Short Syntax: LES.199 LES/BUS:' *ELAN_name*':dscrd ARP REQ:trgt JOIN incmlpt, Src LEC ATM addr = x *source_LEC_address*, Trgt LEC ATM addr = x *target_LEC_address*

Long Syntax: LES.199 LES/BUS:' *ELAN_name*':discarded ARP Request:target JOIN incomplete, Source LEC ATM address = x *source_LEC_address*, Target LEC ATM address = x *target_LEC_address*

Description: ARP Request was discarded, because the JOIN phase has not completed for the target LEC

LES.200

Level: CE_ERROR

Short Syntax: LES.200 LES/BUS:' *ELAN_name*':ARP fld:RD on Eth ELAN, LEC ATM addr = x *LEC_address*

Long Syntax: LES.200 LES/BUS:' *ELAN_name*':ARP failed:Route Descriptor on Ethernet ELAN, LEC ATM address = x *LEC_address*

Description: ARP failed, Route Descriptors are not allowed on Ethernet ELANs

---

LES.201

Level: CE_ERROR

Short Syntax: LES.201 LES/BUS:' *ELAN_name*':dscrd FLUSH REQ:no Proxy Mcast Fwd, Src LEC ATM addr = x *source_LEC_address*, Trgt LEC ATM addr = x *target_LEC_address*

Long Syntax: LES.201 LES/BUS:' *ELAN_name*':discarded Flush Request:no Proxy Multicast Forward, Source LEC ATM address = x *source_LEC_address*, Target LEC ATM addr = x *target_LEC_address*

Description: Flush Request was discarded, because Proxy Multicast Forward VCC is not operational

---

LES.202

Level: CE_ERROR

Short Syntax: LES.202 LES/BUS:' *ELAN_name*':dscrd ARP REQ:invld LAN Dest Tag (x *LAN_dest_tag*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.202 LES/BUS:' *ELAN_name*':discarded ARP Request:invalid LAN Dest Tag (x *LAN_dest_tag*), LEC ATM address = x *LEC_address*

Description: ARP Request was discarded, LAN Dest Tag is invalid

---

LES.203

Level: UI_ERROR

Short Syntax: LES.203 LES/BUS:' *ELAN_name*':=>LES tx err: *error_string* (*error_code*)

Long Syntax: LES.203 LES/BUS:' *ELAN_name*':=>LES transmit error: *error_string* (*error_code*)

Description: A LES transmit error occurred. Depending on the severity of the error, the ELAN may be terminated.

LES.204

Level: UI_ERROR

Short Syntax: LES.204 LES/BUS:' *ELAN_name*':=>DOWN:ATM addr DB add err: *error_string* (*error_code*)

Long Syntax: LES.204 LES/BUS:' *ELAN_name*':DOWN:ATM address DataBase add error: *error_string* (*error_code*)

Description: An error occurred while trying to add ATM address to database, the ELAN will be terminated

Action: Contact your customer service representative

LES.205

Level: UI_ERROR

Short Syntax: LES.205 LES/BUS:' *ELAN_name*':=>DOWN:MAC addr DB add err: *error_string* (*error_code*)

Long Syntax: LES.205 LES/BUS:' *ELAN_name*':DOWN:MAC address DataBase add error: *error_string* (*error_code*)

Description: An error occurred while trying to add MAC address to database, the ELAN will be terminated

Action: Contact your customer service representative

LES.206

Level: UI_ERROR

Short Syntax: LES.206 LES/BUS:' *ELAN_name*':=>DOWN:RD DB add err: *error_string* (*error_code*)

Long Syntax: LES.206 LES/BUS:' *ELAN_name*':DOWN:Route Descriptor DataBase add error: *error_string* (*error_code*)

Description: An error occurred while trying to add Route Descriptor to database, the ELAN will be terminated

Action: Contact your customer service representative

LES.209

Level: C_INFO

Short Syntax: LES.209 LES/BUS:' *ELAN_name*': *VCC_type* call fld:retrying temp failure, LEC ATM addr = x *LEC_address*

Long Syntax: LES.209 LES/BUS:' *ELAN_name*': *VCC_type* call failed:retrying temporary failure, LEC ATM address = x *LEC_address*

Description: A call failed due to a temporary condition, the call will be retried.

LES.211

Level: C_INFO

Short Syntax: LES.211 LES/BUS:' *ELAN_name*':err adding *VCC_type* leaf:cause *cause_code*, LEC ATM addr = x *LEC_address*

Long Syntax: LES.211 LES/BUS:' *ELAN_name*':error adding *VCC_type* leaf:cause *cause_code*, LEC ATM address = x *LEC_address*

Description: An error occurred when trying to add a leaf

LES.213

Level: U_INFO

Short Syntax: LES.213 BCM:' *ELAN_name*':initlzd

Long Syntax: LES.213 BCM:' *ELAN_name*':initialized

Description: BCM for this ELAN has been initialized

LES.214

Level: U_INFO

Short Syntax: LES.214 BCM:' *ELAN_name*':HALTED

Long Syntax: LES.214 BCM:' *ELAN_name*':HALTED

Description: BCM for this ELAN has been halted. No protocols are active

---

LES.215

Level: U_INFO

Short Syntax: LES.215 BCM:' *ELAN_name*':STARTED/RESTARTED prtcl *protocol_name*

Long Syntax: LES.215 BCM:' *ELAN_name*':STARTED/RESTARTED protocol *protocol_name*

Description: BCM for this ELAN has been started (or restarted) for the given protocol

---

LES.216

Level: U_INFO

Short Syntax: LES.216 BCM:' *ELAN_name*':STOPPED prtcl *protocol_name*

Long Syntax: LES.216 BCM:' *ELAN_name*':STOPPED protocol *protocol_name*

Description: BCM for this ELAN has been stopped for the given protocol Frames will not be processed by BCM for the protocol, existing protocol entries will be aged out over time

---

LES.217

Level: C_INFO

Short Syntax: LES.217 BCM:' *ELAN_name*':notfd of LEC actvn, ATM addr = x *LEC_address*

Long Syntax: LES.217 BCM:' *ELAN_name*':notified of LEC activation, ATM address = x *LEC_address*

Description: BCM was notified of a LEC becoming active on this ELAN

LES.218

Level: C_INFO

Short Syntax: LES.218 BCM:' *ELAN_name*':dltd all prtcls from MAC addr x *MAC_address* due to *cause_string*

Long Syntax: LES.218 BCM:' *ELAN_name*':deleted all protocols from MAC address x *MAC_address* due to *cause_string*

Description: BCM has deleted all cached protocol addresses from the given MAC address due to the given cause

---

LES.219

Level: C_INFO

Short Syntax: LES.219 BCM:' *ELAN_name*':notfd of LEC term, ATM addr = x *LEC_address*

Long Syntax: LES.219 BCM:' *ELAN_name*':notified of LEC termination, ATM address = x *LEC_address*

Description: BCM was notified of a LEC being terminated on this ELAN

LES.220

Level: C_INFO

Short Syntax: LES.220 BCM:' *ELAN_name*':notfd of MAC rgstrn, MAC addr = x *MAC_address* ATM addr = x *LEC_address*

Long Syntax: LES.220 BCM:' *ELAN_name*':notified of MAC registration, MAC address = x *MAC_address* ATM address = x *LEC_address*

Description: BCM was notified of a MAC address being registered on this ELAN

LES.221

Level: C_INFO

Short Syntax: LES.221 BCM:' *ELAN_name*':dltd Lrnd MAC addr x *MAC_address* due to *cause_string*

Long Syntax: LES.221 BCM:' *ELAN_name*':deleted Learned MAC address x *MAC_address* due to *cause_string*

Description: BCM has deleted a Learned MAC address from the cache due to the given cause

---

LES.222

Level: U_INFO

Short Syntax: LES.222 BCM:' *ELAN_name*':SHUT DOWN BCM for prtcl *protocol_name*

Long Syntax: LES.222 BCM:' *ELAN_name*':SHUT DOWN BCM for protocol *protocol_name*

Description: BCM for this ELAN has been shut down for the given protocol. Frames will not be processed by BCM for the protocol, all existing protocol entries have been deleted.

LES.223

Level: UI_ERROR

Short Syntax: LES.223 BCM:' *ELAN_name*':net hndlr err on Opn Grp VCC: *error_string* (*error_code*)

Long Syntax: LES.223 BCM:' *ELAN_name*':net handler error on Open Group VCC: *error_string* (*error_code*)

Description: ATM Device Driver call to open a Group VCC was not successful

Action: Contact your customer service representative

LES.224

Level: UI_ERROR

Short Syntax: LES.224 BCM:' *ELAN_name*':SHUT DOWN BCM for prtcl IPX. net hndlr err: *error_string* (*error_code*)

Long Syntax: LES.224 BCM:' *ELAN_name*':SHUT DOWN BCM for protocol IPX. net handler error: *error_string* (*error_code*)

Description: ATM Device Driver call to add to a Group VCC was not successful

Action: Contact your customer service representative

LES.225

Level: C_INFO

Short Syntax: LES.225 BCM:' *ELAN_name*':added VCC to grp VCC for prtcl *protocol_name*, LEC ATM addr = x *LEC_address*

Long Syntax: LES.225 BCM:' *ELAN_name*':added VCC to group VCC for protocol *protocol_name*, LEC ATM address = x *LEC_address*

Description: BCM has added a VCC to the Group VCC for the given protocol

---

LES.226

Level: UI_ERROR

Short Syntax: LES.226 BCM:' *ELAN_name*':SHUT DOWN BCM for prtcl IPX. warn: excd max *ipx_cutoff* in IPX grp VCC.

Long Syntax: LES.226 BCM:' *ELAN_name*':SHUT DOWN BCM for protocol IPX. Warning: exceeded maximum *ipx_cutoff* in IPX group VCC.

Description: BCM IPX has automatically disabled itself. This protective mechanism is triggered when more than the specified number of unique IPX Routers and Servers are discovered in the IPX network containing this ELAN. The reasoning is as follows. Say BCM has learned N unique IPX Routers/Servers in the ELAN. Each IPX broadcast frame received by the BUS is transformed into N unicast frames, once for each IPX Router/Server, and transmitted on the Multicast Send VCCs to the destinations. When N is large, this results in excessive retransmissions which can degrade the performance of the system and the network. Automatically disabling BCM IPX at this point allows the BUS to process a single broadcast frame as usual.

Action: One possible action is to turn BCM for IPX off. This will remove BCM for IPX from the data path in the future. Another possible action is to use BCM static targets. If there are a large number of IPX Routers/Servers located behind a small number of LECs, then these LECs can be defined as BCM static targets. IPX broadcast frames are transmitted only once to each BCM static target. BCM for IPX may still learn additional unique IPX Routers/Servers behind other LECs, up to the number specified in this message. The current limit on the number of BCM static targets is 3. A third possible action is to configure a higher value for the maximum number of BCM IPX entries in its transmit list. Use this third action with caution due to the impact on the performance of the network and on this device.

LES.227

Level: C_INFO

Short Syntax: LES.227 BCM:' *ELAN_name*':added MAC to grp VCC for prtcl *protocol_name*, MAC addr = x *MAC_address* LEC ATM addr = x *LEC_address*

Long Syntax: LES.227 BCM:' *ELAN_name*':added MAC to group VCC for protocol *protocol_name*, MAC address = x *MAC_address* LEC ATM address = x *LEC_address*

Description: BCM has added a MAC address to the Group VCC for the given protocol

---

LES.228

Level: U_INFO

Short Syntax: LES.228 BCM:' *ELAN_name*':cant add VCC to grp VCC for prtcl *protocol_name*, LEC ATM addr = x *LEC_address*

Long Syntax: LES.228 BCM:' *ELAN_name*':can not add VCC to group VCC for protocol *protocol_name*, LEC ATM address = x *LEC_address*

Description: BCM can not add a VCC to the Group VCC for the given protocol. Either the Group VCC is not valid, or the LEC is not operational from the point of view of the BUS.

LES.229

Level: C_INFO

Short Syntax: LES.229 BCM:' *ELAN_name*':dltd MAC from grp VCC for prtcl *protocol_name*, MAC addr = x *MAC_address* LEC ATM addr = x *LEC_address*

Long Syntax: LES.229 BCM:' *ELAN_name*':deleted MAC from group VCC for protocol *protocol_name*, MAC address = x *MAC_address* LEC ATM address = x *LEC_address*

Description: BCM has deleted a MAC address from the Group VCC for the given protocol

---

LES.230

Level: C_INFO

Short Syntax: LES.230 BCM:' *ELAN_name*':dltd VCC from grp VCC for prtcl *protocol_name*, LEC ATM addr = x *LEC_address*

Long Syntax: LES.230 BCM:' *ELAN_name*':deleted VCC from group VCC for protocol *protocol_name*, LEC ATM address = x *LEC_address*

Description: BCM has deleted a VCC from the Group VCC for the given protocol

---

LES.231

Level: U_INFO

Short Syntax: LES.231 BCM:' *ELAN_name*':grp VCC for prtcl *protocol_name* mssng MAC x *MAC_address* or VCC (ATM addr x *LEC_address*) Code ' *error_string*' (*error_code*) due to abnrml LEC term?

Long Syntax: LES.231 BCM:' *ELAN_name*':group VCC for protocol *protocol_name* is missing MAC x *MAC_address* or VCC (to LEC ATM address x *LEC_address*). Code ' *error_string*' (*error_code*) may be due to abnormal LEC termination.

Description: When attempting to unmap a MAC address from the Group VCC for the given protocol, BCM got an unexpected return code. This may be due

to abnormal LEC termination, which should also be logged. The MAC, LEC's ATM address, and unexpected return code are given.

LES.232

Level: U_INFO

Short Syntax: LES.232 BCM:' *ELAN_name*':Rst lcl IPX net info

Long Syntax: LES.232 BCM:' *ELAN_name*':Reset local IPX network information

Description: The last destination on the IPX Group VCC for this ELAN was just removed. BCM has reset the local IPX network information.

LES.233

Level: U_INFO

Short Syntax: LES.233 BCM:' *ELAN_name*':NetBIOS NAME_IN_CONFLICT rcvd. dltd name *protocol_address*

Long Syntax: LES.233 BCM:' *ELAN_name*':NetBIOS NAME_IN_CONFLICT received. deleted name *protocol_address*

Description: NetBIOS BCM has detected a NAME_IN_CONFLICT. Duplicate NetBIOS names were in use in the network of which this ELAN is part. This situation could arise if an outage in the network was just remedied. (BCM has deleted the NetBIOS name from the cache.)

LES.234

Level: U_INFO

Short Syntax: LES.234 BCM:' *ELAN_name*':dltd all Lrnd MAC addrs

Long Syntax: LES.234 BCM:' *ELAN_name*':deleted all Learned MAC addresses

Description: All Learned MAC addresses were deleted.

LES.235

Level: U_INFO

Short Syntax: LES.235 BCM:' *ELAN_name*':dltd all *protocol_name* prtcl entries

Long Syntax: LES.235 BCM:' *ELAN_name*':deleted all entries for protocol *protocol_name*

Description: All protocol entries for the given protocol were deleted.

LES.236

Level: UI_ERROR

Short Syntax: LES.236 BCM:' *ELAN_name*':add to cache fld. prtcl CB alloc err

Long Syntax: LES.236 BCM:' *ELAN_name*':add to cache failed. protocol control block allocation error

Description: BCM could not add a new protocol address because an error occurred while trying to allocate memory for the protocol control block.

Action: Contact your customer service representative

LES.237

Level: UI_ERROR

Short Syntax: LES.237 BCM:' *ELAN_name*':add to cache fld. MAC addr CB alloc err

Long Syntax: LES.237 BCM:' *ELAN_name*':add to cache failed. MAC address control block allocation error

Description: BCM could not add a new learned MAC address because an error occurred while trying to allocate memory for the MAC control block.

Action: Contact your customer service representative

LES.238

Level: UE_ERROR

Short Syntax: LES.238 BCM:' *ELAN_name*':rcvd frm from MAC x *MAC_address*, LEC ATM addr = x *LEC_address*. conflicts with rgstrn by LEC ATM addr = x *LEC_address*

Long Syntax: LES.238 BCM:' *ELAN_name*':received frame from MAC x *MAC_address*, LEC ATM address = x *LEC_address*. conflicts with registration by LEC ATM address = x *LEC_address*

Description: BCM has received a frame on this ELAN from the given MAC address from a different LEC than the LEC that registered that MAC address. A MAC address registered by a LEC is assumed to be unique. Perhaps duplicate MAC addresses exist in the network. This message is only logged one time while the MAC is registered, no matter how many frames are received with this MAC address.

Action: Ensure the MAC addresses in the network are unique.

LES.239

Level: C_INFO

Short Syntax: LES.239 BCM:' *ELAN_name*':added *protocol_type_string protocol_address* on MAC addr x *MAC_address*

Long Syntax: LES.239 BCM:' *ELAN_name*':added *protocol_type_string protocol_address* on MAC address x *MAC_address*

Description: BCM learned the given protocol address on the given MAC address.

LES.240

Level: C_INFO

Short Syntax: LES.240 BCM:' *ELAN_name*':added Lrnd MAC addr x *MAC_address*, LEC ATM addr = x *LEC_address*

Long Syntax: LES.240 BCM:' *ELAN_name*':added Learned MAC address x *MAC_address*, LEC ATM address = x *LEC_address*

Description: BCM learned the given MAC address. This MAC address has not been registered by any LEC in the given ELAN.

LES.241

Level: C_INFO

Short Syntax: LES.241 BCM:' *ELAN_name*':aged *protocol_type_string protocol_address* on MAC addr x *MAC_address* from cache

Long Syntax: LES.241 BCM:' *ELAN_name*':aged *protocol_type_string protocol_address* on MAC address x *MAC_address* from cache

Description: BCM aged out the given protocol address on the given MAC address.

LES.242

Level: U_INFO

Short Syntax: LES.242 BCM:' *ELAN_name*':stpd rapid aging

Long Syntax: LES.242 BCM:' *ELAN_name*':stopped rapid aging

Description: In this ELAN, the Forward Delay Timer has expired following a Spanning Tree Topology Change. BCM has aged out all non-local protocol addresses and learned MAC addresses.

LES.243

Level: U_INFO

Short Syntax: LES.243 BCM:' *ELAN_name*':strtd rapid aging

Long Syntax: LES.243 BCM:' *ELAN_name*':started rapid aging

Description: In this ELAN, a Spanning Tree Topology Change was detected. By the time the Forward Delay Timer has expired, BCM will have aged out all non-local protocol addresses and learned MAC addresses.

LES.244

Level: C_INFO

Short Syntax: LES.244 BCM:' *ELAN_name*':set *protocol_type_string protocol_address* age to *age*

Long Syntax: LES.244 BCM:' *ELAN_name*':set *protocol_type_string protocol_address* age to *age*

Description: The given protocol address age was set to the given age.

LES.245

Level: C_INFO

Short Syntax: LES.245 BCM:' *ELAN_name*':dltd *protocol_type_string protocol_address* from MAC addr x *MAC_address*

Long Syntax: LES.245 BCM:' *ELAN_name*':deleted *protocol_type_string protocol_address* from MAC address x *MAC_address*

Description: BCM has deleted a protocol address from the given MAC address

---

LES.246

Level: C_INFO

Short Syntax: LES.246 SRM:' *ELAN_name*':added route *Route_string*, LEC ATM addr = x *LEC_address*

Long Syntax: LES.246 SRM:' *ELAN_name*':added route *Route_string*, LEC ATM address = x *LEC_address*

Description: SRM added the given route on the given LEC ATM address

---

LES.247

Level: C_INFO

Short Syntax: LES.247 SRM:' *ELAN_name*':rplcd route *Route_string*, LEC ATM addr = x *LEC_address*

Long Syntax: LES.247 SRM:' *ELAN_name*':replaced route *Route_string*, LEC ATM address = x *LEC_address*

Description: SRM replaced the given route on the given LEC ATM address because it was deemed better than the current route cached.

LES.248

Level: U_INFO

Short Syntax: LES.248 SRM:' *ELAN_name*':WRNG: SRM out of resources.

Long Syntax: LES.248 SRM:' *ELAN_name*':WARNING: Source Route Management out of resources.

Description: SRM for this ELAN has encountered an Out of Resources condition. SRM is not shut down. Entries will be aged out if the condition persists.

LES.249

Level: C_INFO

Short Syntax: LES.249 SRM:' *ELAN_name*':dltd route *Route_string* LEC ATM addr = x *LEC_address* due to *cause_string*

Long Syntax: LES.249 SRM:' *ELAN_name*':deleted route *Route_string*, LEC ATM address = x *LEC_address* due to *cause_string*

Description: SRM has deleted the given route on the given LEC ATM address for the given reason.

LES.250

Level: C_INFO

Short Syntax: LES.250 SRM:' *ELAN_name*':aged rte *Route_string* on LEC ATM addr x *LEC_address* from cache

Long Syntax: LES.250 SRM:' *ELAN_name*':aged route *Route_string* on LEC ATM address x *LEC_address* from cache

Description: SRM aged out the given route on the given LEC ATM address

---

LES.251

Level: CE_ERROR

Short Syntax: LES.251 LES/BUS:' *ELAN_name*':dscrd data frm:no Proxy Mcast Fwd, Src LEC ATM addr = x *source_LEC_address*,

Long Syntax: LES.251 LES/BUS:' *ELAN_name*':discarded Flush Request:no Proxy Multicast Forward, Source LEC ATM address = x *source_LEC_address*

Description: Data frame was discarded, because Proxy Multicast Forward VCC is not operational

---

LES.252

Level: CE_ERROR

Short Syntax: LES.252 LES/BUS:' *ELAN_name*':dscrd *frameType* frm:no Proxy Ctrl Dist, Src LEC ATM addr = x *source_LEC_address*,

Long Syntax: LES.252 LES/BUS:' *ELAN_name*':discarded *frameType* frame:no Proxy Control Distribute, Source LEC ATM address = x *source_LEC_address*

Description: A frame of the specified type was discarded. It was to be forwarded over the Proxy Control Distribute VCC, but the Proxy Control Distribute VCC is not operational. This is most likely caused by no proxy clients joining the ELAN.

LES.253

Level: U_INFO

Short Syntax: LES.253 LES/BUS:' *ELAN_name*':dscrd *protocol_name* frm due to *cause_string*, Src LEC ATM addr = x *LEC_address*,

Long Syntax: LES.253 LES/BUS:' *ELAN_name*':discarded *protocol_name* frame due to *cause_string*, Source LEC ATM address = x *LEC_address*

Description: A data frame of the given protocol type was discarded for the given reason.

LES.254

Level: CE_ERROR

Short Syntax: LES.254 LES/BUS:' *ELAN_name*':dscrd cntrl frm:invld sz (x *frame_size*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.254 LES/BUS:' *ELAN_name*':discarded control frame:invalid size (x *frame_size*), LEC ATM addr = x *LEC_address*

Description: A control frame sent to the LES was discarded because the actual size was invalid.

LES.255

Level: CE_ERROR

Short Syntax: LES.255 LES/BUS:' *ELAN_name*':dscrd Mcast Snd frm:invld sz (x *frame_size*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.255 LES/BUS:' *ELAN_name*':discarded Multicast Send frame:invalid size (x *frame_size*), LEC ATM addr = x *LEC_address*

Description: A control frame sent to the BUS was discarded because the actual size was invalid.

LES.256

Level: P_TRACE

Short Syntax: LES.256 Trace LAN Emulation Control frame.

Long Syntax: LES.256 Trace LAN Emulation Control frame.

Description: LAN emulation control frame packet tracing.

LES.257

Level: P_TRACE

Short Syntax: LES.257 Trace LAN Emulation Data frame.

Long Syntax: LES.257 Trace LAN Emulation Data frame.

Description: LAN emulation data frame packet tracing.

LES.258

Level: UE_ERROR

Short Syntax: LES.258 LES/BUS:' *ELAN_name*': *redun_type*Rdndncy call fld:net down, Called ATM addr = x *called_address*

Long Syntax: LES.258 LES/BUS:' *ELAN_name*': *redun_type*Redundancy call failed:net down,Called ATM address = x *called_address*

Description: Redundancy call failed because connection to network is down

---

LES.259

Level: UE_ERROR

Short Syntax: LES.259 LES/BUS:' *ELAN_name*': *redun_type*Rdndncy VCC rlsd:cause *cause_code*

Long Syntax: LES.259 LES/BUS:' *ELAN_name*': *redun_type*Redundancy VCC released:cause *cause_code*

Description: Redundancy VCC was released

LES.260

Level: UE_ERROR

Short Syntax: LES.260 LES/BUS:' *ELAN_name*': *redun_type*Rdndncy VCC rlsd:net down

Long Syntax: LES.260 LES/BUS:' *ELAN_name*': *redun_type*Redundancy VCC released:net down

Description: Redundancy VCC released, connection to network is down

---

LES.261

Level: C_INFO

Short Syntax: LES.261 LES/BUS:' *ELAN_name*':plcng *redun_type*Rdndncy call Called ATM addr = x *called_address*

Long Syntax: LES.261 LES/BUS:' *ELAN_name*':placing *redun_type*Redundancy call, Called ATM address = x *called_address*

Description: Redundancy call was placed

LES.262

Level: UI_ERROR

Short Syntax: LES.262 LES/BUS:' *ELAN_name*':err plcng *redun_type*Rdndncy call: *error_string* (*error_code*) Called ATM addr = x *called_address*

Long Syntax: LES.262 LES/BUS:' *ELAN_name*':error placing *redun_type*Redundancy call: *error_string* (*error_code*), Called ATM address = x *called_address*

Description: An error occured while placing Redundancy call

LES.263

Level: UI_ERROR

Short Syntax: LES.263 LES/BUS:' *ELAN_name*':=>DOWN:err plcng *redun_type*Rdndncy call: *error_string* (*error_code*)

Long Syntax: LES.263 LES/BUS:' *ELAN_name*':=>DOWN:error placing *redun_type*Redundancy call: *error_string* (*error_code*)

Description: An error occured while placing Redundancy call, the ELAN will be terminated

---

LES.264

Level: UI_ERROR

Short Syntax: LES.264 LES/BUS:' *ELAN_name*':JOIN fld:frame buff alloc err LEC ATM addr = x *LEC_address*

Long Syntax: LES.264 LES/BUS:' *ELAN_name*':JOIN failed:frame buffer allocation error, LEC ATM address = x *LEC_address*

Description: Unable to allocate frame buffer, JOIN failed

Action: Contact your customer service representative

LES.265

Level: UI_ERROR

Short Syntax: LES.265 LES/BUS:' *ELAN_name*':=>DOWN:frm buff alloc err: *error_string* (*error_code*)

Long Syntax: LES.265 LES/BUS:' *ELAN_name*':=>DOWN:frame buffer allocation error: *error_string* (*error_code*)

Description: Unable to allocate frame buffer, ELAN will be terminated

Action: Contact your customer service representative

LES.266

Level: CE_ERROR

Short Syntax: LES.266 LES/BUS:' *ELAN_name*':JOIN fld:access denied LEC ATM addr = x *LEC_address*

Long Syntax: LES.266 LES/BUS:' *ELAN_name*':JOIN failed:access denied, LEC ATM address = x *LEC_address*

Description: JOIN validation failed, LEC is denied access to ELAN

LES.267

Level: UE_ERROR

Short Syntax: LES.267 LES/BUS:' *ELAN_name*':JOIN fld:LECS Intf err LEC ATM addr = x *LEC_address*

Long Syntax: LES.267 LES/BUS:' *ELAN_name*':JOIN failed:LECS Interface error, LEC ATM address = x *LEC_address*

Description: LECS Interface unable to send validation request to LECS

LES.268

Level: P_TRACE

Short Syntax: LES.268 Trace LECS Security Interface frame.

Long Syntax: LES.268 Trace LECS Security Intervace frame.

Description: LECS Security Interface frame packet tracing.

LES.269

Level: U_INFO

Short Syntax: LES.269 LECS Intf:dev *device_number*:STARTING

Long Syntax: LES.269 LECS Intf:dev *device_number*:STARTING

Description: LECS Interface was started

LES.270

Level: U_INFO

Short Syntax: LES.270 LECS Intf:dev *device_number*:DELETED

Long Syntax: LES.270 LECS Intf:dev *device_number*:DELETED

Description: LECS Interface was deleted

LES.271

Level: U_INFO

Short Syntax: LES.271 LECS Intf:dev *device_number*:RESTARTING

Long Syntax: LES.271 LECS Intf:dev *device_number*:RESTARTING

Description: LECS Interface was restarted

LES.272

Level: U_INFO

Short Syntax: LES.272 LECS Intf:dev *device_number*:STOPPED

Long Syntax: LES.272 LECS Intf:dev *device_number*:STOPPED

Description: LECS Interface was stopped

LES.273

Level: UI_ERROR

Short Syntax: LES.273 LECS Intf:dev *device_number*:=>DOWN:ATM user reg fld: *error_string* (*error_code*)

Long Syntax: LES.273 LECS Intf:dev *device_number*:=>DOWN:ATM user reg failed: *error_string* (*error_code*)

Description: ATM user registration failed, LECS Interface will be terminated

---

LES.274

Level: U_INFO

Short Syntax: LES.274 LECS Intf:dev *device_number*:wtng for ATM Net Up

Long Syntax: LES.274 LECS Interface:dev *device_number*:waiting for ATM NetUp

Description: LECS Interface is waiting for ATM interface to transition to up state

---

LES.275

Level: U_INFO

Short Syntax: LES.275 LECS Intf:dev *device_number*:wtng for ATM addr actvtn

Long Syntax: LES.275 LEC Intf:dev *device_number*:waiting for ATM address activation

Description: LECS Interace is waiting for ATM address activation to complete

---

LES.276

Level: UI_ERROR

Short Syntax: LES.276 LECS Intf:dev *device_number*:=>DOWN:ATM addr actvtn fld: *error_string* (*error_code*)

Long Syntax: LES.276 LECS Intf:dev *device_number*:=>DOWN:ATM address activation failed: *error_string* (*error_code*)

Description: ATM address activation failed, LECS Interface will be terminated

---

LES.277

Level: UI_ERROR

Short Syntax: LES.277 LECS Intf:dev *device_number*:=>DOWN:err reading ATM addr: *error_string* (*error_code*)

Long Syntax: LES.277 LECS Intf:dev *device_number*:=>DOWN:error reading ATM address: *error_string* (*error_code*)

Description: An error occurred while reading the ATM address, the LECS interface will be terminated

---

LES.278

Level: U_INFO

Short Syntax: LES.278 LECS Intf:dev *device_number*:wtng for UNI Vrsn rprt

Long Syntax: LES.278 LECS Intf:dev *device_number*:waiting for UNI Version report

Description: LECS Interface is waiting for the UNI Version Report

---

LES.279

Level: UI_ERROR

Short Syntax: LES.279 LECS Intf:dev *device_number*:=>DOWN:err reading UNI Vrsn: *error_string* (*error_code*)

Long Syntax: LES.279 LECS Intf:dev *device_number*:=>DOWN:error reading UNI Version Report: *error_string* (*error_code*)

Description: An error occurred while reading the UNI Version,the LECS Interface will be terminated

---

LES.280

Level: UI_ERROR

Short Syntax: LES.280 LECS Intf:dev *device_number*:=>DOWN:err opening ATM Adptr Frm SAP: *error_string* (*error_code*)

Long Syntax: LES.280 LECS Intf:dev *device_number*:=>DOWN:error opening ATM Adapter Frame SAP: *error_string* (*error_code*)

Description: An error occurred while opening the ATM Adapter Frame SAP, the LECS Interface will be terminated

---

LES.281

Level: UI_ERROR

Short Syntax: LES.281 LECS Intf:dev *device_number*:=>DOWN:err opening Call SAP: *error_string* (*error_code*)

Long Syntax: LES.281 LECS Intf:dev *device_number*:=>DOWN:error opening Call SAP: *error_string* (*error_code*)

Description: An error occurred while opening the Call SAP, the LECS Interface will be terminated

---

LES.282

Level: U_INFO

Short Syntax: LES.282 LECS Intf:dev *device_number*:wtng for LECS addr rprt

Long Syntax: LES.282 LECS Intf:dev *device_number*:waiting for LECS address report

Description: LECS Interface is waiting for list of LECS ATM address

---

LES.283

Level: UI_ERROR

Short Syntax: LES.283 LECS Intf:dev *device_number*:=>DOWN:err reading LECS addr: *error_string* (*error_code*)

Long Syntax: LES.283 LECS Intf:dev *device_number*:=>DOWN:error reading LECS address: *error_string* (*error_code*)

Description: An error occurred while reading the LECS address, the LECS Interface will be terminated

---

LES.284

Level: UE_ERROR

Short Syntax: LES.284 LECS Intf:dev *device_number*:ATM Net DOWN

Long Syntax: LES.284 LECS Intf: dev *device_number*:ATM Net DOWN

Description: The ATM interface is in an inoperable state, LECS Interface resources are released

---

LES.285

Level: U_INFO

Short Syntax: LES.285 LECS Intf:dev *device_number*:ATM Net UP

Long Syntax: LES.285 LECS Intf:dev *device_number*:ATM Net UP

Description: The ATM interface is in an operable state, the LECS Interface is restarted

LES.286

Level: U_INFO

Short Syntax: LES.286 LECS Intf:dev *device_number*:ATM addr actvtd

Long Syntax: LES.286 LECS Intf:dev *device_number*:ATM address activated

Description: The ATM address was successfully activated

LES.287

Level: UE_ERROR

Short Syntax: LES.287 LECS Intf:dev *device_number*: ATM addr actvtn timed out:retrying

Long Syntax: LES.287 LECS Intf:dev *device_number*: ATM address activation timed out: retrying

Description: ATM address activation request timed out, address activation will be retried

LES.288

Level: UE_ERROR

Short Syntax: LES.288 LECS Intf:dev *device_number*:ATM addr rjctd by switch

Long Syntax: LES.288 LECS Intf:dev *device_number*:ATM address rejected by switch

Description: ATM address rejected by switch. Another attempt will be made to activate the ATM address.

LES.289

Level: UE_ERROR

Short Syntax: LES.289 LECS Intf:dev *device_number*:ATM addr deactvtd:reactvtng

Long Syntax: LES.289 LECS Intf:dev *device_number*:ATM address deactivated:reactivating

Description: ATM address was deactived, address reactivation will be tried

LES.290

Level: U_INFO

Short Syntax: LES.290 LECS Intf:dev *device_number*:UNI Vrsn rprtd

Long Syntax: LES.290 LECS Intf:dev *device_number*:UNI Version reported

Description: The UNI Version was reported

LES.291

Level: U_INFO

Short Syntax: LES.291 LECS Intf:dev *device_number*:LECS addr list rprtd

Long Syntax: LES.291 LECS Intf:dev *device_number*:LECS address list reported

Description: The list of LECS ATM addresses was reported

LES.292

Level: CE_ERROR

Short Syntax: LES.292 LECS Intf:dev *device_number*:rfsd unexpctd call Calling ATM addr = x *calling_address*

Long Syntax: LES.292 LECS Intf:dev *device_number*:refused unexpected call, Calling ATM address = x *calling_address*

Description: An unexpected call was received, the call will be released

LES.293

Level: CE_ERROR

Short Syntax: LES.293 LECS Intf:dev *device_number*:Config Dir call fld:LECS negotiated parms LECS ATM addr = x *LECS_address*

Long Syntax: LES.293 LECS Intf:dev *device_number*:Config Dir call failed:LECS negotiated parms, LECS ATM address = x *LECS_address*

Description: AAL and BLLI parameters of LAN Emulation calls are not negotiable. LECS tried to negotiate these parms and the call failed.

LES.294

Level: UI_ERROR

Short Syntax: LES.294 LECS Intf:dev *device_number*:=>DOWN:Config Dir data path open err: *error_string* (*error_code*)

Long Syntax: LES.294 LECS Intf:dev *device_number*:=>DOWN:Config Direct data path open error: *error_string* (*error_code*)

Description: An error occurred when trying to open data path for VCC, LECS Interface will be terminated

---

LES.295

Level: UI_ERROR

Short Syntax: LES.295 LECS Intf:dev *device_number*:Config Dir call fld:data path open err:no mem

Long Syntax: LES.295 LECS Intf:dev *device_number*:Config Direct call failed:data path open error:no memory

Description: Insufficient resources to open data path for VCC

Action: Contact your customer service representative

LES.296

Level: C_INFO

Short Syntax: LES.296 LECS Intf:dev *device_number*:Config Dir estblshd LECS ATM addr = x *LECS_address*

Long Syntax: LES.296 LECS Intf:dev *device_number*:Config Dir esatblished, LECS ATM address = x *LECS_address*

Description: Configuration Direct VCC is operational

LES.297

Level: UI_ERROR

Short Syntax: LES.297 LECS Intf:unexpctd add leaf ack

Long Syntax: LES.297 LECS Intf:unexpected add leaf acknowledgement

Description: Unexpected add leaf acknowledgement was received

---

LES.298

Level: C_INFO

Short Syntax: LES.298 LECS Intf:dev *device_number*:Config Dir call fld:rtryng temp failure LECS ATM addr = x *LECS_address*

Long Syntax: LES.298 LECS Intf:dev *device_number*:Config Direct call failed:retrying temporary failure, LECS ATM address = x *LECS_address*

Description: Retry Config Direct call which failed due to a temporary condition

---

LES.299

Level: C_INFO

Short Syntax: LES.299 LECS Intf:dev *device_number*:Config Dir call fld:rtryng with Bearer Class C LECS ATM addr = x *LECS_address*

Long Syntax: LES.299 LECS Intf:dev *device_number*:Config Direct call failed:retrying with Bearer Class C, LECS ATM address = x *LECS_address*

Description: Control Direct call failed, retry with Bearer Class C

LES.300

Level: C_INFO

Short Syntax: LES.300 LECS Intf:dev *device_number*:Config Dir call fld:trying lower PCR (*PCR* Kbps) LECS ATM addr = x *LECS_address*

Long Syntax: LES.300 LECS Intf:dev *device_number*:Config Direct call failed:trying lower PCR (*PCR* Kbps), LECS ATM addr = x *LECS_address*

Description: Config Direct call failed because user cell rate is unavailable, call will be retired with a lower Peak Cell Rate

---

LES.301

Level: CE_ERROR

Short Syntax: LES.301 LECS Intf:dev *device_number*:Config Dir call fld:cause *cause_code* LECS ATM addr = x *LECS_address*

Long Syntax: LES.301 LECS Intf:dev *device_number*:Config Direct call failed:cause *cause_code*, LECS ATM address = x *LECS_address*

Description: A Config Direct call failed for the given reason

---

LES.302

Level: CE_ERROR

Short Syntax: LES.302 LECS Intf:dev *device_number*:Config Dir call fld:net down LECS ATM addr = x *LECS_address*

Long Syntax: LES.302 LECS Intf:dev *device_number*:Config Direct call failed:net down, LECS ATM address = x *LECS_address*

Description: Config Direct call failed, because the network is down

---

LES.303

Level: UI_ERROR

Short Syntax: LES.303 LECS Intf:unexpctd leaf rlse

Long Syntax: LES.303 LECS INTF:unexpected leaf release

Description: Leaf was released unexpectedly

LES.304

Level: C_INFO

Short Syntax: LES.304 LECS Intf:dev *device_number*:dscrded OAM frm

Long Syntax: LES.304 LECS Intf:dev *device_number*:discarded OAM frame

Description: An OAM frame was discarded

LES.305

Level: CE_ERROR

Short Syntax: LES.305 LECS Intf:dev *device_number*:dscrded frm:invld size (x *frame_size*)

Long Syntax: LES.305 LECS Intf:dev *device_number*:discarded frame:invalid size (x *frame_size*)

Description: Discarded frame because size was invalid

LES.306

Level: CE_ERROR

Short Syntax: LES.306 LECS Intf:dev *device_number*:dscrded frm:invld marker (x *marker*)

Long Syntax: LES.306 LECS Intf:dev *device_number*:discarded frame:invalid marker (x *marker*)

Description: Frame was discarded because marker was invalid. The marker should be xFF00

LES.307

Level: CE_ERROR

Short Syntax: LES.307 LECS Intf:dev *device_number*:dscrded frm:invld prtcl (x *protocol*)

Long Syntax: LES.307 LECS Intf:dev *device_number*:discarded frame:invalid protocol (x *protocol*)

Description: Frame was discarded because protocol was invalid. The protocol should be x01

LES.308

Level: CE_ERROR

Short Syntax: LES.308 LECS Intf:dev *device_number*:dscrded frm:invld vrsn (x *version*)

Long Syntax: LES.308 LECS Intf:dev *device_number*:discarded frame:invalid version (x *version*)

Description: Frame was discarded because the version was invalid. The version should be x01

LES.309

Level: CE_ERROR

Short Syntax: LES.309 LECS Intf:dev *device_number*:dscrded frm:invld opcode (x *opcode*)

Long Syntax: LES.309 LECS Intf:dev *device_number*:discarded frame:invalid opcode (x *opcode*)

Description: Frame was discarded because the opcode was invalid. The opcode should be x0101

LES.310

Level: CE_ERROR

Short Syntax: LES.310 LECS Intf:dev *device_number*:dscrded frm:invld number-TLVs (x *number_TLVs*)

Long Syntax: LES.310 LECS Intf:dev *device_number*:discarded frame:invalide number-TLVs (x *number_TLVs*)

Description: Frame was discarded because the number-TLVs field was invalid. Number-TLVs should be x01

LES.311

Level: CE_ERROR

Short Syntax: LES.311 LECS Intf:dev *device_number*:dscrded frm:invld TLV, Type = x *TLV_type*, Length = x *TLV_length*

Long Syntax: LES.311 LECS Intf:dev *device_number*:discarded frame:invalid TLV, Type = x *TLV_type*, Length = x *TLV_length*

Description: Frame was discarded because the TLV type or length were invalid

LES.312

Level: CE_ERROR

Short Syntax: LES.312 LECS Intf:dev *device_number*:dscrded frm:invld ELAN name size in TLV (x *ELAN_name_size*)

Long Syntax: LES.312 LECS Intf:dev *device_number*:discarded frame:invalid ELAN name size in TLV (x *ELAN_name_size*)

Description: Frame was discarded because ELAN name size was invalid

LES.313

Level: CE_ERROR

Short Syntax: LES.313 LECS Intf:dev *device_number*:dscrded frm:unknwn ELAN name in TLV, ELAN name = *ELAN_name*

Long Syntax: LES.313 LECS Intf:dev *device_number*:discarded frame:unknown ELAN name in TLV, ELAN name = x *ELAN_name*

Description: Frame was discarded because the ELAN name in the TLV value field is unknown

---

LES.314

Level: CE_ERROR

Short Syntax: LES.314 LECS Intf:dev *device_number*:dscrded frm:unknwn LEC ATM addr, ELAN name = *ELAN_name* LEC ATM addr = x *LEC_address*

Long Syntax: LES.314 LECS Intf:dev *device_number*:discarded frame:unknown LEC ATM address, ELAN name = x *ELAN_name*, LEC ATM address = x *LEC_address*

Description: Frame was discarded because the LEC ATM address was unknown

---

LES.315

Level: C_INFO

Short Syntax: LES.315 LECS Intf:dev *device_number*:plcng Config Dir call LECS ATM addr = x *LECS_address*

Long Syntax: LES.315 LECS Intf:dev *device_number*:placing Config Direct call, LECS ATM address = x *LECS_address*

Description: Call was placed to establish Config Direct VCC to LECS

LES.316

Level: UI_ERROR

Short Syntax: LES.316 LECS Intf:dev *device_number*:err plcng Config Dir call: *error_string* (*error_code*) LECS ATM addr = x *LECS_address*

Long Syntax: LES.316 LECS Intf:dev *device_number*:error placing Config Direct call: *error_string* (*error_code*), LECS ATM address = x *LECS_address*

Description: An error occurred while placing a call to establish a Config Direct VCC

LES.317

Level: UI_ERROR

Short Syntax: LES.317 LECS Intf:dev *device_number*:=>DOWN:err plcng Config Dir call: *error_string* (*error_code*)

Long Syntax: LES.317 LECS Intf:dev *device_number*:=>DOWN:error placing Config Direct call: *error_string* (*error_code*)

Description: An error occurred while placing a call to establish a Config Direct VCC, the LECS Interface will be terminated

---

LES.318

Level: UI_ERROR

Short Syntax: LES.318 LECS Intf:dev *device_number*:rlsng Config Dir:local LES err

Long Syntax: LES.318 LECS Intf:dev *device_number*:releasing Config Direct:local LES error

Description: Config Direct VCC was released due to a local LES error

---

LES.319

Level: UI_ERROR

Short Syntax: LES.319 LECS Intf:dev *device_number*:frm buff alloc err

Long Syntax: LES.319 LECS Intf:dev *device_number*:frame buffer allocation error

Description: Unable to allocate frame buffer

LES.320

Level: UI_ERROR

Short Syntax: LES.320 LECS Intf:dev *device_number*:=>DOWN:frm buff alloc err: *error_string* (*error_code*)

Long Syntax: LES.320 LECS Intf:dev *device_number*:=>DOWN:frame buffer allocation error: *error_string* (*error_code*)

Description: Unable to allocate frame buffer, LECS Interface will be terminated

---

LES.321

Level: UI_ERROR

Short Syntax: LES.321 LECS Intf:dev *device_number*:=>DOWN:tx err: *error_string* (*error_code*)

Long Syntax: LES.321 LECS Intf:dev *device_number*:=>DOWN:transmit error: *error_string* (*error_code*)

Description: An error occurred while transmitting

frame to LECS. Depending on the severity of the error, the LECS Interface may be terminated.

LES.322

Level: UI_ERROR

Short Syntax: LES.322 LECS Intf:dev *device_number*:trmntng: *error_string* (*error_code*)

Long Syntax: LES.322 LECS Intf:dev *device_number*:terminating: *error_string* (*error_code*)

Description: LECS Interface was terminated due to the given reason

---

LES.323

Level: UE_ERROR

Short Syntax: LES.323 LECS Intf:dev *device_number*:Config Dir rlsd:cause *cause_code* LECS ATM addr = x *LECS_address*

Long Syntax: LES.323 LECS Intf:dev *device_number*:Config Direct released:cause *cause_code*, LECS ATM address = x *LECS_address*

Description: Config Direct VCC was released due to the given reason

---

LES.324

Level: UE_ERROR

Short Syntax: LES.324 LECS Intf:dev *device_number*:Config Dir rlsd:net down LECS ATM addr = x *LECS_address*

Long Syntax: LES.324 LECS Intf:dev *device_number*:Config Direct released:net down, LECS ATM address = x *LECS_address*

Description: Config Dir released because network is down

LES.325

Level: U_INFO

Short Syntax: LES.325 BMON:' *ELAN_name*':initlzd

Long Syntax: LES.325 BMON:' *ELAN_name*':initialized

Description: BMON for this ELAN has been initialized

LES.326

Level: U_INFO

Short Syntax: LES.326 BMON:' *ELAN_name*':halted

Long Syntax: LES.326 BMON:' *ELAN_name*':halted

Description: BMON for this ELAN has been halted.

LES.327

Level: UI_ERROR

Short Syntax: LES.327 LES/BUS:' *ELAN_name*':BMON init fld

Long Syntax: LES.327 LES/BUS:' *ELAN_name*':BMON initialization failed

Description: BUS Monitor initialization failed due to lack of memory. ELAN operation continues.

Action: Contact your customer service representative

LES.328

Level: UI_ERROR

Short Syntax: LES.328 BMON:' *ELAN_name*':topN mem alloc fld

Long Syntax: LES.328 BMON:' *ELAN_name*':Top N memory allocation failed

Description: BUS Monitor could not allocate memory to record the Top N Users for the most recent sample interval. BUS Monitor will retry at the next sample interval.

Action: Contact your customer service representative

LES.329

Level: U_INFO

Short Syntax: LES.329 LES/BUS:' *ELAN_name*':ATM dev lnspeed - *VCC_type* VCC PCR (*peak_rate* Kbps) mismatch:PCR chngd to lnspeed (*linespeed* Kbps)

Long Syntax: LES.329 LES/BUS:' *ELAN_name*'ATM device linespeed - *VCC_type* VCC PCR (*peak_rate* Kbps) mismatch:PCR changed to linespeed (*linespeed* Kbps)

Description: The ATM device's linespeed has changed. The given VCC's PCR was equal to the ATM device's previous linespeed. The VCC's PCR has been changed and now equals the ATM device's current linespeed.

LES.330

Level: C_INFO

Short Syntax: LES.330 LES/BUS:' *ELAN_name*':ATM dev lnspeed is *linespeed* Kbps, *VCC_type* VCC PCR = *peak_rate* Kbps

Long Syntax: LES.330 LES/BUS:' *ELAN_name*'ATM device linespeed is *linespeed* Kbps, *VCC_type* VCC PCR = *peak_rate* Kbps

Description: The given VCC's peak rate is not equal to the ATM device's linespeed.

LES.331

Level: C_INFO

Short Syntax: LES.331 LES/BUS:' *ELAN_name*':ATM dev lnspeed is *linespeed* Kbps, *VCC_type* VCC PCR = *peak_rate* Kbps, SCR = *sustained_rate* Kbps

Long Syntax: LES.331 LES/BUS:' *ELAN_name*'ATM device linespeed is *linespeed* Kbps, *VCC_type* VCC PCR = *peak_rate* Kbps, SCR = *sustained_rate* Kbps

Description: The given VCC's peak rate is not equal to the ATM device's linespeed.

LES.332

Level: UI_ERROR

Short Syntax: LES.332 LES/BUS:' *ELAN_name*':Create fld: *VCC_type* VCC PCR (*peak_rate* Kbps) excds ATM dev lnspeed (*linespeed* Kbps)

Long Syntax: LES.332 LES/BUS:' *ELAN_name*':Create failed: *VCC_type* VCC PCR (*peak_rate* Kbps) exceeds ATM device linespeed (*linespeed* Kbps)

Description: The LES/BUS could not be created because the Peak Cell Rate exceeds the ATM device linespeed.

LES.333

Level: UI_ERROR

Short Syntax: LES.333 LES/BUS:' *ELAN_name*':Rstrt fld: *VCC_type* VCC PCR (*peak_rate* Kbps) excds ATM dev lnspeed (*linespeed* Kbps)

Long Syntax: LES.333 LES/BUS:' *ELAN_name*':Restart failed: *VCC_type* VCC PCR (*peak_rate* Kbps) exceeds ATM device linespeed (*linespeed* Kbps)

Description: The LES/BUS could not be restarted because the Peak Cell Rate exceeds the ATM device linespeed.

LES.334

Level: U_INFO

Short Syntax: LES.334 LECS Intf:dev *device_number*:ATM dev lnspeed - Config Dir VCC PCR (*peak_rate* Kbps) mismatch:PCR chngd to lnspeed (*linespeed* Kbps)

Long Syntax: LES.334 LECS Intf:dev *device_number*:ATM device linespeed - Config Direct VCC PCR (*peak_rate* Kbps) mismatch:PCR changed to linespeed (*linespeed* Kbps)

Description: The ATM device's linespeed has changed. The Config Direct VCC's PCR was equal to the ATM device's previous linespeed. The VCC's PCR has been changed and now equals the ATM device's current linespeed.

LES.335

Level: C_INFO

Short Syntax: LES.335 LECS Intf:dev *device_number*:ATM dev lnspeed is *linespeed* Kbps, Config Dir VCC PCR = *peak_rate* Kbps

Long Syntax: LES.335 LECS Intf:dev *device_number*:ATM device linespeed is *linespeed* Kbps, Config Direct VCC PCR = *peak_rate* Kbps

Description: The Config Direct VCC's peak rate is not equal to the ATM device's linespeed.

LES.336

Level: C_INFO

Short Syntax: LES.336 LECS Intf:dev *device_number*:ATM dev lnspeed is *linespeed* Kbps, Config Dir VCC PCR = *peak_rate* Kbps, SCR = *sustained_rate* Kbps

Long Syntax: LES.336 LECS Intf:dev *device_number*:ATM device linespeed is *linespeed* Kbps, Config Direct VCC PCR = *peak_rate* Kbps, SCR = *sustained_rate* Kbps

Description: The Config Direct VCC's peak rate is not equal to the ATM device's linespeed.

LES.337

Level: UI_ERROR

Short Syntax: LES.337 LECS Intf:dev *device_number*:Create fld:Config Dir VCC PCR (*peak_rate* Kbps) excds ATM dev lnspeed (*linespeed* Kbps)

Long Syntax: LES.337 LECS Intf:dev *device_number*:Create failed:Config Direct VCC PCR (*peak_rate* Kbps) exceeds ATM device linespeed (*linespeed* Kbps)

Description: The LECS Interface could not be created because the Peak Cell Rate exceeds the ATM device linespeed.

LES.338

Level: UI_ERROR

Short Syntax: LES.338 LECS Intf:dev *device_number*:Rstrt fld:Config Dir VCC PCR (*peak_rate* Kbps) excds ATM dev lnspeed (*linespeed* Kbps)

Long Syntax: LES.338 LECS Intf:dev *device_number*:Restart failed:Config Direct VCC PCR (*peak_rate* Kbps) exceeds ATM device linespeed (*linespeed* Kbps)

Description: The LECS Interface could not be restarted because the Peak Cell Rate exceeds the ATM device linespeed.

LES.339

Level: C_INFO

Short Syntax: LES.339 LES/BUS:' *ELAN_name*':updtd cnfgrtn for fld ' *field_name*'

Long Syntax: LES.339 LES/BUS:' *ELAN_name*':updated configuration for field ' *field_name*'

Description: During initialization, an outdated configuration record was discovered. Certain parameters in the configuration of the LES/BUS were updated to reflect new functional abilities. This event is common after moving to a new code release.

LES.340

Level: UI_ERROR

Short Syntax: LES.340 LES/BUS:' *ELAN_name*': *frame_type* fld, *reason*

Long Syntax: LES.340 LES/BUS:' *ELAN_name*': *frame_type* failed, *reason*

Description: A join or register request was rejected because of an error which occurred while processing the TLVs.

LES.341

Level: UI_ERROR

Short Syntax: LES.341 LES/BUS:' *ELAN_name*':rfsd Mcast Send VCC splice to Mcast Fwrd VCC, LEC ATM addr = x *LEC_address*

Long Syntax: LES.341 LES/BUS:' *ELAN_name*':refused Multicast Send VCC splice to Multicast Forward VCC, LEC ATM address = x *LEC_address*

Description: An error occured while attempting to splice the clients Multicast Send VCC to the BUS's Multicast Forward VCC.

Action: No immediate action is required. Peak BUS performance will not be possible for this client, but its participation in the specified ELAN is not effected. Contact customer service if further problem determination is needed.

LES.342

Level: UI_ERROR

Short Syntax: LES.342 LES/BUS:' *ELAN_name*':rfsd Mcast Send VCC unsplice from Mcast Fwrd VCC, LEC ATM addr = x *LEC_address*

Long Syntax: LES.342 LES/BUS:' *ELAN_name*':refused Multicast Send VCC unsplice from Multicast Forward VCC, LEC ATM address = x *LEC_address*

Description: An error occured while attempting to unsplice the clients Multicast Send VCC to the BUS's Multicast Forward VCC.

Action: No immediate action is required. Packets received from this client can not be traced. Contact customer service if further problem determination is needed.

LES.343

Level: UI_ERROR

Short Syntax: LES.343 LES/BUS:' *ELAN_name*':Incompatible hardware for VCC-splice operation, LEC ATM addr = x *LEC_address*

Long Syntax: LES.343 LES/BUS:' *ELAN_name*':Incompatible hardware for VCC-splice operation, LEC ATM address = x *LEC_address*

Description: The ATM Adapter hardware level installed does not support VCC splicing. The client's Multicast Send VCC has not been spliced to the BUS's Multicast Forward VCC.

Action: No immediate action is required. Peak BUS performance will not be possible for this client, but its participation in the specified ELAN is not effected. ATM Adapter may need to be upgraded to support VCC-splice feature. Contact customer service for futher assistance.

LES.344

Level: U_INFO

Short Syntax: LES.344 SUPER ELAN:Super ELAN spans multiple ATM interfaces, ID= *super_elan_id*.

Long Syntax: LES.344 SUPER ELAN:Super ELAN spans multiple ATM interfaces, ID= *super_elan_id*.

Description: Since each ATM Adapter may be connected to separate switched networks, attempts to establish data direct VCCs between clients in different ELANs may fail.

Action: If both ATM Adapters are connected to the same switch network, no action is required. If the ATM Adapters are connected to different switch networks, disable Super ELAN function on one or both interfaces, or assign logical interfaces on each ATM Adapter to different Super ELANs.

LES.345

Level: UI_ERROR

Short Syntax: LES.345 SUPER ELAN:Super ELAN is supported for STB enabled ports only, net *net_1*.

Long Syntax: LES.345 SUPER ELAN:Super ELAN is supported for STB enabled ports only, net *net_1*.

Description: Super ELAN is supported on bridge ports which are STB enabled. Both Token Ring and

Ethernet are supported, but bridge ports with Soure Route only bridging behavior are not supported.

Action: Enable Spanning Tree Transparent Bridge (STB) support on the bridge port associated with the specified interface.

LES.346

Level: UI_ERROR

Short Syntax: LES.346 SUPER ELAN:Super ELAN is supported on ATM interfaces only, net *net_1*.

Long Syntax: LES.346 SUPER ELAN:Super ELAN is supported on ATM interfaces only, net *net_1*.

Description: Super ELAN is supported on ATM interfaces only.

Action: Disable Super ELAN support on the bridge port associated with the specified interface.

LES.347

Level: UI_ERROR

Short Syntax: LES.347 SUPER ELAN:Intf types cannot be mixed within a Super ELAN, ID= *super_elan_id*.

Long Syntax: LES.347 SUPER ELAN:Interface types cannot be mixed within a Super ELAN, ID= *super_elan_id*.

Description: The Super ELAN ID could not be set because Token Ring and Ethernet clients cannot exist on the same Super ELAN.

Action: Change the Super ELAN ID to different value.

LES.348

Level: U_INFO

Short Syntax: LES.348 BCM:' *ELAN_name*':Warning: MAC addr x *MAC_address* replaced MAC addr x *MAC_address* for *protocol_type_string protocol_address*

Long Syntax: LES.348 BCM:' *ELAN_name*':Warning: MAC address x *MAC_address* replaced MAC address x *MAC_address* for *protocol_type_string protocol_address*

Description: BCM has discovered that two MAC addresses are using the same protocol address. The first MAC address displayed was detected more recently and will now be associated with the protocol address.

Action: This may be a misconfiguration of one of the devices.

LES.349

Level: U_INFO

Short Syntax: LES.349 LES/BUS:' *ELAN_name*':delay complete, add LEC to Ctrl Dist, LEC ATM addr = x *LEC_address*

Long Syntax: LES.349 LES/BUS:' *ELAN_name*':delay complete, now adding LEC to Control Distribute, LEC ATM address = x *LEC_address*

Description: addPartyDelayTimer has expired, so Add Leaf will now be attempted for this LEC on the Control Distribute VCC. This only occurs when switch signalling congestion is determined.

LES.350

Level: U_INFO

Short Syntax: LES.350 LES/BUS:' *ELAN_name*':delay complete, add LEC to Mcast Fwd, LEC ATM addr = x *LEC_address*

Long Syntax: LES.350 LES/BUS:' *ELAN_name*':delay complete, now adding LEC to Multicast Forward, LEC ATM address = x *LEC_address*

Description: addPartyDelayTimer has expired, so Add Leaf will now be attempted for this LEC on the Multicast Forward VCC. This only occurs when switch signalling congestion is determined.

LES.351

Level: U_INFO

Short Syntax: LES.351 LES/BUS:' *ELAN_name*':delay add of *VCC_type* leaf for *delay_duration* secs, LEC ATM addr = x *LEC_address*

Long Syntax: LES.351 LES/BUS:' *ELAN_name*':delaying addition of *VCC_type* leaf for *delay_duration* seconds, LEC ATM address = x *LEC_address*

Description: This leaf's addition to the LES Control Distribute VCC is being delayed until later due to switch signalling congestion found.

LES.352

Level: U_INFO

Short Syntax: LES.352 LES/BUS:' *ELAN_name*':delay add of *VCC_type* leaf for *delay_duration* secs, LEC ATM addr = x *LEC_address*

Long Syntax: LES.352 LES/BUS:' *ELAN_name*':delaying adding of *VCC_type* leaf for *delay_duration* seconds, LEC ATM address = x *LEC_address*

Description: This leaf's addition to the BUS Multicast Forward VCC is being delayed until later due to switch signalling congestion found.

LES.353

Level: U_INFO

Short Syntax: LES.353 LES/BUS:' *ELAN_name*' Temp err adding *VCC_type* leaf: cause # *cause_code*: retry later LEC ATM addr = x *LEC_address*

Long Syntax: LES.353 LES/BUS:' *ELAN_name*':temporary error adding *VCC_type* leaf: cause code # *cause_code*: will retry later, LEC ATM address = x *LEC_address*

Description: Add leaf request failed due to a temporary condition, the add leaf request will be retried after random delay

---

LES.354

Level: U_INFO

Short Syntax: LES.354 LES/BUS:' *ELAN_name*': err adding *VCC_type* leaf: out of mem, LEC ATM addr = x *LEC_address*

Long Syntax: LES.354 LES/BUS:' *ELAN_name*': error adding *VCC_type* leaf: memory exhausted, LEC ATM address = x *LEC_address*

Description: Unable to add leaf, because out of memory

LES.355

Level: U_INFO

Short Syntax: LES.355 LES/BUS:' *ELAN_name*' Terminating LEC: err adding *VCC_type* leaf: no memory, LEC ATM addr = x *LEC_address*

Long Syntax: LES.355 LES/BUS:' *ELAN_name*': terminating LEC: error adding *VCC_type* leaf: memory exhausted, LEC AT address = x *LEC_address*

Description: A leaf was not added, because there is no available memory. The LEC's ELAN membership will be terminated

---

LES.356

Level: U_INFO

Short Syntax: LES.356 Interface # *interface_number*: Entering Add Party Delay State

Long Syntax: LES.356 Interface # *interface_number*: Now entering the Add Party Delay state

Description: Network signalling congestion was detected (as evidenced by Add Party being rejected with a temporary cause or ignored) and we are now entering the state where all LES/BUSs on this interface will randomly delay sending Add Party messages)

LES.357

Level: U_INFO

Short Syntax: LES.357 LES/BUS: ' *ELAN_name*' *VCC_type* leaf: drop LEC: max Add Leaf retries LEC ATM addr = x *LEC_address*

Long Syntax: LES.357 LES_BUS: ' *ELAN_name*': terminating LEC: *VCC_type* leaf: no more Add Leaf retries, LEC ATM addr = x *LEC_address*

Description: Another Add Party message needs to be retransmitted due to network signalling congestion, but we have exceeded the max number of retries, so clear the Multicast Send to the LEC.

LES.358

Level: U_INFO

Short Syntax: LES.358 Interface # *interface_number*: Leaving Add Party Delay State

Long Syntax: LES.358 Interface # *interface_number*: Now leaving the Add Party Delay state

Description: No network signalling congestion has been detected by any LES/BUS in the last DELAY_ADD_PARTY_STATE_CLEARING_INTERVAL seconds, and we were in the Delay Add Party state, so leave the Delay Add Party State.

LES.359

Level: U_INFO

Short Syntax: LES.359 LES/BUS:' *ELAN_name*' Temp err adding *VCC_type* leaf: network down: retry later LEC ATM addr = x *LEC_address*

Long Syntax: LES.359 LES/BUS:' *ELAN_name*':temporary error adding *VCC_type* leaf: network down: will retry later, LEC ATM address = x *LEC_address*

Description: Add leaf request failed due to a NETWORK_DOWN condition, the add leaf request will be retried after random delay. This usually means either SAAL is down or the Add Party was not responded to.

LES.360

Level: CE_ERROR

Short Syntax: LES.360 LES/BUS:' *ELAN_name*':0-hop rings overlap with Rtr mac= *MAC_address* (*ring_number*: *ring_number*) Req Rtr mac= *MAC_address* (*ring_number*: *ring_number*)

Long Syntax: LES.360 LES/BUS:' *ELAN_name*':0-hop rings overlap with Router MAC address= *MAC_address* (range= *ring_number*: *ring_number*) Requesting Router MAC address= *MAC_address*(range= *ring_number*: *ring_number*)

Description: A zeroHop router tried to register an overlapping virtual ring range

---

LES.361

Level: UI_ERROR

Short Syntax: LES.361 LES/BUS:' *ELAN_name*':refuse Mcast Send call: no mem, LEC ATM addr = x *LEC_address*

Long Syntax: LES.361 LES/BUS:' *ELAN_name*':refused Multicast Send call: no memory, LEC ATM address = x *LEC_address*

Description: Unable to accept Multicast Send Call due to lack of memory needed for adding the joining internal LEC's bound Mcast Send VCC to the corresponding LES/BUS queue.

LES.362

Level: U_INFO

Short Syntax: LES.362 BCM:' *ELAN_name*':NetBIOS NAME_RECOGNIZED not rcvd, deleting Name *protocol_address* from MAC addr x *MAC_address*

Long Syntax: LES.362 BCM:' *ELAN_name*':NetBIOS NAME_RECOGNIZED not received, deleting Name *protocol_address* from MAC addr x *MAC_address*

Description: BCM NetBIOS has not received a NAME_RECOGNIZED within 1 second of directing a NAME_QUERY to the given Name and MAC address. The BCM NetBIOS cache entry may no longer be valid, so the entry is being deleted. The NAME_QUERY currently being processed by BCM will be broadcast. If the Namesharing feature is in use for the given Name, it is possible all sessions using the given MAC address have been exhausted, and this event is normal.

LES.363

Level: C_INFO

Short Syntax: LES.363 LECS Intf:dev *device_num*:updtd cnfgrtn for fld ' *field_name*'

Long Syntax: LES.363 LECS Intf:dev *device_num*:updated configuration for field ' *field_name*'

Description: During initialization, an outdated configuration record was discovered. Certain parameters in the configuration of the LECS Interface were updated to reflect new functional abilities. This event is common after moving to a new code release.

LES.364

Level: CE_ERROR

Short Syntax: LES.364 LES/BUS:' *ELAN_name*':JOIN fld:invld flags (x *flags*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.364 LES/BUS:' *ELAN_name*':JOIN failed:invalid flags (x *flags*), LEC ATM address = x *LEC_address*

Description: JOIN failed due to invalid flags field. This may be a result of a LEC which is not a LUNIv2 LEC setting LUNIv2 flags.

LES.365

Level: CE_ERROR

Short Syntax: LES.365 LES/BUS:' *ELAN_name*':JOIN fld:dplct RD (x *route_descriptor*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.365 LES/BUS:' *ELAN_name*':JOIN failed:duplicate route descriptor (x *route_descriptor*), LEC ATM address = x *LEC_address*

Description: JOIN failed, because route descriptor address was not unique.

LES.366

Level: UI_ERROR

Short Syntax: LES.366 LES/BUS:' *ELAN_name*':JOIN fld:RD CB alloc err, LEC ATM addr = x *LEC_address*

Long Syntax: LES.366 LES/BUS:' *ELAN_name*':JOIN failed:Route Descriptor Control Block allocation error, LEC ATM address =x *LEC_address*

Description: JOIN failed, because an error occurred while trying to allocate memory for the route descriptor control block.

Action: Contact your customer service representative

LES.367

Level: CE_ERROR

Short Syntax: LES.367 LES/BUS:' *ELAN_name*':dscrd VRFY REQ:JOIN incmplt, LEC ATM addr = x *LEC_address*

Long Syntax: LES.367 LES/BUS:' *ELAN_name*':discarded Verify Request:JOIN incomplete, LEC ATM addr = x *LEC_address*

Description: Verify Request was discarded, because the JOIN phase has not completed.

LES.368

Level: CE_ERROR

Short Syntax: LES.368 LES/BUS:' *ELAN_name*':VRFY fld:invld LECID (*LECID*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.368 LES/BUS:' *ELAN_name*':Verify failed:invalid LECID (*LECID*), LEC ATM addr = x *LEC_address*

Description: Verify failed, because the LECID is invalid.

LES.369

Level: C_INFO

Short Syntax: LES.369 LES/BUS:' *ELAN_name*':VRFY fld:invld ATM addr (x *atm_addr_to_verify*), LEC ATM addr = x *LEC_address*

Long Syntax: LES.369 LES/BUS:' *ELAN_name*':Verify failed:invalid ATM address (x *atm_addr_to_verify*), LEC ATM addr = x *LEC_address*

Description: Verify failed, because the ATM address is not the ATM address of a BUS.

LES.370

Level: C_INFO

Short Syntax: LES.370 LES/BUS:' *ELAN_name*':VRFY ok for ATM addr x *atm_addr_to_verify*, LEC ATM addr = x *LEC_address*

Long Syntax: LES.370 LES/BUS:' *ELAN_name*':Verify ok for ATM address x *atm_addr_to_verify*, LEC ATM addr = x *LEC_address*

Description: Verify succeeded. The specified ATM address was indead an ATM address of the BUS.

LES.371

Level: CE_ERROR

Short Syntax: LES.371 LES/BUS:' *ELAN_name*':trmntng LEC:Mcast Send dscnnctd time-out, LEC ATM addr = x *LEC_address*

Long Syntax: LES.371 LES/BUS:' *ELAN_name*':terminating LEC:Mcast Send disconnected time-out, LEC ATM address = x *LEC_address*

Description: LEC has been terminated because there has been no Multicast Send VCC from the LEC to the BUS for the Multicast Send Disconnect Time.

LES.372

Level: CE_ERROR

Short Syntax: LES.372 LES/BUS:' *ELAN_name*':dscrd data frm:bad RIF lngth x *rif_length*, Src LEC ATM addr = x *source_LEC_address*,

Long Syntax: LES.372 LES/BUS:' *ELAN_name*':discarded data frame:invalid RIF length x *rif_length*, Source LEC ATM address = x *source_LEC_address*

Description: Data frame was discarded because RIF length was invalid.

LES.373

Level: CE_ERROR

Short Syntax: LES.373 LES/BUS:' *ELAN_name*':dscrd FLUSH REQ:trgt msmtch, Src LEC ATM addr = x *source_LEC_address*, Trgt LEC ATM addr = x *target_LEC_address*, Trgt MAC addr = x *target_MAC*

Long Syntax: LES.373 LES/BUS:' *ELAN_name*':discarded Flush request:target mismatch, Source LEC ATM address = x *source_LEC_address*, Target LEC ATM addr = x *target_LEC_address* Target MAC address = x *target_MAC*

Description: Flush Request discarded because target MAC and target ATM address indicated different LECs.

LES.374

Level: CE_ERROR

Short Syntax: LES.374 LES/BUS:' *ELAN_name*':dscrd FLUSH REQ:trgt msmtch, Src LEC ATM addr = x *source_LEC_address*, Trgt LEC ATM addr = x *target_LEC_address*, Trgt RD = x *target_RD*

Long Syntax: LES.374 LES/BUS:' *ELAN_name*':discarded Flush request:target mismatch, Source LEC ATM address = x *source_LEC_address*, Target LEC ATM addr = x *target_LEC_address* Target RD = x *target_RD*

Description: Flush Request discarded because target RD and target ATM address indicated different LECs.

LES.375

Level: CE_ERROR

Short Syntax: LES.375 LES/BUS:' *ELAN_name*':dscrd FLUSH REQ:invld tag (x *lan_dest_tag*), Src LEC ATM addr = x *source_LEC_address*, Trgt LEC ATM addr = x *target_LEC_address*,

Long Syntax: LES.375 LES/BUS:' *ELAN_name*':discarded Flush request:invalid tag (x *lan_dest_tag*), Source LEC ATM address = x *source_LEC_address*, Target LEC ATM addr = x *target_LEC_address*

Description: Flush Request discarded because target LAN destination field was not valid. Must be 1 or 2 for LANEv2 LE Clients.

LES.376

Level: UI_ERROR

Short Syntax: LES.376 LECS Intf:dev *device_number*:mem alloc err: src ATM addrx *lec_atm_addr*

Long Syntax: LES.376 LECS Intf:dev *device_number*:memory allocation error: source ATM address x *lec_atm_addr*

Description: Validation of security request failed because the LECS Interface was unable to allocate required memory.

LES.377

Level: UE_ERROR

Short Syntax: LES.377 LECS Intf:dev *device_number*:cpy TLVs for scrty fld: src ATM addrx *lec_atm_addr*

Long Syntax: LES.377 LECS Intf:dev *device_number*:copy TLVs for security request failed: source ATM address x *lec_atm_addr*

Description: LECS Interface was unable to append join TLVs to security request. Either TLVs in join were corrupted, or the addition of the security TLV creates a frame that is too large to send to a LECS.

LES.378

Level: U_INFO

Short Syntax: LES.378 BCM:' *ELAN_name*':added IPX Server Farm (reached *ipx_threshold* Srvrs/Rtrs), LEC ATM addr = x *LEC_address*

Long Syntax: LES.378 BCM:' *ELAN_name*':added IPX Server Farm (reached *ipx_threshold* Servers/Routers), LEC ATM address = x *LEC_address*

Description: BCM IPX has detected a Server Farm behind the given LEC. The number of dynamically discovered IPX Servers/Routers behind the LEC has reached the given Server Farm Threshold.

LES.379

Level: U_INFO

Short Syntax: LES.379 BCM:' *ELAN_name*':removed IPX Server Farm (less than *ipx_threshold* Srvrs/Rtrs), LEC ATM addr = x *LEC_address*

Long Syntax: LES.379 BCM:' *ELAN_name*':removed IPX Server Farm (less than *ipx_threshold* Servers/Routers), LEC ATM address = x *LEC_address*

Description: BCM IPX has removed a previously detected Server Farm behind the given LEC. The number of dynamically discovered IPX Servers/Routers behind the LEC has dropped below the given Server Farm Threshold.

LES.380

Level: UI_ERROR

Short Syntax: LES.380 LES/BUS:Receive frame bad vcc correlator 0x *vcc_corr*, atm user 0x *atm_user_corr*

Long Syntax: LES.380 LES/BUS:Receive frame bad vcc correlator 0x *vcc_corr*, atm user 0x *atm_user_corr*

Description: A frame was received on a connection type which is not supported. Received frame was discarded.

LES.381

Level: U_INFO

Short Syntax: LES.381 LES/BUS:' *ELAN_name*':v2 Lec->v1 Lec:frame buff alloc err LEC ATM addr = x *LEC_address*

Long Syntax: LES.381 LES/BUS:' *ELAN_name*':v2 Lec->v1 Lec:frame buffer allocation error, LEC ATM address = x *LEC_address*

Description: Unable to allocate frame buffer, LUNI v2 control frame could not be converted for transmission to v1 LECs. Frame transmitted in LUNI v2 format to v1 LECs.

Action: Contact your customer service representative

LES.382

Level: C_INFO

Short Syntax: LES.382 LES/BUS:' *ELAN_name*':dscrd *frameType* frm:no V2 Proxy Ctrl Dist, Src LEC ATM addr = x *source_LEC_address*,

Long Syntax: LES.382 LES/BUS:' *ELAN_name*':discarded *frameType* frame:no V2 Proxy Control Distribute, Source LEC ATM address = x *source_LEC_address*

Description: A frame of the specified type was discarded. It was to be forwarded over the V2 Proxy Control Distribute VCC, but the V2 Proxy Control Distribute VCC is not operational. This is most likely caused by no proxy clients joining the ELAN.

LES.383

Level: C_INFO

Short Syntax: LES.383 LES/BUS:' *ELAN_name*':dscrd *frameType* frm:no V1 or V2 Proxy Ctrl Dist, Src LEC ATM addr = x *source_LEC_address*,

Long Syntax: LES.383 LES/BUS:' *ELAN_name*':discarded *frameType* frame:no V1 or V2 Proxy Control Distribute, Source LEC ATM address = x *source_LEC_address*

Description: A frame of the specified type was discarded. It was to be forwarded over the Proxy Control Distribute VCC, but neither the V1 Proxy Control Distribute VCC nor the V2 Proxy Control Distribute VCC is operational. This is most likely caused by no proxy clients joining the ELAN.

LES.384

Level: UI_ERROR

Short Syntax: LES.384 LES/BUS:' *ELAN_name*':Unable to allocate memory for configuration

Long Syntax: LES.384 LES/BUS:' *ELAN_name*':Unable to allocate memory for configuration

Description: Memory could not be allocated for the purpose of migrating or creating the LES/BUS configuration. Use config tool to migrate configuration. This config is probably invalid and cannot be used.

LES.385

Level: U_INFO

Short Syntax: LES.385 LES/BUS:' *ELAN_name*':yielding to Partner, # LEC's terminated= *num_lecs reason* Partner Atm addr = x *les_address*

Long Syntax: LES.385 LES/BUS:' *ELAN_name*':yielding to Partner, # LEC's terminated= *num_lecs*. *reason*, Partner Atm addr = x *les_address*

Description: LES/BUS is terminating all LEC's because it is yielding to the Partner LES/BUS for the given reason.

LES.386

Level: U_INFO

Short Syntax: LES.386 LES/BUS:' *ELAN_name*':now accptng LEC's, taking over Partner *reason* Partner Atm addr = x *les_address*

Long Syntax: LES.386 LES/BUS:' *ELAN_name*':now accepting LEC's, taking over Partner *reason*, Partner Atm addr = x *les_address*

Description: LES/BUS is now accepting new LE Clients because it is taking over from the Partner LES/BUS for the given reason.

LES.387

Level: UI_ERROR

Short Syntax: LES.387 LES/BUS:' *ELAN_name*':=>DOWN:Enhncd Rdndncy Call dt pth opn err: *error_string* (*error_code*)

Long Syntax: LES.387 LES/BUS:' *ELAN_name*':=>DOWN:Enhanced Redundancy Call data path open error: *error_string* (*error_code*)

Description: An error occurred when trying to open data path for Enhanced Redundancy VCC, the ELAN will be terminated

---

LES.388

Level: UI_ERROR

Short Syntax: LES.388 LES/BUS:' *ELAN_name*':rlsd Enhncd Rdndncy Call:dt pth opn err:no mem

Long Syntax: LES.388 LES/BUS:' *ELAN_name*':released Enhanced Redundancy Call:data path open error:no memory

Description: Insufficient resources to open data path for Enhanced Redundancy VCC. The Enhanced Redundancy VCC will be released and re-tried later.

Action: Contact your customer service representative

LES.389

Level: P_TRACE

Short Syntax: LES.389 Trace LES/BUS Enhncd Rdndncy Status Message.

Long Syntax: LES.389 Trace LES/BUS Enhanced Redundancy Status Message.

Description: LES/BUS Enhanced Redundancy Status Message packet tracing.

LES.390

Level: CE_ERROR

Short Syntax: LES.390 LES/BUS:' *ELAN_name*':dscrd rdndcy status msg:invld sz (x *frame_size*)

Long Syntax: LES.390 LES/BUS:' *ELAN_name*':discarded redundancy status message:invalid size (x *frame_size*)

Description: A redundancy status message sent to the LES was discarded because the actual size was invalid.

LES.391

Level: CE_ERROR

Short Syntax: LES.391 LES/BUS:' *ELAN_name*':dscrd rdndcy status msg:invld mrkr (x *marker*)

Long Syntax: LES.391 LES/BUS:' *ELAN_name*':discarded redundancy status message:invalid Marker (x *marker*)

Description: A redundancy status message was discarded, because the Marker was invalid. The Marker should be xFF00

LES.392

Level: CE_ERROR

Short Syntax: LES.392 LES/BUS:' *ELAN_name*':dscrd rdndcy status msg:invld prtcl (x *protocol*)

Long Syntax: LES.392 LES/BUS:' *ELAN_name*':discarded redundancy status

message:invalid prtcl (x *protocol*)

Description: A redundancy status message was discarded, because the protocol was invalid. The protocol should be x01

LES.393

Level: CE_ERROR

Short Syntax: LES.393 LES/BUS:' *ELAN_name*':dscrd rdndcy status msg:invld Vrsn (x *version*)

Long Syntax: LES.393 LES/BUS:' *ELAN_name*':discarded redundancy status message:invalid Version (x *version*)

Description: A redundancy status message was discarded, because the Version is invalid. The version should be x01

LES.394

Level: CE_ERROR

Short Syntax: LES.394 LES/BUS:' *ELAN_name*':dscrd rdndcy status msg:invld opcode (x *opcode*)

Long Syntax: LES.394 LES/BUS:' *ELAN_name*':discarded redundancy status message:invalid opcode (x *opcode*)

Description: A redundancy status message was discarded, because the opcode is invalid.

LES.395

Level: U_INFO

Short Syntax: LES.395 LES/BUS:' *ELAN_name*':rdndcy status msg frame buff alloc err

Long Syntax: LES.395 LES/BUS:' *ELAN_name*':redundancy status message frame buffer allocation error

Description: Unable to allocate frame buffer, redundancy status message frame could not be sent to partner LES/BUS.

LES.396

Level: U_INFO

Short Syntax: LES.396 LES/BUS:' *ELAN_name*':takeover req. sent to backup Partner Atm addr = x *les_address*

Long Syntax: LES.396 LES/BUS:' *ELAN_name*':takeover request sent to backup. Partner Atm addr = x *les_address*

Description: User has requested this LES/BUS takeover from the active backup LES/BUS. The primary LES/BUS will not begin accepting new LE Clients until the backup LES/BUS yields.

LES.397

Level: C_INFO

Short Syntax: LES.397 BUSFILTER:' *ELAN_name*':initlzd

Long Syntax: LES.397 BUSFILTER:' *ELAN_name*':initialized

Description: Bus Filter for this Elan has been initialized.

LES.398

Level: U_INFO

Short Syntax: LES.398 BUSPOLICE:' *ELAN_name*':initlzd

Long Syntax: LES.398 BUSPOLICE:' *ELAN_name*':initialized

Description: Bus Police for this Elan has been initialized.

LES.399

Level: UI_ERROR

Short Syntax: LES.399 BUSFILTER:' *ELAN_name*':init fld

Long Syntax: LES.399 BUSFILTER:' *ELAN_name*':initialization failed

Description: BUS Filter initialization failed; memory allocation error. ELAN operation continues.

Action: Contact your customer service representative

LES.400

Level: UI_ERROR

Short Syntax: LES.400 BUSPOLICE:' *ELAN_name*':init fld

Long Syntax: LES.400 BUSPOLICE:' *ELAN_name*':initialization failed

Description: BUS Police initialization failed due to lack of memory. ELAN operation continues.

Action: Contact your customer service representative

LES.401

Level: U_INFO

Short Syntax: LES.401 BUSFILTER:' *ELAN_name*':Rmv'd filter items

Long Syntax: LES.401 BUSFILTER:' *ELAN_name*':Removed all filter items

Description: All Filter items for this Bus Filter have been removed and associated memory has been freed.

LES.402

Level: U_INFO

Short Syntax: LES.402 BUSPOLICE:' *ELAN_name*':Rmv'd filter items

Long Syntax: LES.402 BUSPOLICE:' *ELAN_name*':Removed all filter items

Description: All Filter items for this Bus Police have been removed and associated memory has been freed.

LES.403

Level: UE_ERROR

Short Syntax: LES.403 BUSPOLICE:' *ELAN_name*':threshold exc'd Src mac addr = *SOURCE_address*

Long Syntax: LES.403 BUSPOLICE:' *ELAN_name*':Threshold has been exceeded. SOURCE mac address = x *SOURCE_address*

Description: The defined BUS POLICE threshold has been exceeded.

LES.404

Level: U_INFO

Short Syntax: LES.404 BUSPOLICE:' *ELAN_name*':Not added to Filter List. On immunity list Src mac addr = *SOURCE_address*

Long Syntax: LES.404 BUSPOLICE:' *ELAN_name*':Mac address not added to Filter list. Already on immunity list. SOURCE mac address = x *SOURCE_address*

Description: The Mac Address was not added to the Filter List because it is on the Immunity List.

LES.405

Level: UI_ERROR

Short Syntax: LES.405 BUSPOLICE:' *ELAN_name*':Not put on Filter List. Mem alloc error. src mac addr = x *SOURCE_address*

Long Syntax: LES.405 BUSPOLICE:' *ELAN_name*':Address was not put on Filter List. Mem Allocation error. SOURCE mac address = x *SOURCE_address*

Description: The Mac address was not put on the Filter list. Not enough memory was available.

Action: Contact your customer service representative

LES.406

Level: U_INFO

Short Syntax: LES.406 BUSPOLICE:' *ELAN_name*':Added to Filter List. Src mac addr = *SOURCE_address*

Long Syntax: LES.406 BUSPOLICE:' *ELAN_name*':Mac address was added to Filter list. SOURCE mac address = x *SOURCE_address*

Description: The Mac Address was successfully added to the Filter List.

LES.407

Level: C_INFO

Short Syntax: LES.407 BUSFILTER:' *ELAN_name*':Match Exclude List. cmp value = 0x *Item_string*

Long Syntax: LES.407 BUSFILTER:' *ELAN_name*':Frame matched filter on Exclude list. Compare Value = 0x *Item_string*.

Description: The Frame match a filter on Exclude list of Bus Filter. The frame was filtered.

LES.408

Level: C_INFO

Short Syntax: LES.408 BUSFILTER:' *ELAN_name*':Match Include List. cmp value = 0x *Item_string*

Long Syntax: LES.408 BUSFILTER:' *ELAN_name*':Frame matched filter on Include list. Compare Value = 0x *Item_string*.

Description: The Frame match a filter on Include list of Bus Filter. The frame was forwarded.

LES.409

Level: C_INFO

Short Syntax: LES.409 BUSFILTER:' *ELAN_name*':No Match. Dflt performed-EXCLUDE.

Long Syntax: LES.409 BUSFILTER:' *ELAN_name*':No Match on filter. Default action was performed. Frame was filtered.

Description: The Frame did not match a filter on either filter list. The default action of EXCLUDE was performed. The frame was filtered.

LES.410

Level: C_INFO

Short Syntax: LES.410 BUSFILTER:' *ELAN_name*':No Match. Dflt performed-INCLUDE.

Long Syntax: LES.410 BUSFILTER:' *ELAN_name*':No Match on filter. Default action was performed. Frame was forwarded.

Description: The Frame did not match a filter on

either filter list. The default action of INCLUDE was performed. The frame was forwarded.

LES.411

Level: C_INFO

Short Syntax: LES.411 BUSPOLICE:' *ELAN_name*':Frame filtered. Src mac addr = *SOURCE_address*

Long Syntax: LES.411 BUSPOLICE:' *ELAN_name*':Frame was filtered by Bus Police. Source mac address = 0x *SOURCE_address*

Description: Frame from this mac address was filtered by Bus Police.

Chapter 66. LAN Network Manager (LNM)

This chapter describes LAN Network Manager (LNM) messages. For information on message content and how to use the message, refer to the Introduction.

LNM.001

Level: C-INFO

Short Syntax: LNM.001 Configuring port *port_number*

Long Syntax: LNM.001 Configuring port *port_number*

Description: LNM is beginning Configuration of the specified port.

LNM.002

Level: C-INFO

Short Syntax: LNM.002 Configuration complete port *port_number* nt *network*

Long Syntax: LNM.002 Configuration complete port *port_number* network *network*

Description: LNM has completed the Configuration of the specified port.

LNM.003

Level: U-INFO

Short Syntax: LNM.003 LNM configured for port *port_number*, port does not exist in Bridge Configuration

Long Syntax: LNM.003 LNM configured for port *port_number*, but the port is not configured in the Bridge Configuration

Description: The port is configured in the LNM configuration, but not in the SRT configuration.

Cause: User configuration error.

Action: Reconfigure LNM or SRT. Ensure Bridge is enabled.

LNM.004

Level: U-INFO

Short Syntax: LNM.004 LNM configured for port *port_number*, is not SRB port

Long Syntax: LNM.004 LNM configured for port *port_number*, is not configured for SRB

Description: The port is configured in the LNM configuration, but is not configured as an SRB port in the SRT configuration.

Cause: User configuration error.

Action: Reconfigure LNM or SRT.

LNM.005

Level: U-INFO

Short Syntax: LNM.005 LNM configured for port *port_number*, is not token ring

Long Syntax: LNM.005 LNM configured for port *port_number*, is not a token ring interface

Description: The port is configured in the LNM configuration, but the interface is not a Token-Ring interface.

Cause: User configuration error.

Action: Reconfigure LNM or the interface.

LNM.006

Level: UI-ERROR

Short Syntax: LNM.006 No iorb to transmit packet

Long Syntax: LNM.006 No buffer available to copy one or more packets

Description: No buffer available to copy one or more packets in order to send through LLC.

Cause: Severe packet buffer shortage.

Action: Check memory statistics in GWCON to verify packet buffer level.

Cause: Traffic peak using all available buffers.

Action: This is the problem if this message occurs very infrequently.

LNM.007

Level: C-INFO

Short Syntax: LNM.007 Initializing port *port_number* nt *network*

Long Syntax: LNM.007 Initializing port *port_number* network *network*

Description: LNM is beginning initialization of the specified port.

LNM.008

Level: C-INFO

Short Syntax: LNM.008 Initialization complete port *port_number* nt *network*

Long Syntax: LNM.008 Initialization complete port *port_number* network *network*

Description: LNM has completed the initialization of the specified port.

LNM.009

Level: C-INFO

Short Syntax: LNM.009 Activating LLC for port *port_number* nt *network*

Long Syntax: LNM.009 Activating LLC for port *port_number* network *network*

Description: LNM is activating the connection to LLC for the specified port.

LNM.010

Level: C-INFO

Short Syntax: LNM.010 Activating REM for port *port_number* nt *network*

Long Syntax: LNM.010 Activating REM for port *port_number* network *network*

Description: LNM is activating the Ring Error Monitor on the specified port.

LNM.011

Level: C-INFO

Short Syntax: LNM.011 Activating RPS for port *port_number* nt *network*

Long Syntax: LNM.011 Activating RPS for port *port_number* network *network*

Description: LNM is activating the Ring Parameter Server on the specified port.

LNM.012

Level: C-INFO

Short Syntax: LNM.012 Activating CRS for port *port_number* nt *network*

Long Syntax: LNM.012 Activating CRS for port *port_number* network *network*

Description: LNM is activating the Configuration Report Server for the specified port.

LNM.013

Level: C-INFO

Short Syntax: LNM.013 Activating LRM for port *port_number* nt *network*

Long Syntax: LNM.013 Activating LRM for port *port_number* network *network*

Description: LNM is activating the LAN Reporting Mechanism for the specified port.

LNM.014

Level: C-INFO

Short Syntax: LNM.014 Activating MAC frame int for port *port_number* nt *network*

Long Syntax: LNM.014 Activating MAC frame interface for port *port_number* network *network*

Description: LNM is activating the interface to the Token-Ring for the transfer of MAC frames to and from the specified port.

LNM.015

Level: C-INFO

Short Syntax: LNM.015 Proc net up ind for port *port_number* nt *network*

Long Syntax: LNM.015 Processing network up indication for port *port_number* network *network*

Description: LNM received an indication that an interface on which LNM is enabled is now up. LNM will perform actions necessary to start processing requests to or from the interface.

LNM.016

Level: C-INFO

Short Syntax: LNM.016 Proc net dwn ind for port *port_number* nt *network*

Long Syntax: LNM.016 Processing network down indication for port *port_number* network *network*

Description: LNM received an indication that an interface on which LNM is enabled is now down. LNM will terminate processing requests to or from the interface.

LNM.017

Level: UI-ERROR

Short Syntax: LNM.017 No memory to queue event

Long Syntax: LNM.017 No memory available to create an LNM event queue block

Description: No memory available to create an LNM event queue block. This is a fatal condition and in all

probability indicates a memory leak.

LNM.018

Level: C-INFO

Short Syntax: LNM.018 Rem cn req refused port *port_number* nt *network*

Long Syntax: LNM.018 Remote connection request refused for port *port_number* network *network*

Description: LNM received an indication that a connection request initiated by a remote station was received. LNM does not accept remote connection requests, so the connection request will be refused.

LNM.019

Level: C-INFO

Short Syntax: LNM.019 cn cnfm rcvd but not cnctg link *link* port *port_number* nt *network*

Long Syntax: LNM.019 A connect confirm indication was received but the link is not in connecting state for link *link* port *port_number* network *network*

Description: LNM received an indication that a previously issued connection request has been confirmed by LLC, but the state of the link indicates that no connection request is outstanding.

Cause: The outstanding connection request may have been cancelled due to a netdown condition.

LNM.020

Level: C-INFO

Short Syntax: LNM.020 disc rcvd when cnctg link *link* port *port_number* nt *network*

Long Syntax: LNM.020 A disconnect indication was received while the link is in connecting state for link *link* port *port_number* network *network*

Description: LNM received a disconnect indication while a previously issued connection request is outstanding.

LNM.021

Level: C-INFO

Short Syntax: LNM.021 disc rcvd but conn not act port *port_number* nt *network*

Long Syntax: LNM.021 A disconnect indication was received but the connection is not active on port *port_number* network *network*

Description: LNM received an indication that the specified link has been disconnected, but the state of the link indicates that the connection is not active.

Cause: The connection may have been closed due to a netdown condition.

LNM.022

Level: C-INFO

Short Syntax: LNM.022 reset rcvd link *link* port *port_number* nt *network*

Long Syntax: LNM.022 A reset indication was received for link *link* port *port_number* network *network*

Description: LNM received a reset indication for the specified link. LNM will return a reset response.

LNM.023

Level: C-INFO

Short Syntax: LNM.023 cannot open conn SAP clsd port *port_number* nt *network*

Long Syntax: LNM.023 Cannot open a connection SAP closed on port *port_number* network *network*

Description: LNM attempted to open a connection, but found that the LNM SAP had been closed.

Cause: The SAP may have been closed due to a netdown condition.

LNM.024

Level: C-INFO

Short Syntax: LNM.024 cannot open conn lnk in use lnk *link* port *port_number* nt *network*

Long Syntax: LNM.024 Cannot open a connection link in use link *link* port *port_number* network *network*

Description: LNM attempted to open a connection, but found that the requested link is already in use.

LNM.025

Level: C-INFO

Short Syntax: LNM.025 open sta fld rtn = *retval* lnk *link* port *port_number* nt *network*

Long Syntax: LNM.025 Open station failed return = *retval* link *link* port *port_number* network *network*

Description: LNM attempted to open a station, but LLC rejected the operation.

LNM.026

Level: C-INFO

Short Syntax: LNM.026 conn req fld rtn = *retval* lnk *link* port *port_number* nt *network*

Long Syntax: LNM.026 Connect req failed return = *retval* link *link* port *port_number* network *network*

Description: LNM attempted to open a connection, but LLC rejected the operation.

LNM.027

Level: C-INFO

Short Syntax: LNM.027 disc req fld rtn = *retval* lnk *link* port *port_number* nt *network*

Long Syntax: LNM.027 Disconnect req failed return = *retval* link *link* port *port_number* network *network*

Description: LNM attempted to disconnect a connection, but LLC rejected the operation.

LNM.028

Level: C-INFO

Short Syntax: LNM.028 netdwn rcvd clsg LNM SAP port *port_number* nt *network*

Long Syntax: LNM.028 Netdown received closing LNM SAP port *port_number* network *network*

Description: LNM received a network down indication for the specified port. LNM is closing the LNM SAP X'F4' as a result.

LNM.029

Level: C-INFO

Short Syntax: LNM.029 netup rcvd opening LNM SAP port *port_number* nt *network*

Long Syntax: LNM.029 Netup received, opening LNM SAP port *port_number* network *network*

Description: LNM received a network up indication for the specified port. LNM is opening the LNM SAP X'F4' as a result.

LNM.030

Level: C-INFO

Short Syntax: LNM.030 No rsrc for open LNM SAP port *port_number* nt *network*

Long Syntax: LNM.030 No resources for opening LNM SAP port *port_number* network *network*

Description: LLC indicated that not enough resources exist for opening the LNM SAP X'F4'. LNM will not be enabled as a result.

LNM.031

Level: C-INFO

Short Syntax: LNM.031 LNM UI frm not sent rsn = *reason* port *port_number* nt *network*

Long Syntax: LNM.031 LNM UI LLC frame not sent reason = *reason* port *port_number* network *network*

Description: LNM attempted to send a UI frame via LLC, but the frame could not be sent for the indicated reason.

LNM.032

Level: C-INFO

Short Syntax: LNM.032 LNM UI frm not sent net dwn port *port_number* nt *network*

Long Syntax: LNM.032 LNM UI LLC frame not sent network down port *port_number* network *network*

Description: LNM attempted to send a UI frame via LLC, but the frame could not be sent because the network interface is down.

LNM.033

Level: C-INFO

Short Syntax: LNM.033 LNM I frm not sent conn clsd port *port_number* nt *network*

Long Syntax: LNM.033 LNM I frame not sent connection closed port *port_number* network *network*

Description: LNM attempted to send an I frame via LLC, but the frame could not be sent because the connection has been closed.

Cause: The connection may have been closed because the network interface went down.

LNM.034

Level: C-INFO

Short Syntax: LNM.034 LNM I frm not sent rsn = *reason* link *link* port *port_number* nt *network*

Long Syntax: LNM.034 LNM I frame not sent reason = *reason* link *link* port *port_number* network *network*

Description: LNM attempted to send an I frame via LLC, but the frame could not be sent for the indicated reason.

LNM.035

Level: C-INFO

Short Syntax: LNM.035 packet rcvd but no connection on link *link* port *port_number* nt *network*

Long Syntax: LNM.035 An LLC packet was received but no connection exists for link number: *link* port *port_number* network *network*

Description: LNM received an LLC packet for an inactive link, possibly indicating that a previously activated link has become inactive.

Cause: The previous connection request may have been cancelled due to a netdown condition.

LNM.036

Level: C-INFO

Short Syntax: LNM.036 *server* PARSE error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.036 *server* LLC parsing error, code = *error* port *port* network *network msgptr*

Description: LNM received a LLC packet which contained a architectural syntax error and could not be parsed properly. The code defines the specific parsing failure.

Cause: Implementation error.

LNM.037

Level: C-INFO

Short Syntax: LNM.037 *server* EXECUTION error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.037 *server* EXECUTION error, code = *error* port *port* network *network msgptr*

Description: LNM received a LLC packet which, although syntactically correct, could not be executed.

Cause: The request in the packet cannot be executed or is not supported.

LNM.038

Level: C-INFO

Short Syntax: LNM.038 *server* PCK_ALLOC error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.038 *server* packet allocation error, code = *error* port *port* network *network msgptr*

Description: The indicated server component was unable to allocate a packet buffer.

Cause: Either the system is very busy, or more packet buffers need to be allocated.

LNM.039

Level: C-INFO

Short Syntax: LNM.039 *server* GET_CHAR error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.039 *server* error, code = *error* port *port* network *network msgptr*

Description: The indicated server component was unable to obtain the next character from a LLC packet.

Cause: Implementation error.

LNM.040

Level: C-INFO

Short Syntax: LNM.040 *server* error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.040 *server* error, code = *error* port *port* network *network msgptr*

Description: The indicated server component was unable to perform a packet character operation.

Cause: Implementation error.

LNM.041

Level: C-INFO

Short Syntax: LNM.041 *server* error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.041 *server* error, code = *error* port *port* network *network msgptr*

Description: The indicated server component was unable to perform a buffer operation.

Cause: Implementation error.

LNM.042

Level: C-INFO

Short Syntax: LNM.042 *server* error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.042 *server* error, code = *error* port *port* network *network msgptr*

Description: The indicated server component was unable to perform a timer operation.

Cause: Implementation error.

LNM.043

Level: C-INFO

Short Syntax: LNM.043 *server* error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.043 *server* error, code = *error* port *port* network *network msgptr*

Description: The indicated server component was unable to perform a socket operation.

Cause: Implementation error.

LNM.044

Level: C-INFO

Short Syntax: LNM.044 *server* error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.044 *server* error, code = *error* port *port* network *network msgptr*

Description: The indicated server component was unable to perform memory list.

Cause: Implementation error.

LNM.045

Level: C-INFO

Short Syntax: LNM.045 *server* error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.045 *server* error, code = *error* port *port* network *network msgptr*

Description: The indicated server component was unable to perform an LSS parse operation.

Cause: The TRD sent a bad packet.

LNM.046

Level: C-INFO

Short Syntax: LNM.046 *server* error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.046 *server* error, code = *error* port *port* network *network msgptr*

Description: The indicated server component received an error return status from LLC2.

Cause: Implementation or execution error.

LNM.047

Level: C-INFO

Short Syntax: LNM.047 *server* error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.047 *server* error, code = *error* port *port* network *network msgptr*

Description: The indicated server component was unable to perform an LSCM operation.

Cause: Configuration error.

LNM.048

Level: C-INFO

Short Syntax: LNM.048 *server* error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.048 *server* error, code = *error* port *port* network *network msgptr*

Description: The indicated server component received an error return status from LRM.

Cause: Implementation or execution error.

LNM.049

Level: C-INFO

Short Syntax: LNM.049 *server* error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.049 *server* error, code = *error* port *port* network *network msgptr*

Description: The indicated server component received an error return status from LBS.

Cause: Implementation or execution error.

LNM.050

Level: C-INFO

Short Syntax: LNM.050 *server* error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.050 *server* error, code = *error* port *port* network *network msgptr*

Description: The indicated server component received an error return status from CRS.

Cause: Implementation or execution error.

LNM.051

Level: C-INFO

Short Syntax: LNM.051 *server* error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.051 *server* error, code = *error* port *port* network *network msgptr*

Description: The indicated server component received an error return status from REM.

Cause: Implementation or execution error.

LNM.052

Level: C-INFO

Short Syntax: LNM.052 *server* error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.052 *server* error, code = *error* port *port* network *network msgptr*

Description: The indicated server component received an error return status from RPS.

Cause: Implementation or execution error.

LNM.053

Level: C-INFO

Short Syntax: LNM.053 *server* error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.053 *server* error, code = *error* port *port* network *network msgptr*

Description: The indicated server component received an error return status from TRD.

Cause: Implementation or execution error.

LNM.054

Level: C-INFO

Short Syntax: LNM.054 *server* error, code = *error* port *port* nt *network msgptr*

Long Syntax: LNM.054 *server* error, code = *error* port *port* network *network msgptr*

Description: The indicated server component received a system error return status.

Cause: Implementation error.

LNM.055

Level: C-INFO

Short Syntax: LNM.055 packet rcvd but no connection nt *network*

Long Syntax: LNM.055 An LLC packet was received but no connection exists for network *network*

Description: LNM received a LLC packet for an inactive link, possibly indicating that a previously activated link has become inactive. The data is discarded.

Cause: The previous connection request may have been cancelled due to a netdown condition.

LNM.056

Level: C-INFO

Short Syntax: LNM.056 packet rcvd but SAP not open nt *network*

Long Syntax: LNM.056 AN llC packet was received but the LNM SAP is not open for network *network*

Description: LNM received an LLC packet, but the LNM SAP is not open, possibly indicating that the interface has gone down. The data is discarded.

Cause: The LNM SAP may have been closed due to a netdown condition.

LNM.057

Level: U-INFO

Short Syntax: LNM.057 This LNM release supports only one LAN to one WAN bridge

Long Syntax: LNM.057 This LNM release supports only one LAN to one WAN bridge

Description: The first release of LNM (14.0a) is restricted to DNX 300 with LAN to WAN only.

Cause: User configuration error.

Action: Reconfigure bridge to be LAN to WAN or use any later release.

LNM.058

Level: C-TRACE

Short Syntax: LNM.058 LNM- *major-vector direction*, link *link*, port *port*, nt *network*

Long Syntax: LNM.058 LNM protocol message *major-vector direction*, link *link*, port *port*, network *network*

Description: This message traces all incoming and outgoing IBM LAN Network Manager protocol messages. Major-vectors values are described in the IBM Token-Ring Architecture Manual SC30-3374. Message direction, rcvd or sent, is indicated in the message. The link values are as follows: 0-3, if LINK is established; 242, for non-LINK messages (UNITDATA messages).

LNM.059

Level: C-TRACE

Short Syntax: LNM.059 MAC- *MAC-vector direction*, port *port*, nt *network*

Long Syntax: LNM.059 MAC protocol message *MAC-vector direction*, port *port*, network *network*

Description: This message traces all incoming and outgoing MAC messages. MAC-vector values are described in the IBM Token-Ring Architecture Manual, SC30-3370. Message direction rcvd or sent is indicated in the message.

LNM.060

Level: UI-ERROR

Short Syntax: LNM.060 Drp LNM frm, len *frame-length*, nt *network*

Long Syntax: LNM.060 Dropping LNM frame, length *frame-length*, network *network*

Description: The router is dropping an incoming message for one of two reasons: (1) The length of the frame is zero and LNM is defensively discarding the packet, or (2) LNM cannot obtain an internal LNM buffer, which is never expected to happen. The length of the frame appears so you can tell if it is reason (1) or (2). Customer service should be informed whether it is (1) or (2).

LNM.061

Level: U-INFO

Short Syntax: LNM.061 LNM support *action* in LAN Switch: domain *domain-index* net *network*

Long Syntax: LNM.061 LNM support *action* in LAN Switch: domain *domain-index* network *network*

Description: LNM support was configured for a network related to a port on the LAN Switch. The corresponding domain on the LAN Switch is listed. The action requested was successful.

Cause: None.

Action: None.

LNM.062

Level: U-INFO

Short Syntax: LNM.062 LNM support failed for domain *domain-index* net *network* action *action* error code *rc*

Long Syntax: LNM.062 LNM support failed for domain *domain-index* network *network* action *action* error code *rc*

Description: LNM support was configured for a network related to a port on the LAN Switch. The corresponding domain on the LAN Switch is listed. The action requested of this support has failed for the reason indicated by one of the following error codes:. 1 - Invalid domain number 2 - Multiple physical ports configured in domain 3 - No ports configured in domain 4 - Unable to set functional addresses on port 5 - Switch microcode is down level 6 - Incompatible ARI/FCI option (set to non-routed only) 7 - Incompatible hardware (MPC 3.0)

Cause: User Configuration Error.

Action: Determine which error code has been returned and reconfigure the appropriate devices.

Chapter 67. LAN Switch Integration (LSI)

This chapter describes LAN Switch Integration (LSI) messages. For information on message content and how to use the message, refer to the Introduction.

LSI.001

Level: U-INFO

Short Syntax: LSI.001 Rcv *packet_type* len *length* dom *domain* from *from_hw_add* to *to_hw_add* net *network ID*

Long Syntax: LSI.001 Received a packet type: *packet_type* length *length* on logical port *domain* from *from_hw_add* to *to_hw_add* to net *network ID*

Description: A frame is received from the lan switch.

Cause: A frame is received from the lan switch.

LSI.002

Level: U-INFO

Short Syntax: LSI.002 Snd *type* len *length* dom *domain* from *from_hw_add* to *to_hw_add* net *network ID*

Long Syntax: LSI.002 Sending a packet type *type* of length *length* on domain *domain* from *from_hw_add* to *to_hw_add* from net *network ID*

Description: A frame is being sent to the lan switch.

Cause: A frame is being sent to the lan switch.

LSI.003

Level: U-INFO

Short Syntax: LSI.003 Frame rcv too big for net Max: *max* Len: *length* Elp: *elp* from *from_hw_add* to *to_hw_add* net *network ID*

Long Syntax: LSI.003 Frame received too large for net Max: *max* Len: *length* Elp: *elp* from *from_hw_add* to *to_hw_add* net *network ID*

Description: A packet is received that is too large to be processed by the net interface.

Cause: A packet is received that is too large to be processed by the net interface.

LSI.004

Level: U-INFO

Short Syntax: LSI.004 Disc frame *type* len *length* from dom: *domain* from *from_hw_add* to *to_hw_add* type: *elptype*

Long Syntax: LSI.004 Discard frame *type* length *length* from domain *domain* from *from_hw_add* to *to_hw_add* type: *elptype*

Description: A packet is received that does not match the addressing of the associated net.

Cause: Packet received that is not needed by the associated nets for this domain, and their configured protocols.

LSI.005

Level: U-INFO

Short Syntax: LSI.005 Rif Walk post bdg Dom: *domain* DestSeg: *sseg* SrcSeg *pseg* ret:

Long Syntax: LSI.005 Rif Walk post bdg Domain: *domain* DestSeg: *sseg* SrcSeg *pseg* ret:

Description: A packet was received from the bridge function that needed to be rerouted to its correct source segment. source bridge segment.

Cause: Packet received post bridge.

LSI.006

Level: P-TRACE

Short Syntax: LSI.006 Trace LSI packet

Long Syntax: LSI.006 Trace LSI packet

Description: Trace an input or output packet.

LSI.007

Level: U-INFO

Short Syntax: LSI.007 VCC Handle: *txId* added for lcd: *vcc* domain: *domain*

Long Syntax: LSI.007 VCC Handle: *txId* added for lcd: *vcc* domain: *domain*

Description: A VCC Handle has been added to this LCD for hardware path.

LSI.008

Level: U-INFO

Short Syntax: LSI.008 TxID: *txId* added for VCC Handle: *vcc* domain: *domain*

Long Syntax: LSI.008 TxID: *txId* added for VCC Handle: *vcc* domain: *domain*

Description: A lan switch transmit channel has been created.

LSI.009

Level: U-INFO

Short Syntax: LSI.009 mac addr *mac* assigned TxID: *txId* domain: *domain*

Long Syntax: LSI.009 mac addr *mac* assigned TxID: *txId* domain: *domain*

Description: A mac address will be associated with a txid, so that packets destined to that address will be switched via hardware from the lan switch to the atm network.

LSI.010

Level: U-INFO

Short Syntax: LSI.010 Route Desc *rd* assigned TxID: *txId* domain: *domain*

Long Syntax: LSI.010 Route Desc *rd* assigned TxID: *txId* domain: *domain*

Description: A route descriptor will be associated with a txid, so that packets destined to that route descriptor will be switched via hardware from the lan switch to the atm network.

LSI.011

Level: U-INFO

Short Syntax: LSI.011 HwPath VCC deleted vcc_id: *vcc* lcd_p: *lcd_p*

Long Syntax: LSI.011 HwPath VCC deleted vcc_id: *vcc* lcd_p: *lcd_p*

Description: A vcc has been deleted from the bridged hardware path.

LSI.012

Level: U-INFO

Short Syntax: LSI.012 TxId deleted id: *txid*

Long Syntax: LSI.012 TxId deleted id: *txid*

Description: A txid has been deleted from the bridged hardware path when the associated VCC has also been deleted.

LSI.014

Level: U-INFO

Short Syntax: LSI.014 mac addr *mac* reassigned txId: *txId* domain: *domain* sw pth

Long Syntax: LSI.014 mac addr *mac* reassigned txId: *txId* domain: *domain* sw pth

Description: A mac address associated with a txid is being reassigned, so that packets destined to that address will be processed by the software to the atm network.

LSI.015

Level: U-INFO

Short Syntax: LSI.015 Route Desc *rd* reassigned txId: *txId* domain: *domain* sw pth

Long Syntax: LSI.015 Route Desc *rd* reassigned txId: *txId* domain: *domain* sw pth

Description: A Route Descriptor associated with a txid is being reassigned, so that packets destined to that route will be processed by the software to the atm network.

LSI.016

Level: U-INFO

Short Syntax: LSI.016 Packet Disc bad Rif len: *len* domain: *domain* Intf: *intf*

Long Syntax: LSI.016 Packet Disc bad Rif len: *len* domain: *domain* Intf: *intf*

Description: A packet was discard that was sent from the software bridge into the lan switch because of a bad rif field in the packet.

Cause: Internal software error.

LSI.017

Level: U-INFO

Short Syntax: LSI.017 Packet Disc bad txid: *txid* len: *len*

Long Syntax: LSI.017 Packet Disc bad txid: *txid* len: *len*

Description: A packet was discard that was received from the lan switch because of an invalid txchannel.

Cause: Internal software error.

LSI.018

Level: U-INFO

Short Syntax: LSI.018 Snd Pkt Disc no mimic buffer dom: *dom,* len: *len*

Long Syntax: LSI.018 Send Packet Disc no mimic buffers dom: *dom,* len: *len*

Description: A packet was discarded because there were no mimic buffers available to copy the packet into for transmission into the lan switch.

Cause: Internal software error.

LSI.019

Level: U-INFO

Short Syntax: LSI.019 Get Domain Members Request issued for domain: *domain*

Long Syntax: LSI.019 Get Domain Members Request issued for domain: *domain*

Description: The Get Domain Members Request has been issued to the lan switch.

LSI.020

Level: U-INFO

Short Syntax: LSI.020 Get Domain Members Response received for domain: *domain*

Long Syntax: LSI.020 Get Domain Members Response received for domain: *domain*

Description: The Get Domain Members Response has been received from the lan switch.

LSI.021

Level: U-INFO

Short Syntax: LSI.021 function:feCrfMembersResponse, LEC Net pointer for domain: *domain* is 0

Long Syntax: LSI.021 Net pointer for domain: *domain* is 0

Description: The net pointer associated with the Proxy LEC is invalid.

LSI.022

Level: UE-ERROR

Short Syntax: LSI.022 VCC Handle not available for lcd: *vcc* domain: *domain*

Long Syntax: LSI.022 VCC Handle not available for lcd: *vcc* domain: *domain*

Description: No more VCC Handles are available for hardware path.

LSI.023

Level: UE-ERROR

Short Syntax: LSI.023 TxID not available for VCC Handle: *vcc* domain: *domain*

Long Syntax: LSI.023 TxID not available for VCC Handle: *vcc* domain: *domain*

Description: No more Transmit Channel IDs are available for hardware path.

LSI.024

Level: U-INFO

Short Syntax: LSI.024 MPOA mac addr *mac* was NOT assigned to TxID: *txId* domain: *domain*

Long Syntax: LSI.024 MPOA mac addr *mac* was NOT assigned to TxID: *txId* domain: *domain*

Description: The MAC address is an MPOA address and will NOT be associated with a TxID.

LSI.025

Level: UE-ERROR

Short Syntax: LSI.025 net *bdg* and net *rtr* are both SRB bridged on domain *dom*, net *network ID* disabled

Long Syntax: LSI.025 net *bdg* and net *rtr* are both SRB bridged on domain *dom*, net *network ID* disabled

Description: Two SRB interfaces, with TWO bridge segements have been assigned to the same domain. This is an invalid configuration, which has resulted in the second of these two interfaces being forced into a disabled state.

LSI.026

Level: U-INFO

Short Syntax: LSI.026 A packet was discarded because the destined net was down

Long Syntax: LSI.026 A packet was discarded because the destined net was down

Description: A packet was discarded beecause its detined net was disabled, or down.

LSI.027

Level: UE-ERROR

Short Syntax: LSI.027 The hw path received a null pointer on rx

Long Syntax: LSI.027 The hw path received a null pointer on rx

Description: An event was received by the hardware bridge from the network interface, that had a null packet pointer.

LSI.028

Level: UE-ERROR

Short Syntax: LSI.028 The hw path detected a PCI error

Long Syntax: LSI.028 The hw path detected a PCI error

Description: The hardware path detected a PCI error.

Additional messages may follow that give additional details.

LSI.029

Level: UE-ERROR

Short Syntax: LSI.029 The hw path detected a PCI DMA error

Long Syntax: LSI.029 The hw path detected a PCI DMA error

Description: The hardware path detected a PCI DMA error. Additional messages may follow that give additional details.

LSI.030

Level: UE-ERROR

Short Syntax: LSI.030 The hw path detected a PCI DMA error on the transmit path

Long Syntax: LSI.030 The hw path detected a PCI DMA error on the transmit path

Description: The hardware path detected a PCI DMA error. Additional messages may follow that give additional details.

LSI.031

Level: UE-ERROR

Short Syntax: LSI.031 The hw path detected a PCI DMA error on the receive path

Long Syntax: LSI.031 The hw path detected a PCI DMA error on the receive path

Description: The hardware path detected a PCI DMA error. Additional messages may follow that give additional details.

LSI.032

Level: UE-ERROR

Short Syntax: LSI.032 The hw path detected an error in a packet that was received from the lan switch

Long Syntax: LSI.032 The hw path detected an error in a packet that was received from the lan switch

Description: The hardware path detected an error in the control block header on packet received from the lan switch destined out the network interface.

LSI.033

Level: UE-ERROR

Short Syntax: LSI.033 The hw path detected target abort signal on the PCI bus.

Long Syntax: LSI.033 The hw path detected target abort signal on the PCI bus.

Description: The hardware path detected an error when trying to transfer a packet from, or to the network interface.

LSI.034

Level: UE-ERROR

Short Syntax: LSI.034 The hw path detected master abort signal on the PCI bus.

Long Syntax: LSI.034 The hw path detected master abort signal on the PCI bus.

Description: The hardware path detected an error when trying to transfer a packet from, or to the network interface.

LSI.035

Level: UE-ERROR

Short Syntax: LSI.035 The hw path received a parity error signal from a target.

Long Syntax: LSI.035 The hw path received a parity error signal from a target.

Description: The hardware path detected an error when trying to transfer a packet to the network interface. This was detected on the network side as the packetwas being DMAed into its memory space.

LSI.036

Level: UE-ERROR

Short Syntax: LSI.036 The hw path received a detected a parity on the pci bus.

Long Syntax: LSI.036 The hw path received a detected a parity on the pci bus.

Description: The hardware path detected an error when trying to transfer a packet from the network interface. This was detected on the hardware path side as the packet was being DMAed into its memory space.

LSI.037

Level: UE-ERROR

Short Syntax: LSI.037 The hw path dma was unexpectedly halted.

Long Syntax: LSI.037 The hw path dma was unexpectedly halted.

Description: The hardware path DMA entity was unexpectdly halted. A restart will be attempted, but the hardware path may not be able to recover.

LSI.038

Level: UE-ERROR

Short Syntax: LSI.038 An inv int was reg net *network ID/ dom* on dom:

Long Syntax: LSI.038 An inv int was reg net *network ID/ dom* on dom:

Description: A invalid network was registered to the LSI client. This was probably caused by an invalid cofiguration, or a corrupted config.

LSI.039

Level: UE-ERROR

Short Syntax: LSI.039 A hw path reset timed out.

Long Syntax: LSI.039 A hw path reset timed out.

Description: A reset was issued to the hw path, and it timed out before the proper state was attained. This could be the result of a PCI error, or other problems. The system may still be operational.

LSI.040

Level: UE-ERROR

Short Syntax: LSI.040 A hw path looped packet failed.

Long Syntax: LSI.040 A hw path looped packet failed.

Description: A looped packet to the switch was not successfully received. This may result in the a severe error on the system that may require a reset. The system may still be operational.

LSI.041

Level: UI-ERROR

Short Syntax: LSI.041 nt *network error_lvl log_point*

Long Syntax: LSI.041 network *network*: lsi error log: *error_lvl log_point*

Description: lsi generic error

LSI.042

Level: UI-ERROR

Short Syntax: LSI.042 inproper looped frame *type* len *length* from dom: *domain* from *from_hw_add* to *to_hw_add* type: *elptype*

Long Syntax: LSI.042 inproper looped frame *type* len *length* from dom: *domain* from *from_hw_add* to *to_hw_add* type: *elptype*

Description: A frame was received that was inappropiately looped back from the MSS UFC card, back to the MSS card. This may be the result of a back level version of switch micro code.

LSI.043

Level: U-INFO

Short Syntax: LSI.043 loop Snd *type* len *length* dom *domain* from *from_hw_add* to *to_hw_add* net *network ID*

Long Syntax: LSI.043 looping a packet type *type* of length *length* on domain *domain* from *from_hw_add* to *to_hw_add* from net *network ID*

Description: A frame is being sent to the lan switch with the purpose of having the frame looped back into the MSS Card.

LSI.044

Level: U-INFO

Short Syntax: LSI.044 Disc frm out inl: t *type* dom *length* len *domain from_hw_add -> to_hw_add* n *network ID*

Long Syntax: LSI.044 Disc frame out when in_low set:type *type* dom *length* length *domain* from *from_hw_add* to *to_hw_add* net *network ID*

Description: A frame that was to be sent to the lan switch was discarded because the source interface was in a buffer depleted state.

LSI.045

Level: U-INFO

Short Syntax: LSI.045 Buff not avail for cpy pckt: type: *type* length *length* dom *domain* from *from_hw_add* to *to_hw_add* net *network ID*

Long Syntax: LSI.045 Buff not avail for cpy pckt: type: *type* length *length* dom *domain* from *from_hw_add* to *to_hw_add* net *network ID*

Description: A frame that was to be sent from the LAN Switch to the MSS Card was to be copied to a second net that is configured on this domain. Since the LAN Switch Interface was low on buffers, a copy of the buffer could not be sent to the second interface. If this event happens frequently, it is recomended that the second interface be moved to its own domain if possible, or if the interface is being used for Box Management only, IPHOST be used instead.

LSI.046

Level: U-INFO

Short Syntax: LSI.046 LAN Switch Out queue over flow

Long Syntax: LSI.046 LAN Switch Out queue over flow

Description: The LAN Switch is sending the MSS UFC more taffic than the MSS UFC is capable of receiving at this time. An uspecified number of packets

may have been discarded. Although this event should not be considered an error, it may be an indication of network over utilization if this event is received at high rates. Network reconfiguration, or filtering of iproper traffic may lesson this error.

Chapter 68. Layer Two Tunneling (L2)

This chapter describes Layer Two Tunneling (L2) messages. For information on message content and how to use the message, refer to the Introduction.

L2.001

Level: UI_ERROR

Short Syntax: L2.001 ERROR: *errorString*

Long Syntax: L2.001 ERROR: *errorString*

Description: General Error - If this happens, something is seriously wrong. A brief description of what happened and/or where it happened will accompany this message.

L2.002

Level: CI_ERROR

Short Syntax: L2.002 WARNING: *warningString*

Long Syntax: L2.002 WARNING: *warningString*

Description: General Warning - This can happen in normal conditions when resources are depleted (buffers, nets, calls, tunnels, etc.) or when our peer does something unexpected. A brief description of what happened and/or where it happened will accompany this message.

L2.003

Level: C-INFO

Short Syntax: L2.003 L2 Slf Tst net *net*

Long Syntax: L2.003 Performing Self Test on L2 network *net*

Description: Entering l2tp_slftst

L2.004

Level: C-INFO

Short Syntax: L2.004 L2 init net *net*

Long Syntax: L2.004 Intitalizing L2 network *net*

Description: Entering l2tp_init

L2.005

Level: C-INFO

Short Syntax: L2.005 L2 install net *net*

Long Syntax: L2.005 Installing L2 network *net*

Description: Entering l2tp_install

L2.006

Level: C-INFO

Short Syntax: L2.006 LCP start net *net* cause= *cause*

Long Syntax: L2.006 Renegotiate LCP-L2 net *net*, cause= *cause*

Description: LCP restarted on the L2 network due to the listed cause.

L2.007

Level: C-INFO

Short Syntax: L2.007 LNS *action* L2 net *net*

Long Syntax: L2.007 LNS *action* L2 network *net*

Description: Allocated/Freed (action) an L2 Network on the LNS

L2.008

Level: P_TRACE

Short Syntax: L2.008 Call Make *AVPtype* AVP,attr= *attributeNo*,val= *value*,len= *length*,flag= *flags*

Long Syntax: L2.008 Call Make *AVPtype* AVP,attr= *attributeNo*,val= *value*,len= *length*,flag= *flags*

Description: Call is creating an AVP with the attributes given. A value of 0 can indicate that the value is very long and could not be displayed in the message.

L2.009

Level: P_TRACE

Short Syntax: L2.009 Call Rcv *AVPtype* AVP,attr= *attributeNo*,val= *value*,len= *length*,flag= *flags*

Long Syntax: L2.009 Call Rcv *AVPtype* AVP,attr= *attributeNo*,value= *value*,len= *length*,flags= *flags*

Description: Call is receiving an AVP with the attributes given. A value of 0 can indicate that the value is very long and could not be displayed in the message, or is not supported locally.

L2.010

Level: C_TRACE

Short Syntax: L2.010 Start Call LAC net *net*,speed=

speed,btype= *bearer*,frame= *framing*,auth= *proxy_auth*

Long Syntax: L2.010 Start Call LAC net *net*,speed= *speed*,btype= *bearer*,ftype= *framing*,auth= *proxy_auth*

Description: Starting Call FSM from LAC. Listed parameters are speed (bits/sec), bearer type, framing type, and proxy-auth-type.

L2.011

Level: C_TRACE

Short Syntax: L2.011 Stopping Call id= *callid*, net *net* int /

Long Syntax: L2.011 Stopping Call id= *callid*, net *net* int /

Description: Stopping a call

L2.012

Level: C-INFO

Short Syntax: L2.012 Local Term net *net*

Long Syntax: L2.012 Local Terminate L2 network *net*

Description: Local Terminate means graceful teardown or a physical down event for any reason.

L2.013

Level: C-INFO

Short Syntax: L2.013 *type* Call *id* State Changed *old* -> *new*

Long Syntax: L2.013 *type* Call *id* State Changed *old* -> *new*

Description: Call state changed.

L2.014

Level: C_TRACE

Short Syntax: L2.014 *CallORTunnel* from net *net*

Long Syntax: L2.014 *CallORTunnel* from net *net*

Description: This is the Entry into the L2 system. Starting a call (and possibly needing to start a tunnel).

L2.015

Level: C_TRACE

Short Syntax: L2.015 Call Established- *Type*,net= *net*,speed= *speed*,flags= *sendFlags*

Long Syntax: L2.015 Call Established- *Type*,net= *net*,speed= *speed*,flags= *sendFlags*

Description: Call FSM reached the established state.

L2.016

Level: C_TRACE

Short Syntax: L2.016 Forcing intial CHAP challenge

Long Syntax: L2.016 Forcing intial CHAP challenge

Description: The peer has sent Proxy-CHAP, but the user has configured to force an initial CHAP re-challenge for security reasons.

L2.017

Level: C_TRACE

Short Syntax: L2.017 Using Proxy- *authType* AUTH on net *net*

Long Syntax: L2.017 Using Proxy- *authType* AUTH on net *net*

Description: LNS is accepting the proxy-auth type indicated from the peer.

L2.018

Level: C_TRACE

Short Syntax: L2.018 Aborting Call, callid= *callid*, net= *net*

Long Syntax: L2.018 Aborting Call, callid= *callid*, net= *net*

Description: Aborting Call

L2.019

Level: C_TRACE

Short Syntax: L2.019 Cleaning up *type* Call id= *callid*

Long Syntax: L2.019 Cleaning up *type* Call id= *callid*

Description: Cleaning up call structure and state

L2.020

Level: P_TRACE

Short Syntax: L2.020 RCV *type*, callid= *callid*, net= *net*

Long Syntax: L2.020 RCV *type*, callid= *callid*, net= *net*

Description: Receive the indicated call control message. Net is 0 for a resource condition.

L2.021

Level: P_TRACE

Short Syntax: L2.021 SEND *type*, callid= *callid*, net= *net*

Long Syntax: L2.021 SEND *type*, callid= *callid*, net= *net*

Description: Send the indicated call control message. Net is 0 for a resource condition.

L2.022

Level: P_TRACE

Short Syntax: L2.022 *protocol* PAYLOAD RCVD *bytes* bytes, net *net*, callid= *cid*

Long Syntax: L2.022 *protocol* PAYLOAD RCVD *bytes* bytes, net *net*, callid= *cid*

Description: Received a PAYLOAD packet on LAC/LNS or PAC/PNS

L2.023

Level: P_TRACE

Short Syntax: L2.023 Send *type* Zero Len Body (ZLB), tid= *tid*,cid= *cid*

Long Syntax: L2.023 Send *type* Zero Len Body (ZLB), tid= *tid*,cid= *cid*

Description: About to send a ZLB to the peer to ACK.

L2.024

Level: P_TRACE

Short Syntax: L2.024 *protocol* PAYLOAD SEND *bytse* bytes, net= *net*, callid= *cid*

Long Syntax: L2.024 *protocol* PAYLOAD SEND *bytse* bytes, net= *net*, callid= *cid*

Description: Send L2TP/PPTP Payload

L2.025

Level: P_TRACE

Short Syntax: L2.025 LNS Rcvd Proxy-Lcp *type* updating *local_remote*

Long Syntax: L2.025 LNS Rcvd Proxy-Lcp *type* updating *local_remote*

Description: LNS received proxy-lcp from LAC

L2.026

Level: P_TRACE

Short Syntax: L2.026 LNS Forcing LCP MRU= *mru*

Long Syntax: L2.026 LNS Forcing LCP MRU= *mru*

Description: LNS processing proxy-mru from LAC

L2.027

Level: P_TRACE

Short Syntax: L2.027 LNS Forcing LCP ACCM= *accm*

Long Syntax: L2.027 LNS Forcing LCP ACCM= *accm*

Description: LNS processing proxy-accm from LAC

L2.028

Level: P_TRACE

Short Syntax: L2.028 LNS Forcing LCP Auth= *auth_type*

Long Syntax: L2.028 LNS Forcing LCP Auth= *auth_type*

Description: LNS processing proxy-lcp auth from LAC

L2.029

Level: P_TRACE

Short Syntax: L2.029 LNS Forcing LCP Magic Number= *auth_type*

Long Syntax: L2.029 LNS Forcing LCP Magic Number= *auth_type*

Description: LNS processing proxy-magic-number from LAC

L2.030

Level: P_TRACE

Short Syntax: L2.030 LNS Forcing LCP option *option*

Long Syntax: L2.030 LNS Forcing LCP option *option*

Description: LNS processing proxy-lcp option from LAC

L2.031

Level: P_TRACE

Short Syntax: L2.031 LNS Forcing LCP Quality prot= *prot*, period= *period*

Long Syntax: L2.031 LNS Forcing LCP Quality prot= *prot*, period= *period*

Description: LNS processing proxy-lcp quality from LAC

L2.032

Level: P_TRACE

Short Syntax: L2.032 LNS Forcing LCP MRRU= *MRRU*

Long Syntax: L2.032 LNS Forcing LCP MRRU= *MRRU*

Description: LNS processing proxy-mrru from LAC

L2.033

Level: P_TRACE

Short Syntax: L2.033 LNS Forcing LCP Endpt Disc cls= *class*, addr= *address*

Long Syntax: L2.033 LNS Forcing LCP Endpt Disc cls= *class*, addr= *address*

Description: LNS processing proxy-endpt-discriminator from LAC

L2.034

Level: P_TRACE

Short Syntax: L2.034 LNS Forcing LCP Link-Discriminator= *ld*

Long Syntax: L2.034 LNS Forcing LCP Link-Discriminator= *ld*

Description: LNS processing proxy-link-discriminator from LAC

L2.035

Level: P_TRACE

Short Syntax: L2.035 Tunnel Auth Create *type*, Tid= *tid/ len*, Len=

Long Syntax: L2.035 Tunnel Auth Create *type*, Tid= *tid/ len*, Len=

Description: Creating Tunnel Auth AVPs

L2.036

Level: P_TRACE

Short Syntax: L2.036 Create Result Code AVP:rslt= *result*,err= *error*

Long Syntax: L2.036 Create Result Code AVP:rslt= *result*,err= *error*

Description: Creating result code AVP

L2.037

Level: C_INFO

Short Syntax: L2.037 l2tp_conf init L2-tunneling maxcalls= *maxcalls* maxtunnels= *maxtunnels*

Long Syntax: L2.037 l2tp_conf init L2-tunneling maxcalls= *maxcalls* maxtunnels= *maxtunnels*

Description: performing layer-2-tunneling intialization

L2.038

Level: CE_ERROR

Short Syntax: L2.038 PPP Discard packet - setting up tunnel, net *net*

Long Syntax: L2.038 PPP Discard packet - setting up tunnel, net *net*

Description: discarding PPP packet from client because the tunnel/call has not been established.

L2.039

Level: C_INFO

Short Syntax: L2.039 NOTE: *note_msg*

Long Syntax: L2.039 NOTE: *note_msg*

Description: General Note

L2.040

Level: P_TRACE

Short Syntax: L2.040 RCV *protocol*:F= *flags*,L= *length*,Tid= *tunnelid*,Cid= *callid*,NS= *ns*,NR= *nr*,O= *offset*

Long Syntax: L2.040 RCV *protocol*:F= *flags*,L= *length*,Tid= *tunnelid*,Cid= *callid*,NS= *ns*,NR= *nr*,O= *offset*

Description: Layer-2-Tunneling component received a tunneled packet. It is important to note that some of the displayed fields may NOT have been rcvd - use the "flags" mask to find which ones were rcvd.

L2.041

Level: P_TRACE

Short Syntax: L2.041 SND *protocol*:F= *flags*,L= *length*,Tid= *tunnelid*,Cid= *callid*,NS= *ns*,NR= *nr*,O= *offset*

Long Syntax: L2.041 SND *protocol*:F= *flags*,L= *length*,Tid= *tunnelid*,Cid= *callid*,NS= *ns*,NR= *nr*,O= *offset*

Description: Layer-2-Tunneling component sending a tunneled packet. It is important to note that some of the displayed fields may NOT have been sent - use the "flags" mask to find which ones were sent.

L2.042

Level: P_TRACE

Short Syntax: L2.042 Rcvd pkt udp_len= *WUDP_LEN*, L2_len= *L2_LEN*

Long Syntax: L2.042 Rcvd pkt udp_len= *WUDP_LEN*, L2_len= *L2_LEN*

Description: UDP pkt length does not match L2 packet length

L2.043

Level: P_TRACE

Short Syntax: L2.043 RCV *type* Zero Len Body (ZLB), tid= *tid*,cid= *cid*

Long Syntax: L2.043 RCV *type* Zero Len Body (ZLB), tid= *tid*,cid= *cid*

Description: Received a ZLB from our peer

L2.044

Level: C_INFO

Short Syntax: L2.044 Allocating UDP port *port* for tunnelid= *tid*

Long Syntax: L2.044 Allocating UDP port *port* for tunnelid= *tid*

Description: allocated a UDP source port for tunnel

L2.045

Level: P_TRACE

Short Syntax: L2.045 *packet* Pkt Queued for delayed *type* XMT, id= *id*

Long Syntax: L2.045 *packet* Pkt Queued for delayed *type* XMT, id= *id*

Description: The Layer-2-Tunneling system is busy (xmt window is full), this packet was queued for delayed transmit.

L2.046

Level: C_INFO

Short Syntax: L2.046 Clearing callid= *callid*, tunnelid= *tunnelid*

Long Syntax: L2.046 Clearing callid= *callid*, tunnelid= *tunnelid*

Description: clearing call

L2.047

Level: C-INFO

Short Syntax: L2.047 Tunnel *tid/ peer-tid* State Changed *old -> new*

Long Syntax: L2.047 Tunnel *tid/ peer-tid* State Changed *old -> new*

Description: Call state changed.

L2.048

Level: P_TRACE

Short Syntax: L2.048 RCV *type*, tid= *tid/ peer-tid*

Long Syntax: L2.048 RCV *type*, tid= *tid/ peer-tid*

Description: Receive the indicated tunnel control message.

L2.049

Level: P_TRACE

Short Syntax: L2.049 SEND *type*, tid= *tid/ peer-tid*

Long Syntax: L2.049 SEND *type*, tid= *tid/ peer-tid*

Description: Send the indicated tunnel control message.

L2.050

Level: C_INFO

Short Syntax: L2.050 EVENT *event*,tid= *tunnelid/ peerid*,state= *state*

Long Syntax: L2.050 EVENT *event*,tid= *tunnelid/ peerid*,state= *state*

Description: Tunnel Originator/Receiver Event

L2.051

Level: C_INFO

Short Syntax: L2.051 Cleaning up tunnelid *tid/ peerid*

Long Syntax: L2.051 Cleaning up tunnelid *tid/ peerid*

Description: cleaning up tunnel

L2.052

Level: C_INFO

Short Syntax: L2.052 Tunnel *tid/ peer-tid* has *seconds* seconds to establish itself

Long Syntax: L2.052 Tunnel *tid/ peer-tid* has *seconds* seconds to establish itself

Description: Bring down the tunnel if it is not established in <seconds> seconds.

L2.053

Level: C_INFO

Short Syntax: L2.053 tid *tid/ peerid* will shutdown in *seconds* seconds

Long Syntax: L2.053 tid *tid/ peerid* will shutdown in *seconds* seconds

Description: Tunnel will shutdown in <seconds> seconds.

L2.054

Level: C_INFO

Short Syntax: L2.054 Assigning tunnel peer *peer*, tid= *IP/ address*

Long Syntax: L2.054 Assigning tunnel peer *peer*, tid= *IP/ address*

Description: Assigning tunnel

L2.055

Level: C_INFO

Short Syntax: L2.055 Delayed Tunnel clean-up tid= *tid*/ *peer-tid*, already doomed

Long Syntax: L2.055 Delayed Tunnel clean-up tid= *tid*/ *peer-tid* already doomed

Description: Delayed tunnel clean-up

L2.056

Level: C_INFO

Short Syntax: L2.056 Call to kill tunnel *tid*/ now, already doomed, die gracefully

Long Syntax: L2.056 Call to kill tunnel *tid*/ now, already doomed, die gracefully

Description: Kill tunnel

L2.057

Level: P_TRACE

Short Syntax: L2.057 Processing Challenge Response from Peer *peer*

Long Syntax: L2.057 Processing Challenge Response from Peer *peer*

Description: process challenge response

L2.058

Level: P_TRACE

Short Syntax: L2.058 Peer *Attribute* = *value*

Long Syntax: L2.058 Peer *Attribute* = *value*

Description: process Tunnel AVP (value is integer)

L2.059

Level: P_TRACE

Short Syntax: L2.059 Peer *Attribute* = *value*

Long Syntax: L2.059 Peer *Attribute* = *value*

Description: process Tunnel AVP (value is hex)

L2.060

Level: P_TRACE

Short Syntax: L2.060 Peer *Attribute* = *value*

Long Syntax: L2.060 Peer *Attribute* = *value*

Description: process Tunnel AVP (value is string)

L2.061

Level: C_INFO

Short Syntax: L2.061 unsuccessful result: code= *result*,error= *error* msg=

Long Syntax: L2.061 unsuccessful result: code= *result*,error= *error* msg=

Description: Result of processing Start Control Connection Request/Reply/Connected.

L2.062

Level: C_INFO

Short Syntax: L2.062 *action* all calls on tunnel *tid*

Long Syntax: L2.062 *action* all calls on tunnel *tid*

Description: killing/clearing all calls on tunnel.

L2.063

Level: P_TRACE

Short Syntax: L2.063 Retransmit *msgtype* on tunnel *tid* call *cid*

Long Syntax: L2.063 Retransmit *msgtype* on tunnel *tid* call *cid*

Description: Retransmitting packet after timeout waiting for ACK.

L2.064

Level: CE_ERROR

Short Syntax: L2.064 Timeout waiting for ACK call *cid*

Long Syntax: L2.064 Timeout waiting for ACK call *cid*

Description: ACK timeout

L2.065

Level: CE_ERROR

Short Syntax: L2.065 Declaring LOST pkt on call *cid*

Long Syntax: L2.065 Declaring LOST pkt on call *cid*

Description: declare lost packet

L2.066

Level: CE_ERROR

Short Syntax: L2.066 Tunnel Retransmit limit exceeded - killing tunnel *tid*

Long Syntax: L2.066 Tunnel Retransmit limit exceeded - killing tunnel *tid*

Description: Tunnel retransmit maximum

L2.067

Level: C_INFO

Short Syntax: L2.067 Result Code Rx code= *result_code* error= *error_code* msg= *message*

Long Syntax: L2.067 Result Code Rx code= *result_code* error= *error_code* msg= *message*

Description: Result Code recevied

L2.068

Level: C_INFO

Short Syntax: L2.068 L2TPSEC: Processing Encoded Key AVP *a b c d e f g h i j k l m n o p*

Long Syntax: L2.068 L2TPSEC: Processing Encoded Key AVP *a b c d e f g h i j k l m n o p*

Description: l2tpsec encoded key avp

L2.069

Level: C_INFO

Short Syntax: L2.069 L2TPSEC: Decoded Key: *a b c d e f g h i j k l m n o p*

Long Syntax: L2.069 L2TPSEC: Decoded Key: *a b c d e f g h i j k l m n o p*

Description: l2tpsec encoded key avp

L2.070

Level: C_INFO

Short Syntax: L2.070 L2TPSEC: Create Encoded Key AVP *a b c d e f g h i j k l m n o p*

Long Syntax: L2.070 L2TPSEC: Create Encoded Key AVP *a b c d e f g h i j k l m n o p*

Description: l2tpsec encoded key avp

L2.071

Level: C_INFO

Short Syntax: L2.071 L2TPSEC: Actual Key is: *a b c d e f g h i j k l m n o p*

Long Syntax: L2.071 L2TPSEC: Actual Key is: *a b c d e f g h i j k l m n o p*

Description: l2tpsec encoded key avp

L2.072

Level: C_INFO

Short Syntax: L2.072 RESET: R Bit rcvd - clear q thru ns= *ns*, mynr= *nr*

Long Syntax: L2.072 RESET: R Bit rcvd - clear q thru ns= *ns*, mynr= *nr*

Description: R bit received from peer

L2.073

Level: C_INFO

Short Syntax: L2.073 *type* Originate Tunnel to peer *peer*

Long Syntax: L2.073 *type* Originate Tunnel to peer *peer*

Description: Originating Tunnel Session

L2.074

Level: C_INFO

Short Syntax: L2.074 Upcall from AAA subsystem, request *status*

Long Syntax: L2.074 Upcall from AAA subsystem, request *status*

Description: Output the result from the upcall from AAA (RADIUS/local-list)

L2.075

Level: C_INFO

Short Syntax: L2.075 Idle Timer Expired - Net *net*

Long Syntax: L2.075 Idle Timer Expired - Net *net*

Description: Demand circuit going down due to idle timeout

L2.076

Level: C_INFO

Short Syntax: L2.076 Fixed Outbound L2 circuit - intiate call Nt *net*

Long Syntax: L2.076 Fixed Outbound L2 circuit - intiate call Nt *net*

Description: fixed circuit running self-test - initiate call

L2.077

Level: C_INFO

Short Syntax: L2.077 Demand Outbound L2 circuit - n_up Nt *net*

Long Syntax: L2.077 Demand Outbound L2 circuit - n_up Nt *net*

Description: Demand circuit running self-test - bring protocols up

---

L2.078

Level: C_INFO

Short Syntax: L2.078 *protocol callid/ peer_callid* EVENT *event*,state= *state*

Long Syntax: L2.078 *protocol callid/ peer_callid* EVENT *event*,state= *state*

Description: PPTP/L2F session event

L2.079

Level: C-INFO

Short Syntax: L2.079 AVAILABLE

Long Syntax: L2.079 AVAILABLE

Description: AVAILABLE

L2.080

Level: P-TRACE

Short Syntax: L2.080 PPTP Send *messageType*, tid= *tunnelid*,cid= *callid*

Long Syntax: L2.080 PPTP Send *messageType*, tid= *tunnelid*,cid= *callid*

Description: PPTP Sends a Call Control Message

L2.081

Level: P-TRACE

Short Syntax: L2.081 PPTP Rcv *messageType*, tid= *tunnelid*,cid= *callid*

Long Syntax: L2.081 PPTP Rcv *messageType*, tid= *tunnelid*,cid= *callid*

Description: PPTP Receives a Call Control Message

L2.082

Level: P-TRACE

Short Syntax: L2.082 ICRQ: serial num= *sn*, bearer

Long Syntax: L2.082 ICRQ: serial num= *sn*, bearer

Description: PPTP Receives Incoming Call Request with values for call serial number ...

L2.083

Level: C-INFO

Short Syntax: L2.083 PPTP Tunnel *id* State Changed -> *newstate*

Long Syntax: L2.083 PPTP Tunnel *id* State Changed -> *newstate*

Description: PPTP Tunnel state changed.

L2.084

Level: C-INFO

Short Syntax: L2.084 PPTP Tunnel *id/ peer_id* EVENT *event*,state= *state*

Long Syntax: L2.084 PPTP Tunnel *id/ peer_id* EVENT *event*,state= *state*

Description: PPTP tunnel event

L2.085

Level: P-TRACE

Short Syntax: L2.085 Rcv PPTP Start-Control-Conn-Req from host *remote_hostname*

Long Syntax: L2.085 Rcv PPTP Start-Control-Conn-Req from host *remote_hostname*

Description: Received PPTP Start control connection request from peer

L2.086

Level: P-TRACE

Short Syntax: L2.086 Rcv PPTP Stop-Control-Conn-Req reason *reason*

Long Syntax: L2.086 Rcv PPTP Stop-Control-Conn-Req reason *reason*

Description: Received PPTP Stop control connection request from peer

L2.087

Level: P-TRACE

Short Syntax: L2.087 Rcv PPTP Echo-Req id *id*

Long Syntax: L2.087 Rcv PPTP Echo-Req id *id*

Description: Received PPTP Echo request

L2.088

Level: C_INFO

Short Syntax: L2.088 Generr in *Function*: *errorcode*

Long Syntax: L2.088 General error in *Function* with error code of *errorcode*

Description: A General Error has occured during Control exchange

L2.089

Level: C_INFO

Short Syntax: L2.089 Conn close in *Function*: *errorcode*

Long Syntax: L2.089 Connection being closed in *Function* due to error: *errorcode*

Description: A socket error has occured and the error is displayed 50 Network is down 51 Network is unreachable 52 Network dropped connection on reset* 53 Software caused connection abort 54 Connection reset by peer 55 No buffer space available 56 Socket is already connected 57 Socket is not connected 58 Can't send after socket shutdown 59 Too many references: can't splice 60 Connection timed out 61 Connection refused 62 Too many levels of symbolic links 63 File name too long 64 Host is down 65 No route to host 66 Directory not empty

---

L2.090

Level: P-TRACE

Short Syntax: L2.090 Rcv PPTP Start-CC-Reply from host *remote_hostname*,result/error *result/ error*

Long Syntax: L2.090 Rcv PPTP Start-CC-Reply from host *remote_hostname*,result/error *result/ error*

Description: Received PPTP Start control connection reply from peer Result Codes: 1 - Successful channel establishment 2 - General error -- Error Code indicates the problem 3 - Command channel already exists; 4 - Requester is not authorized to establish a command channel 5 - The protocol version of the requester is not supported

---

L2.091

Level: P-TRACE

Short Syntax: L2.091 Rcv PPTP OutCall-Reply from host *remote_hostname*,result/error *result/ error*

Long Syntax: L2.091 Rcv PPTP OutCall-Reply from host *remote_hostname*,result/error *result/ error*

Description: Received PPTP Outgoing Call Reply from peer Result Codes: 1 - Successful Connection 2 - General error -- Error Code indicates the problem 3 - No carrier 4 - Busy 5 - No dial tone 5 - Timeout 5 - Do not accept

---

L2.092

Level: C-INFO

Short Syntax: L2.092 PPTP Tunnel *id/ peer_id* Send *msg_type*

Long Syntax: L2.092 PPTP Tunnel *id/ peer_id* Send *msg_type*

Description: PPTP tunnel sent a control message

L2.093

Level: C-INFO

Short Syntax: L2.093 PPTP Session *id/ peer_id* Send *msg_type*

Long Syntax: L2.093 PPTP Session *id/ peer_id* Send *msg_type*

Description: PPTP session sent a control message

L2.094

Level: P-TRACE

Short Syntax: L2.094 Rcv PPTP InCall-Reply from host *remote_hostname*,result/error *result/ error*

Long Syntax: L2.094 Rcv PPTP InCall-Reply from host *remote_hostname*,result/error *result/ error*

Description: Received PPTP Incoming Call Reply from peer Result Codes: 1 - Successful Connection 2 - General error -- Error Code indicates the problem 3 - Do Not Accept Call

---

L2.095

Level: C_TRACE

Short Syntax: L2.095 L2F Call Opened- *Type*,net= *net*

Long Syntax: L2.095 L2F Call Opened- *Type*,net= *net*

Description: L2F Call FSM reached the OPENED state.

L2.096

Level: P_TRACE

Short Syntax: L2.096 SEND client *type*, cid= *cid* tid= *peer-cid*

Long Syntax: L2.096 SEND client *type*, cid= *cid* tid= *peer-cid*

Description: Send the indicated client control message.

L2.097

Level: P_TRACE

Short Syntax: L2.097 RCV *protocol/ L2F_proto*:F= *flags*,L= *length*,Tid= *tunnelid*,Cid= *callid*,NS= *ns*,O= *offset*

Long Syntax: L2.097 RCV *protocol/ L2F_proto*:F= *flags*,L= *length*,Tid= *tunnelid*,Cid= *callid*,NS= *ns*,O= *offset*

Description: Layer-2-Tunneling component received a tunneled packet. It is important to note that some of the displayed fields may NOT have been rcvd - use the "flags" mask to find which ones were rcvd.

L2.098

Level: P_TRACE

Short Syntax: L2.098 SND *protocol/ L2F_proto*:F= *flags*,L= *length*,Tid= *tunnelid*,Cid= *callid*,NS= *ns*,O= *offset*

Long Syntax: L2.098 SND *protocol/ L2F_proto*:F= *flags*,L= *length*,Tid= *tunnelid*,Cid= *callid*,NS= *ns*,O= *offset*

Description: Layer-2-Tunneling component sending a tunneled packet. It is important to note that some of the displayed fields may NOT have been sent - use the ″flags″ mask to find which ones were sent.

L2.099

Level: UE-ERROR

Short Syntax: L2.099 Multi L2 Tunnel - discard nt *netnum*

Long Syntax: L2.099 Multiple L2 tunnel - discard network *netnum*

Description: We do not support Layer-2-tunnels that get tunneled in another layer-2-tunnel.

Cause: Packet being passed to L2 has already been encapsulated with L2 tunnel.

Action: Reconfigure L2 or routing protocols to ensure that traffic is not caught in tunneling recursion.

L2.100

Level: UE-ERROR

Short Syntax: L2.100 Process *type* L2F_CLOSE code= *code*

Long Syntax: L2.100 Process *type* L2F_CLOSE code= *code*

Description: received L2F_CLOSE from peer with a bitmask error code: 0x00000001 Authentication failed 0x00000002 Out of resources 0x00000004 Administrative intervention 0x00000008 User quota exceeded 0x00000010 Protocol error 0x00000020 Unknown user 0x00000040 Incorrect password 0x00000080 PPP configuration incompatible 0x00000100 Wrong multilink PPP destination 0xFF000001 No calls up on tunnel

---

L2.101

Level: UI_ERROR

Short Syntax: L2.101 Err: no remote hostname on Out/Both nt *net*

Long Syntax: L2.101 Error: Outbound/Both net *net* without remote hostname

Description: L2 Net is configured as connection-direction outbound or both and is not configured with a remote hostname.

Action: Reconfigure L2 net as inbound or configure remote hostname.

Chapter 69. LCS virtual Network Interface (LCS)

This chapter describes LCS virtual Network Interface (LCS) messages. For information on message content and how to use the message, refer to the Introduction.

LCS.001

Level: P-TRACE

Short Syntax: LCS.001 brd rcv unkwn typ *packet_type source_Ethernet_address -> destination_Ethernet_address* nt *network*

Long Syntax: LCS.001 broadcast packet received with unknown Ethernet type *packet_type* from host *source_Ethernet_address* to *destination_Ethernet_address* network *network*

Description: A broadcast packet was received with an unknown or unsupported Ethernet type field.

LCS.002

Level: UE-ERROR

Short Syntax: LCS.002 rcv unkwn typ *packet_type source_Ethernet_address -> destination_Ethernet_address* nt *network*

Long Syntax: LCS.002 packet received with unknown Ethernet type field *packet_type* from *source_Ethernet_address* to *destination_Ethernet_address* network *network*

Description: A non-broadcast packet was received with an unknown or unsupported Ethernet type field.

LCS.003

Level: P-TRACE

Short Syntax: LCS.003 brd 802.3 bd ln *actual_length claimed_length source_Ethernet_address -> destination_Ethernet_address* nt *network*

Long Syntax: LCS.003 broadcast packet received with a bad 802.3 length field actual *actual_length* claimed *claimed_length* from *source_Ethernet_address* to *destination_Ethernet_address* network *network*

Description: A broadcast packet was received with a type field that indicated 802.3 but was shorter than data length claimed in the 802.3 header.

LCS.004

Level: UE-ERROR

Short Syntax: LCS.004 802.3 bd ln *actual_length claimed_length source_Ethernet_address -> destination_Ethernet_address* nt *network*

Long Syntax: LCS.004 packet received with a bad 802.3 length field actual *actual_length* claimed *claimed_length* from *source_Ethernet_address* to *destination_Ethernet_address* network *network*

Description: A non-broadcast packet was received with a type field that indicated 802.3 but was shorter than data length claimed in the 802.3 header.

LCS.005

Level: UI_ERROR

Short Syntax: LCS.005 MAC frm typ *mac_frametype* unex from *hardware_address* nt *network*

Long Syntax: LCS.005 MAC frame type *mac_frametype* unexpected from *hardware_address* network *network*

Description: The handler received a frame with an unexpected frame type.

LCS.006

Level: C-INFO

Short Syntax: LCS.006 LLC unk SAP *DSAP source_Ethernet_address -> destination_Ethernet_address* nt *network*

Long Syntax: LCS.006 802.2 LLC packet received with unknown DSAP *DSAP* from host *source_Ethernet_address* to *destination_Ethernet_address* network *network*

Description: An 802.2 LLC packet was received from the network with an inactive (unrecognized) DSAP.

LCS.012

Level: ALWAYS

Short Syntax: LCS.012 LCS Eth nt *network* set to *eth_vers*

Long Syntax: LCS.012 LCS Ethernet network *network* set to Ethernet *eth_vers*

Description: An ARP frame in the indicated format has been received. The LCS net is set to operate using the indicated Ethernet version.

LCS.013

Level: UE-ERROR

Short Syntax: LCS.013 ARP rcv bd hdw type on nt

network rec_hdw_type exp_hdw_type rec_hdw_len exp_hdw_len

Long Syntax: LCS.013 ARP packet received with bad hardware information on network *network*: type received *rec_hdw_type* expected *exp_hdw_type*, length received *rec_hdw_len* expected *exp_hdw_len*

Description: An ARP packet was received in which either the hardware type or hardware length did not match what was expected.

LCS.014

Level: UE-ERROR

Short Syntax: LCS.014 ARP rcv bd prot type on nt *network rec_prot_type exp_prot_type rec_prot_len exp_prot_len*

Long Syntax: LCS.014 ARP packet received with bad protocol information on network *network*: type received *rec_prot_type* expected *exp_prot_type*, length received *rec_prot_len* expected *exp_prot_len*

Description: An ARP packet was received in which either the protocol type or protocol length did not match what was expected.

LCS.015

Level: UE-ERROR

Short Syntax: LCS.015 ARP rcv bd dest addr *dest_address* not *local_addr* on nt *network*

Long Syntax: LCS.015 ARP packet received for destination address *dest_address* not *local_addr* on network *network*

Description: An ARP packet was received in which the destination IP address did not match the local IP address.

LCS.016

Level: UE-ERROR

Short Syntax: LCS.016 ARP rcv bd type *arp_type* on nt *network*

Long Syntax: LCS.016 ARP packet received with unknown type *arp_type* on network *network*

Description: An ARP packet was received which was not a ARP request.

LCS.017

Level: UI-ERROR

Short Syntax: LCS.017 LCS frm rcvd when net not op on nt *network*

Long Syntax: LCS.017 LCS frame received when network *network* is not operational

Description: An LCS frame was received while the network was not enabled for input

---

LCS.018

Level: P-TRACE

Short Syntax: LCS.018 ARP rsp sent on nt *network*

Long Syntax: LCS.018 An ARP response was sent to the host on network *network*

Description: A ARP response was sent.

LCS.019

Level: P-TRACE

Short Syntax: LCS.019 Eth frm rcvd on nt *network*

Long Syntax: LCS.019 An Ethernet frame was received on network *network*

Description: An Ethernet frame was received.

LCS.020

Level: P-TRACE

Short Syntax: LCS.020 Tok frm rcvd on nt *network*

Long Syntax: LCS.020 A Token-Ring frame was received on network *network*

Description: A Token-Ring frame was received.

LCS.021

Level: C-INFO

Short Syntax: LCS.021 nt *network* set to IP *IP_address*

Long Syntax: LCS.021 network *network* set to IP address *IP_address*

Description: The net handler has been set to an IP address.

LCS.022

Level: P-TRACE

Short Syntax: LCS.022 IP frm sent on nt *network*

Long Syntax: LCS.022 An IP frame was sent on network *network*

Description: An IP frame was sent.

LCS.023

Level: P-TRACE

Short Syntax: LCS.023 FDDI frame rcvd on nt *network*

Long Syntax: LCS.023 An FDDI frame was received on network *network*

Description: An FDDI frame was received.

LCS.024

Level: C-INFO

Short Syntax: LCS.024 nt *network* IP addr *IP_address* removed

Long Syntax: LCS.024 network *network* IP address *IP_address* was reset

Description: The net handler's IP address has been reset.

LCS.025

Level: C-INFO

Short Syntax: LCS.025 nt *network* IP *IP_address* was ignored

Long Syntax: LCS.025 network *network* IP address *IP_address* was ignored.

Description: The net handler is currently configured for Bridging or a different local IPV4 address. IP address ignored.

LCS.026

Level: C-INFO

Short Syntax: LCS.026 nt *network* Bdg port *bridge_port* was ignored

Long Syntax: LCS.026 network *network* Bridge port number *bridge_port* was ignored.

Description: The net handler has been configured for IP. Bridge port number ignored.

LCS.028

Level: C-INFO

Short Syntax: LCS.028 nt *network* IPV6 *IPV6_address* was ignored

Long Syntax: LCS.028 network *network* IPV6 address *IPV6_address* was ignored.

Description: The net handler is currently configured for Bridging or a different local IPV6 address. IPV6 address ignored.

LCS.029

Level: C-INFO

Short Syntax: LCS.029 nt *network* set to IPV6 *IPV6_address*

Long Syntax: LCS.029 network *network* IPV6 address *IPV6_address* was reset

Description: The net handler's IPV6 address has been reset.

LCS.030

Level: UE-ERROR

Short Syntax: LCS.030 *file*(*line*) *command* failed: *reason* (nt *network*)

Long Syntax: LCS.030 *file*(*line*) *command* failed: *reason* (network *network*)

Description: Some command failed.

Cause: LCS command received and was rejected.

Action: Examine the reason.

Cause: Invalid packet received from the bridge code.

Action: Examine the reason and check the Host messages.

LCS.031

Level: C-INFO

Short Syntax: LCS.031 LCS net *netnum* rcvd netup from net *lan_netnum* (nt *network*)

Long Syntax: LCS.031 LCS net *netnum* received netup from net *lan_netnum* (network *network*)

Description: LCS net handler received netup from attached downstream LAN.

LCS.032

Level: C-INFO

Short Syntax: LCS.032 LCS net *netnum* MAC addr *macaddr* set (nt *network*)

Long Syntax: LCS.032 LCS net *netnum* MAC address set to *macaddr* (network *network*)

Description: LCS net handler received netup from attached downstream LAN.

LCS.033

Level: CI-ERROR

Short Syntax: LCS.033 LCS frame not sent to LAN - rc *rc* (nt *network*)

Long Syntax: LCS.033 LCS frame not sent to LAN - return code *rc* (network *network*)

Description: The LCS net handler could not send a frame to the downstream LAN.

LCS.034

Level: CI-ERROR

Short Syntax: LCS.034 LCS Error, frame rcvd from wrong LAN, (nt *network*)

Long Syntax: LCS.034 LCS Error, frame received from wrong LAN, (network *network*)

Description: An IP or ARP frame was received from the wrong LAN (3172 Emulation only).

LCS.035

Level: C-INFO

Short Syntax: LCS.035 LCS net *netnum* netdwn by LAN net *lan_netnum* netdwn (nt *network*)

Long Syntax: LCS.035 LCS net *netnum* went netdown because LAN net *lan_netnum* went netdown (nt *network*)

Description: The LCS net handler went netdown because the downstream LAN adapter went netdown.

Chapter 70. Link Aggregation (LAG)

This chapter describes link aggregation (LAG) messages. For information on message content and how to use the message, refer to the Introduction.

LAG.001

Level: UI_ERROR

Short Syntax: LAG.001 nt *net_num* lacp_register failed

Long Syntax: LAG.001 LACP Registration failed on network *net_num*

Description: LACP Registration function failed

LAG.002

Level: C-INFO

Short Syntax: LAG.002 nt *net_num* netup called

Long Syntax: LAG.002 LAG network *net_num* up

Description: LAG network passed self test and is going up

LAG.003

Level: C-INFO

Short Syntax: LAG.003 nt *net_num* netdwn called

Long Syntax: LAG.003 LAG network *net_num* down

Description: LAG network failed self test and is going down

LAG.004

Level: C-INFO

Short Syntax: LAG.004 nt *net_num* discarding LACP frame

Long Syntax: LAG.004 nt *net_num* discarding LACP frame, LAG net is down

Description: LAG network failed self test and is going down

LAG.005

Level: C-INFO

Short Syntax: LAG.005 nt *net_num* dstrbtng frame onto nt *net_num*

Long Syntax: LAG.005 nt *net_num* distributing frame onto network *net_num*

Description: LAG is transmitting a frame onto network/port

LAG.006

Level: CI-ERROR

Short Syntax: LAG.006 nt *net_num* get buffer failed in lacp_transmit

Long Syntax: LAG.006 nt *net_num* failed to allocate a buffer in lacp_transmit

Description: An iorb allocation failed

LAG.007

Level: CI-ERROR

Short Syntax: LAG.007 nt *net_num* get mem failed in lacp_register

Long Syntax: LAG.007 nt *net_num* get memory failed in lacp_register

Description: An iorb allocation failed

LAG.008

Level: CI-ERROR

Short Syntax: LAG.008 nt *net_num* LAG FDDL Registration failed

Long Syntax: LAG.008 nt *net_num* failed to successfully register with FDDL

Description: An iorb allocation failed

LAG.009

Level: CI-ERROR

Short Syntax: LAG.009 nt *net_num* LAG not found

Long Syntax: LAG.009 nt *net_num* LAG not found

Description: LAG not found

LAG.010

Level: CI-ERROR

Short Syntax: LAG.010 nt *net_num* get mem failed in lacp_addlink

Long Syntax: LAG.010 nt *net_num* get memory failed in lacp_addlink

Description: An iorb allocation failed

LAG.011

Level: CI-ERROR

Short Syntax: LAG.011 nt *net_num* Link not found

Long Syntax: LAG.011 nt *net_num* Link not found

Description: Link not found

LAG.012

Level: CI-ERROR

Short Syntax: LAG.012 nt *net_num1* Non-LACP msg rcvd on link *net_num2*. Discarding

Long Syntax: LAG.012 nt *net_num1* received Non-LACP message on link *net_num2*. Discarding

Description: Non-LACP msg received on a LAG link

LAG.013

Level: C-INFO

Short Syntax: LAG.013 nt *net_num1* rcvd LACP frame on Non-agg-ing link *net_num2*. Discarding

Long Syntax: LAG.013 nt *net_num1* received LACP frame on Non-aggregating link *net_num2*. Discarding

Description: LACP msg received on a LAG link in Non-aggregating state

---

LAG.014

Level: C-INFO

Short Syntax: LAG.014 nt *net_num1* LACP Tt timer popped on link *net_num2*

Long Syntax: LAG.014 nt *net_num1* LACP Tt timer popped on link *net_num2*

Description: LACP Tt timer popped

LAG.015

Level: C-INFO

Short Syntax: LAG.015 nt *net_num1* LACP Tk timer popped on link *net_num2*

Long Syntax: LAG.015 nt *net_num1* LACP Tk timer popped on link *net_num2*

Description: LACP Tk timer popped

LAG.016

Level: CI-ERROR

Short Syntax: LAG.016 nt *net_num1* LACP process event re-entered

Long Syntax: LAG.016 nt *net_num1* LACP process event re-entered

Description: LACP process_event() re-entered

LAG.017

Level: C-INFO

Short Syntax: LAG.017 nt *net_num1* LACP match check failed

Long Syntax: LAG.017 nt *net_num1* LACP match check failed

Description: LACP LACP LAG ID match check failed\r\n

LAG.018

Level: CI-ERROR

Short Syntax: LAG.018 nt *net_num1* lacp_distribute, desired link not found

Long Syntax: LAG.018 nt *net_num1* lacp_distribute, desired link not found

Description: LACP distribution could not find matching link

LAG.019

Level: DEBUG

Short Syntax: LAG.019 nt *net_num1* Snding in frm to eth_bdg_in()

Long Syntax: LAG.019 nt *net_num1* Sending inbound frame to eth_bdg_in()

Description: Indicates that LAG is passing a data frame to the ethernet bridge

---

LAG.020

Level: CI-ERROR

Short Syntax: LAG.020 nt *net_num1* Snding in frm to eth_in()

Long Syntax: LAG.020 nt *net_num1* Sending inbound frame to eth_in()

Description: Indicates that LAG is passing a data frame to the ethernet in routine

---

LAG.021

Level: DEBUG

Short Syntax: LAG.021 nt *net_num1* link *net_num2* FDDL LAG Add called

Long Syntax: LAG.021 nt *net_num1* link *net_num2* FDDL LAG Add called

Description: The FDDL LAG Add message was sent

LAG.022

Level: DEBUG

Short Syntax: LAG.022 nt *net_num1* link *net_num2* FDDL LAG Delete called

Long Syntax: LAG.022 nt *net_num1* link *net_num2* FDDL LAG Delete called

Description: The FDDL LAG Delete message was sent

LAG.023

Level: DEBUG

Short Syntax: LAG.023 nt *net_num1* FDDL LAG Update called

Long Syntax: LAG.023 nt *net_num1* FDDL LAG Update called

Description: The FDDL LAG Update message was sent

LAG.024

Level: C-INFO

Short Syntax: LAG.024 nt *net_num* netdwn called link speed mismatch

Long Syntax: LAG.024 LAG network *net_num* down because of link speed mismatch

Description: LAG links are not all running at same link speed

---

LAG.025

Level: CI-ERROR

Short Syntax: LAG.025 nt *net_num1* HDX link on LAG *net_num2*, intf

Long Syntax: LAG.025 nt *net_num1* Half-Duplex link on LAG *net_num2*, interface

Description: This indicates that named link in the LAG is not full-duplex

---

LAG.026

Level: CI-ERROR

Short Syntax: LAG.026 nt *net_num1* Not used

Long Syntax: LAG.026 nt *net_num1* Not used

Description: Not used

LAG.027

Level: CI-ERROR

Short Syntax: LAG.027 nt *net_num1* Not used

Long Syntax: LAG.027 nt *net_num1* Not used

Description: Not used

LAG.028

Level: CI-ERROR

Short Syntax: LAG.028 nt *net_num1* Not used

Long Syntax: LAG.028 nt *net_num1* Not used

Description: Not used

LAG.029

Level: CI-ERROR

Short Syntax: LAG.029 nt *net_num1* Not used

Long Syntax: LAG.029 nt *net_num1* Not used

Description: Not used

LAG.030

Level: DEBUG

Short Syntax: LAG.030 nt *net_num1* link *net_num2* E1 + S1 --> A3/S2

Long Syntax: LAG.030 nt *net_num1* link *net_num2* E1 + S1 --> A3/S2

Description: LACP state message

LAG.031

Level: DEBUG

Short Syntax: LAG.031 nt *net_num1* link *net_num2* Going into INIT_S state

Long Syntax: LAG.031 nt *net_num1* link *net_num2* Going into INIT_S state

Description: LACP state message

LAG.032

Level: DEBUG

Short Syntax: LAG.032 nt *net_num1* link *net_num2* (S2 || S3) + E7 --> A5/A1/S1

Long Syntax: LAG.032 nt *net_num1* link *net_num2* (S2 || S3) + E7 --> A5/A1/S1

Description: LACP state message

LAG.033

Level: DEBUG

Short Syntax: LAG.033 nt *net_num1* link *net_num2* ((S2 + E3) --> A4/S3) || (S2 + E4 --> S2))

Long Syntax: LAG.033 nt *net_num1* link *net_num2* ((S2 + E3) --> A4/S3) || (S2 + E4 --> S2))

Description: LACP state message

LAG.034

Level: DEBUG

Short Syntax: LAG.034 nt *net_num1* link *net_num2* Going into COLLECTING_S state

Long Syntax: LAG.034 nt *net_num1* link *net_num2* Going into COLLECTING_S state

Description: LACP state message

LAG.035

Level: DEBUG

Short Syntax: LAG.035 nt *net_num1* link *net_num2* Match check failed, Expected= *mac1*, Received= *mac2*

Long Syntax: LAG.035 nt *net_num1* link *net_num2* Match check failed, Expected= *mac1*, Received= *mac2*

Description: LACP LAG ID match check failed

LAG.036

Level: DEBUG

Short Syntax: LAG.036 nt *net_num1* link *net_num2* (S3 + E5) --> A2/S3

Long Syntax: LAG.036 nt *net_num1* link *net_num2* (S3 + E5) --> A2/S3

Description: LACP state message

LAG.037

Level: DEBUG

Short Syntax: LAG.037 nt *net_num1* link *net_num2* (S3 + E6) --> A6/A12/A10/S4

Long Syntax: LAG.037 nt *net_num1* link *net_num2* (S3 + E6) --> A6/A12/A10/S4

Description: LACP state message

LAG.038

Level: DEBUG

Short Syntax: LAG.038 nt *net_num1* link *net_num2* Going into FLUSHING_ADD_S state

Long Syntax: LAG.038 nt *net_num1* link *net_num2* Going into FLUSHING_ADD_S state

Description: LACP state message

LAG.039

Level: DEBUG

Short Syntax: LAG.039 nt *net_num1* link *net_num2* (S3 + E9 --> S2)

Long Syntax: LAG.039 nt *net_num1* link *net_num2* (S3 + E9 --> S2)

Description: LACP state message

LAG.040

Level: DEBUG

Short Syntax: LAG.040 nt *net_num1* link *net_num2* (S4 + E7 --> A12/A1/A11/S1

Long Syntax: LAG.040 nt *net_num1* link *net_num2* (S4 + E7 --> A12/A1/A11/S1

Description: LACP state message

LAG.041

Level: DEBUG

Short Syntax: LAG.041 nt *net_num1* link *net_num2* Going into UP_S state

Long Syntax: LAG.041 nt *net_num1* link *net_num2* Going into UP_S state

Description: LACP state message

LAG.042

Level: DEBUG

Short Syntax: LAG.042 nt *net_num1* link *net_num2* (S4 + E5 --> A3/A12/S2)

Long Syntax: LAG.042 nt *net_num1* link *net_num2* (S4 + E5 --> A3/A12/S2)

Description: LACP state message

LAG.043

Level: DEBUG

Short Syntax: LAG.043 nt *net_num1* link *net_num2* ((S3||S4||55||S6) + E8 --> A7/(S3||S4||S5||S6))

Long Syntax: LAG.043 nt *net_num1* link *net_num2* ((S3||S4||55||S6) + E8 --> A7/(S3||S4||S5||S6))

Description: LACP state message

LAG.044

Level: DEBUG

Short Syntax: LAG.044 nt *net_num1* link *net_num2* (S4 + E9) --> A2/S6

Long Syntax: LAG.044 nt *net_num1* link *net_num2* (S4 + E9) --> A2/S6

Description: LACP state message

LAG.045

Level: DEBUG

Short Syntax: LAG.045 nt *net_num1* link *net_num2* (S5 + E9) --> A1/S1

Long Syntax: LAG.045 nt *net_num1* link *net_num2* (S5 + E9) --> A1/S1

Description: LACP state message

LAG.046

Level: DEBUG

Short Syntax: LAG.046 nt *net_num1* link *net_num2* (S6 + E7) --> A6/A12/A11/S5

Long Syntax: LAG.046 nt *net_num1* link *net_num2* (S6 + E7) --> A6/A12/A11/S5

Description: LACP state message

LAG.047

Level: DEBUG

Short Syntax: LAG.047 nt *net_num1* link *net_num2* Going into FLUSHING_REMOVE_S state

Long Syntax: LAG.047 nt *net_num1* link *net_num2* Going into FLUSHING_REMOVE_S state

Description: LACP state message

LAG.048

Level: DEBUG

Short Syntax: LAG.048 nt *net_num1* link *net_num2* (S6 + E5 --> A3/A11/A12/S2)

Long Syntax: LAG.048 nt *net_num1* link *net_num2* (S6 + E5 --> A3/A11/A12/S2)

Description: LACP state message

LAG.049

Level: DEBUG

Short Syntax: LAG.049 nt *net_num1* link *net_num2* Going into DOWN_S state

Long Syntax: LAG.049 nt *net_num1* link *net_num2* Going into DOWN_S state

Description: LACP state message

Panic plaginitmemfail

Short Syntax: plaginitmemfail: lag tbl cannot get memory.

Description: The LAG net Initialization could net allocate memory

Action: Contact customer service

Panic plaglinkbadadd

Short Syntax: plaglinkbadadd: lag link could not be added.

Description: The LAG net Link Initialization failed.

Action: Contact customer service

Chapter 71. Logical Link Control (LLC) ELS Messages

This chapter describes Logical Link Control (LLC) ELS Messages messages. For information on message content and how to use the message, refer to the Introduction.

LLC.001

Level: C-TRACE

Short Syntax: LLC.001 Sent *frame_type*, *src_mac*-> *dst_mac*, *rif* saps *src_sap*-> *dst_sap*, dlci *dlci* nt *network*

Long Syntax: LLC.001 Sent *frame_type*, *src_mac*-> *dst_mac*, *rif* saps *src_sap*-> *dst_sap*, dlci *dlci* network *network*

Description: LLC is sending a frame. Possible frame types are: SABME_C0 or SABME_C1 (Set Asynchronous Balanced Mode Extended), DM_R0 or DM_R1 (Disconnected Mode), DISC_C0 or DISC_C1 (Disconnect), RR_C0 or RR_C1 or RR_R0 or RR_R1 (Receiver Ready), RNR_C0 or RNR_C1 or RNR_R0 or RNR_R1 (Receiver Not Ready), REJ_C0 or REJ_C1 or REJ_R0 or REJ_R1 (Reject), UA_R0 or UA_R1 (Unnumbered Acknowledgement), FRMR_R0 or FRMR_R1 (Frame Reject), and I_C0 or I_C1 or I_R0 or I_R1 (Information Frame). The abbrevation suffixes are C0 (command, poll bit off), C1 (command, poll bit on), R0 (response, final bit off), and R1 (response, final bit on).

LLC.002

Level: C-TRACE

Short Syntax: LLC.002 ev= *llc_event* in st= *llc_state*, *llc2_connection*, dlci *dlci*, nt *network*

Long Syntax: LLC.002 event= *llc_event* in state= *llc_state*, *llc2_connection*, dlci *dlci*, network *network*

Description: An event occurred on an llc2 connection. The LLC2 FSM (Finite State Machine) has been called to process the event. The LLC2 connection is uniquely identified by the combination destination MAC address, source MAC address, destination sap, and source sap on a particular network. The possible events are: SET_ABME (user request to connect to remote), SET_ADM (user request to disconnect from remote), SEND_BTU (user request to send data), FLOW_REQ_ON (user request to turn off local busy condition), FLOW_REQ_OFF (user request to turn on local busy condition), T1_EXP (T1 timer expiration), T2_EXP (T2 timer expiration), Ti_EXP (Ti timer epiration), OS_I_C0 or OS_I_C1 or OS_I_R0 or OS_I_R1 (Ns on I-frame is out of sequence), I_C0 or I_C1 or I_R0 or I_R1 (valid I-frame received), RR_C0 or RR_C1 or RR_R0 or RR_R1 (RR frame received), RNR_C0 or RNR_C1 or RNR_R0 or RNR_R1 (RNR frame received), REJ_C0 or REJ_C1 or REJ_R0 or REJ_R1 (REJ frame received), UA_R0 or UA_R1 (UA frame received), DISC_C0 or DISC_C1 (DISC frame received), DM_R0 or DM_R1 (DM frame received), FRMR_R0 or FRMR_R1 (FRMR frame received), BAD_FRAME_0 or BAD_FRAME_1 (received frame will generate FRMR), SABME_C0 or SABME_C1 (SABME frame received), and SEND_I_POLL (Sending I frame with Poll bit on). The abbrevation suffixes are C0 (command, poll bit off), C1 (command, poll bit on), R0 (response, final bit off), and R1 (response, final bit on).

LLC.003

Level: C-INFO

Short Syntax: LLC.003 *llc_state*-> *llc_state*, *llc2_connection*, dlci *dlci*, nt *network*

Long Syntax: LLC.003 *llc_state* to *llc_state*, *llc2_connection*, dlci *dlci*, network *network*

Description: There is LLC2 state change. The possible states are: DISCONNECTED (initial state), LINK_OPENING (link establishment in progress), DISCONNECTING (DISC sent, awaiting DM), FRMR_SENT (frmr sent), LINK_OPENED (normal state), LOCAL_BUSY (local is busy), REJECTION (remote sent an out of sequence frame), CHECKPOINTING (poll sent, awaiting response sending of data supended), CKPT_LB (combination state), CKPT_REJ (combination state), RESETTING (awaiting user response to reset), REMOTE_BUSY (remote is busy), LB_RB (combination state), REJ_LB (combination state), REJ_RB (combination state), CKPT_REJ_LB (combination state), CKPT_CLR (clearing from CKPT_LB state), CKPT_REJ_CLR (clearing from CKPT_REJ_LB state), REJ_LB_RB (combination state), FRMR_RECEIVED (received frmr). The abbreviations above are CKPT=CHECKPOINTING, CLR=CLEARING, LB=LOCAL BUSY, RB=REMOTE BUSY, and REJ=REJECTION.

LLC.004

Level: C-INFO

Short Syntax: LLC.004 Up evt *user_event* args *user_value/ event_reason* on *llc2-conn*

Long Syntax: LLC.004 Upcall user event *user_event user_value event_reason* on *llc2-conn*

Description: LLC2 event upcall is occuring. Some of

the arguments on the upcall are shown. User Cookie is meaningful to the router software running over the LLC subsystem. Event reason sometimes further specifies the event. The possible upcall events are: CONN_IND (cookie=session, reason=none), CONN_IND_PASS (cookie=sap, reason=none), CONN_CONFIRM (cookie=session, reason=none), DISC_IND (cookie=session, possible reasons: local term (disconnecting), remote term, conn refused, local term (disconnected)), RESET_IND (cookie=session, possible reasons: local reset, remote reset, frmr rcvd, frmr sent), RESET_CONF (cookie=session, reason=none), FLOW_IND (cookie=session, possible reasons: flow off, flow on), and DISC_CONFIRM (cookie=session, reason=none).

LLC.005

Level: C-INFO

Short Syntax: LLC.005 prim *user_primitive* sap *SAP_value* on nt *network*

Long Syntax: LLC.005 user primitive *user_primitive* sap *SAP_value* on network *network*

Description: A sap-releated LLC user-primitive was called. The possible SAP primitives are: OPEN_SAP, CLOSE_SAP, CLOSE_SAP_FORCED, MODIFY_SAP, OPEN_STATION, and UNITDATA.

LLC.006

Level: C-INFO

Short Syntax: LLC.006 prim *user_primitive*, *llc2_connection*, dlci *dlci*, nt *network*

Long Syntax: LLC.006 primitive *user_primitive*, *llc2_connection*, dlci *dlci*, network *network*

Description: A llc2 connection non-data user-primitive was called. The possible primitives are: CLOSE_STATION, CLOSE_STATION_FORCED, CONNECT_REQUEST, CONNECT_RESPONSE, DISCONNECT_REQUEST, RESET_REQUEST, RESET_RESPONSE, FLOW_REQ.

LLC.007

Level: C-TRACE

Short Syntax: LLC.007 data prim, *llc2_connection*, dlci *dlci*, nt *network*

Long Syntax: LLC.007 data primitive, *llc2_connection*, dlci *dlci*, network *network*

Description: A DATA_REQUEST data primitive was called. DATA_REQUEST passes the data in buffer memory.

LLC.008

Level: C-TRACE

Short Syntax: LLC.008 data prim, *llc2_connection*, dlci *dlci*, nt *network*

Long Syntax: LLC.008 data primitive, *llc2_connection*, dlci *dlci*, network *network*

Description: A DATA_LOCAL data primitive was called. DATA_LOCAL passes the data in data memory.

LLC.009

Level: C-TRACE

Short Syntax: LLC.009 unitdata prim, sap *SAP_value*, dlci *dlci* nt *network*

Long Syntax: LLC.009 unitdata primitive, sap *SAP_value* dlci *dlci* network *network*

Description: A UNITDATA llc1 data primitive was called.

LLC.010

Level: UI-ERROR

Short Syntax: LLC.010 out q too big, *llc2_connection*, dlci *dlci*, nt *network*

Long Syntax: LLC.010 outboudn queue too big, *llc2_connection*, dlci *dlci*, network *network*

Description: The oubound queue has grown grossly large. The llc2 connection is being automatically terminated.

Cause: LLC application is not responding to flow control.

Action: Contact customer service.

LLC.011

Level: UI-ERROR

Short Syntax: LLC.011 No buf to dup I-frame, *llc2_connection*, dlci *dlci*, nt *network*

Long Syntax: LLC.011 No buffer available to duplicate I-frame, *llc2_connection*, dlci *dlci*, network *network*

Description: No buffer available to duplicate I-frame.

Cause: Severe packet buffer shortage.

Action: Check memory statistics in GWCON to verify packet buffer level. Reduce buffer usage of other router software. Reduce buffer usage by reducing LLC conections, by changing LLC configuration, especially making sure that LLC Transmit and Receive windows are normal sizes.

LLC.012

Level: UI-ERROR

Short Syntax: LLC.012 No mem to dup I-frame, *llc2_connection*, dlci *dlci*, nt *network*

Long Syntax: LLC.012 No memory available to duplicate I-frame, *llc2_connection*, dlci *dlci*, network *network*

Description: No memory to duplicate I-frame.

Cause: Memory shortage.

Action: Reduce memory usage by reducing tables in other software. Reduce memory by reducing LLC conections, by changing LLC configuration, especially making sure that LLC Transmit and Receive windows are normal sizes.

LLC.013

Level: UI-ERROR

Short Syntax: LLC.013 No buf for LLC frame, *llc2_connection*, dlci *dlci*, nt *network*

Long Syntax: LLC.013 No buffer for LLC frame, *llc2_connection*, dlci *dlci*, network *network*

Description: A buffer could not be obtained to to build an LLC Supervisory or Unnumbered frame. No loss of data integrity has occurred yet, but unless buffers for this purpose become available within a few seconds, the other end of the LLC2 connection will most likely terminate this LLC connection as part of the normal LLC2 protocol.

Cause: Severe packet buffer shortage.

Action: Check memory statistics in GWCON to verify packet buffer level.

LLC.014

Level: UI-ERROR

Short Syntax: LLC.014 fr type inv, *llc2_connection*, dlci *dlci*, nt *network*

Long Syntax: LLC.014 frame type invalid, *llc2_connection*, dlci *dlci*, network *network*

Description: The frame type the LLC is trying to build is invalid.

Cause: Hardware failure or software bug.

Action: Contact customer service.

LLC.015

Level: UI-ERROR

Short Syntax: LLC.015 Inv LLC2 ev *event_code_number*

Long Syntax: LLC.015 Invalid LLC2 FSM event *event_code_number*

Description: The LLC2 Finite State Machine (FSM) was called with an event that was out of range.

Cause: Hardware failure or software bug.

Action: Contact customer service.

LLC.016

Level: UI-ERROR

Short Syntax: LLC.016 inv nt typ *network_type* on nt *network*

Long Syntax: LLC.016 invalid network type *network_type* on network *network*

Description: An OPEN SAP operation was tried on a network type that LLC does not support. Network types Token-Ring, Ethernet, and FDDI are supported.

Cause: Software bug.

Action: Contact customer service.

LLC.017

Level: UI-ERROR

Short Syntax: LLC.017 dup sap *SAP_value* on nt *network*

Long Syntax: LLC.017 duplicate sap *SAP_value* on network *network*

Description: A OPEN SAP operation was tried on a sap that has already been opened.

Cause: Software bug.

Action: Contact customer service.

LLC.018

Level: UI-ERROR

Short Syntax: LLC.018 No mem for sap blk on nt *network*

Long Syntax: LLC.018 No memory for SAP control block on network *network*

Description: Unable to allocate memory for SAP control block.

Cause: Severe shortage of memory.

Action: Reduce table sizes in other protocols, use system with less protocols, expand memory in router.

LLC.019

Level: UI-ERROR

Short Syntax: LLC.019 No mem for stn blk on nt *network*

Long Syntax: LLC.019 No memory for station control block on network *network*

Description: Unable to allocate memory for station control block.

Cause: Severe shortage of memory.

Action: Reduce table sizes in other protocols, use system with less protocols, expand memory in router. Reduce number of LLC2 connections.

LLC.020

Level: U-INFO

Short Syntax: LLC.020 UI frm drp *llc2_connection*, dlci *dlci*, nt *network*

Long Syntax: LLC.020 UI frame dropped, *llc2_connection*, dlci *dlci*, network *network*

Description: UI frame refused by the local application within the router.

Cause: The frame was not the type the local application wanted to handle.

Action: None.

LLC.021

Level: U-INFO

Short Syntax: LLC.021 TST frm refused *llc2_connection*, dlci *dlci*, nt *network*

Long Syntax: LLC.021 TEST frame refused, *llc2_connection*, dlci *dlci*, network *network*

Description: TEST frame refused by the local application within the router. The frame is passed on to the bridge code, etc.

Cause: The frame was not the type the local application wanted to handle.

Action: None.

LLC.022

Level: U-INFO

Short Syntax: LLC.022 XID frm refused *llc2_connection*, dlci *dlci*, nt *network*

Long Syntax: LLC.022 XID frame refused, *llc2_connection*, dlci *dlci*, network *network*

Description: XID frame refused by the local application within the router. The frame is passed on to the bridge code, etc.

Cause: The frame was not the type the local application wanted to handle.

Action: None.

LLC.023

Level: C-INFO

Short Syntax: LLC.023 Upcall frm *frame_type*, *src_mac-> dst_mac*, *rif* saps *src_sap-> dst_sap*, dlci *dlci*, nt *network*

Long Syntax: LLC.023 Upcall frame *frame_type*, *src_mac-> dst_mac*, *rif* saps *src_sap-> dst_sap*, dlci *dlci*, network *network*

Description: LLC makes an upcall providing the LLC with a unitdata frame. The possible unidata frames are: TEST_C0 or TEST_C1 or TEST_R0 or TEST_R1 (TEST frame), XID_C0 or XID_C1 or XID_R0 or XID_R1 (Exchange Identification frame), UI_C0 or UI_R0 (Unnumbered Information). The abbrevation suffixes are C0 (command, poll bit off), C1 (command, poll bit on), R0 (response, final bit off), and R1 (response, final bit on).

LLC.024

Level: UI-ERROR

Short Syntax: LLC.024 llc2 out drp, rsn *reason_code*, *llc2_connection*, dlci *dlci*, nt *network*

Long Syntax: LLC.024 llc2 outbound frame dropped, reason *reason_code*, *llc2_connection*, dlci *dlci*, network *network*

Description: The sending of an LLC2 related outbound frame failed. The reason_code is the internal error code for the failure.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

LLC.025

Level: UI-ERROR

Short Syntax: LLC.025 *frame_type* out frm drp, rsn *reason_code*, *llc2_connection*, dlci *dlci*, nt *network*

Long Syntax: LLC.025 *frame_type* outbound frame

dropped, reason *reason_code*, *llc2_connection*, dlci *dlci*, network *network*

Description: The sending of the user's UNITDATA or an LLC-generated XID or TEST response outbound frame failed. The possible frame test are: TEST_C0 or TEST_C1 or TEST_R0 or TEST_R1 (TEST frame), XID_C0 or XID_C1 or XID_R0 or XID_R1 (Exchange Identification frame), UI_C0 or UI_R0 (Unnumbered Information frame), and unexpected (not one of the above types). The abbrevation suffixes are C0 (command, poll bit off), C1 (command, poll bit on), R0 (response, final bit off), and R1 (response, final bit on).

Cause: See LLC.024.

Action: See LLC.024

LLC.026

Level: UI-ERROR

Short Syntax: LLC.026 No mem for cfg blk on nt *network*

Long Syntax: LLC.026 No memory for LLC CONF BLOCK on network *network*

Description: Unable to allocate memory for an LLC_CONF_BLOCK at initialization time. LLC configuration defaults are used.

Cause: Severe shortage of memory.

Action: Reduce table sizes in other protocols, use system with less protocols, expand memory in router.

LLC.027

Level: U-INFO

Short Syntax: LLC.027 Read LLC Cfg for nt *network*

Long Syntax: LLC.027 Read LLC Configuration record for network *network*

Description: LLC Configuration record read for this network. This only occurs at initialization time. The values in the LLC configuration record are used as default value on the network.

LLC.028

Level: U-INFO

Short Syntax: LLC.028 Inv acc *access_priority* for nt *network*

Long Syntax: LLC.028 Inv access priority *access_priority* for network *network*

Description: access_priority, on a network that that is not a token ring must be zero because it is not used.

Cause: As devices are deleted and added, it is possible for one of the LLC config records to contain a non-zero access priority on a non-Token-Ring LAN interface.

Action: None. You may reconfigure the LLC config on this network to avoid getting this message.

LLC.029

Level: UI-ERROR

Short Syntax: LLC.029 Inv acc *access_priority* for nt *network*

Long Syntax: LLC.029 Inv acc *access_priority* for network *network*

Description: The access priority is greater than 7. A default of 0 is used.

Cause: Configuration memory corruption.

Action: Reconfigure the LLC on this network to avoid getting this message.

LLC.030

Level: C-TRACE

Short Syntax: LLC.030 Inv hw type *hardware_type* in cfg for nt *network*

Long Syntax: LLC.030 Invalid hardware type *hardware_type* for network *network*

Description: An LLC config record exists for an interface that does not have a LAN hardware type.

Cause: As devices are deleted and added, it is possible for one of the LLC config records to contain an interface that is no longer a LAN interface.

Action: None. Situation is not harmful.

LLC.031

Level: C-TRACE

Short Syntax: LLC.031 Inv int *interface_number* in cfg

Long Syntax: LLC.031 Invalid interface *interface_number* in config

Description: An LLC config record exists for an interface that does not exist.

Cause: As devices are deleted and added, it is possible for one of the LLC config records to contain an invalid interface number.

Action: None. Situation is not harmful.

LLC.032

Level: C-INFO

Short Syntax: LLC.032 Sent *frame_type*, *src_mac*-> *dst_mac*, *rif* saps *src_sap*-> *dst_sap*, dlci *dlci*,nt *network*

Long Syntax: LLC.032 Sent *frame_type*, *src_mac*-> *dst_mac*, *rif* saps *src_sap*-> *dst_sap*, dlci *dlci*, network *network*

Description: LLC user is sending a frame, or LLC itself is sending a TEST or XID response frame. The possible frame types are: TEST_C0 or TEST_C1 or TEST_R0 or TEST_R1 (TEST frame), XID_C0 or XID_C1 or XID_R0 or XID_R1 (Exchange Identification frame), UI_C0 or UI_R0 (Unnumbered Information frame). The abbrevation suffixes are C0 (command, poll bit off), C1 (command, poll bit on), R0 (response, final bit off), and R1 (response, final bit on).

LLC.033

Level: C-INFO

Short Syntax: LLC.033 frm to LLC, frm *frame_type*, *src_mac*-> *dst_mac*, *rif* saps *src_sap*-> *dst_sap*, dlci *dlci*, nt *network*

Long Syntax: LLC.033 frm to LLC, frm *frame_type*, *src_mac*-> *dst_mac*, *rif* saps *src_sap*-> *dst_sap*, dlci *dlci*, network *network*

Description: LLC subsystem itself is responding to a TEST or XID frame. The possible frame types are: TEST_C0 or TEST_C1 (TEST frame), and XID_C0 or XID_C1 (Exchange Identification frame). The abbrevation suffixes are: C0=(command, poll bit off), and C1=(command, poll bit on),

---

LLC.034

Level: C-INFO

Short Syntax: LLC.034 LLC loopback invoked *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, nt *network*

Long Syntax: LLC.034 LLC loopback invoked, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, network *network*

Description: Loopback processing has been invoked to route frames within the router.

LLC.035

Level: C-INFO

Short Syntax: LLC.035 Dest SCB not found *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, nt *network*

Long Syntax: LLC.035 Destination SCB not found, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, network *network*

Description: Processing has not found the Session Control Block for the destination.

Cause: The destination application may not have done an open station. The destination application may have gone down.

Action: None.

LLC.036

Level: C-INFO

Short Syntax: LLC.036 Loopback CONNECT, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, nt *src_net*, nt *dst_net*

Long Syntax: LLC.036 Loopback CONNECT, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, network *src_net*, network *dst_net*

Description: Connect in is being sent from origin net to destination net.

LLC.037

Level: C-INFO

Short Syntax: LLC.037 Loopback CONNECT Rsp, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, nt *src_net*, nt *dst_net*

Long Syntax: LLC.037 Loopback CONNECT Response, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, network *src_net*, nt *dst_net*

Description: Connect response is being sent from origin net to destination net.

LLC.038

Level: C-INFO

Short Syntax: LLC.038 Loopback DISCONNECT, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, nt *src_net*, nt *dst_net*

Long Syntax: LLC.038 Loopback DISCONNECT, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, nt *src_net*, nt *dst_net*

Description: Disconnect is being sent from origin net to destination net.

LLC.039

Level: C-INFO

Short Syntax: LLC.039 Loopback DISCONNECT Rsp, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, nt *src_net*, nt *dst_net*

Long Syntax: LLC.039 Loopback DISCONNECT Response, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, nt *src_net*, nt *dst_net*

Description: Disconnect response is being sent from origin net to destination net.

LLC.040

Level: C-INFO

Short Syntax: LLC.040 Loopback RESET, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, nt *src_net*, nt *dst_net*

Long Syntax: LLC.040 Loopback RESET, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, nt *src_net*, nt *dst_net*

Description: Reset is being sent from origin net to destination net.

LLC.041

Level: C-INFO

Short Syntax: LLC.041 Loopback RESET Rsp, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, nt *src_net*, nt *dst_net*

Long Syntax: LLC.041 Loopback RESET Response, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, nt *src_net*, nt *dst_net*

Description: Reset response is being sent from origin net to destination net.

LLC.042

Level: C-INFO

Short Syntax: LLC.042 Loopback FLOW ON, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, nt *src_net*, nt *dst_net*

Long Syntax: LLC.042 Loopback FLOW ON, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, network *src_net*, nt *dst_net*

Description: Flow on is being sent from origin net to destination net.

LLC.043

Level: C-INFO

Short Syntax: LLC.043 Loopback FLOW OFF, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, nt *src_net*, nt *dst_net*

Long Syntax: LLC.043 Loopback FLOW OFF, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, network *src_net*, nt *dst_net*

Description: Flow off is being sent from origin net to destination net.

LLC.044

Level: C-INFO

Short Syntax: LLC.044 Loopback FLOW OFF Data, st *state,*, busy *busy_flag,*, qnum *qnum,*, nt *src_net*

Long Syntax: LLC.044 Loopback FLOW OFF Data, state *state,*, busy *busy_flag,*, tr_queue_num *qnum,*, network *src_net*

Description: Flow off data to get the exact status of the application sending flow off

---

LLC.045

Level: C-INFO

Short Syntax: LLC.045 Loopback FLOW ON Data, st *state,*, busy *busy_flag,*, qnum *qnum,*, nt *src_net*

Long Syntax: LLC.045 Loopback FLOW ON Data, state *state,*, busy *busy_flag,*, tr_queue_num *qnum,*, network *src_net*

Description: Flow on data to get the exact status of the application sending flow on

---

LLC.046

Level: C-INFO

Short Syntax: LLC.046 LLC Busy No Resource, st *state,*, busy *busy_flag,*, num *qnum,*, nt *src_net*

Long Syntax: LLC.046 LLC Busy No Resource, state *state,*, busy *busy_flag,*, num *qnum,*, network *src_net*

Description: Exceeded Max IORB in queue, data to get the exact status of the application

---

LLC.047

Level: C-INFO

Short Syntax: LLC.047 Loopback Net Not Found, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, nt *network*

Long Syntax: LLC.047 Loopback Net Not Found, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, network *network*

Description: Net structure for the loopback destination net has not been found.

LLC.048

Level: C-INFO

Short Syntax: LLC.048 Loopback Dest Matching SAP Not Found, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, nt *network*

Long Syntax: LLC.048 Loopback Dest Matching SAP Not Found, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, network *network*

Description: Sap structure for the loopback destination sap has not been found.

LLC.049

Level: C-INFO

Short Syntax: LLC.049 Loopback Connect Data, st *state,*, dest_st *dest_st,*, nt *src_net,*, nt *dst_net*

Long Syntax: LLC.049 Loopback Connect Data, state *state,*, dest_st *dest_st,*, network *src_net,*, network *dst_net*

Description: Loopback Connect Data

LLC.050

Level: C-INFO

Short Syntax: LLC.050 Loopback Connect Failed, st *state,*, dest_st *dest_st,*, nt *src_net,*, nt *dst_net*

Long Syntax: LLC.050 Loopback Connect Failed, state *state,*, dest_st *dest_st,*, network *src_net,*, network *dst_net*

Description: Loopback Connect Failed

LLC.051

Level: C-INFO

Short Syntax: LLC.051 Loopback Send Failed, st *state,*, dest_st *dest_st,*, nt *src_net,*, nt *dst_net*

Long Syntax: LLC.051 Loopback Send Failed, state *state,*, dest_st *dest_st,*, network *src_net,*, network *dst_net*

Description: Loopback Send Failed

LLC.053

Level: C-INFO

Short Syntax: LLC.053 LLC Config Block not found, sap *src_sap*, nt *network*

Long Syntax: LLC.053 LLC Config Block not found, sap *src_sap* network *network*

Description: Open station Failed for the described sap

LLC.054

Level: C-INFO

Short Syntax: LLC.054 LLC Client Registered Fail, sap *src_sap*, nt *network*

Long Syntax: LLC.054 LLC Client Registered Fail, sap *src_sap* network *network*

Description: Open station Failed for the described sap

LLC.055

Level: C-INFO

Short Syntax: LLC.055 Dynamic Config for Loopback *network_type* requested, nt *network*

Long Syntax: LLC.055 Dynamic Config for Loopback *network_type* requested network *network*

Description: Dynamic Config for Loopback Requested

LLC.056

Level: UI-ERROR

Short Syntax: LLC.056 No mem for loopback net on nt *network*

Long Syntax: LLC.056 No memory for loopback pseudonet on network *network*

Description: Unable to allocate memory for loopback net

LLC.057

Level: C-INFO

Short Syntax: LLC.057 LLC init loop addr *loopbk_addr_p,* dest_net *dest_net_ptr,* macaddr *macaddr,* nt *network*

Long Syntax: LLC.057 LLC init loop addr *loopbk_addr_p,* dest_net *dest_net_ptr,* macaddr *macaddr,* net *network*

Description: Loop net init

LLC.058

Level: C-INFO

Short Syntax: LLC.058 Frame Numbers Vr *Vr,*, Vs *Vs,*, Nr *Nr,* Ns *Ns,*, scb *scb,* nt *dst_net*

Long Syntax: LLC.058 Frame Numbers Vr *Vr,*, Vs *Vs,*, Nr *Nr,*, Ns *Ns,*, scb *scb,* network *dst_net*

Description: Frame Numbers

LLC.059

Level: C-INFO

Short Syntax: LLC.059 LLC FLOW ON, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, dlci *dlci*, nt *dst_net*

Long Syntax: LLC.059 Primitive FLOW ON, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, dlci *dlci*, nt *dst_net*

Description: Flow on is being sent from origin to destination net.

LLC.060

Level: C-INFO

Short Syntax: LLC.060 LLC FLOW ON Data, st *state,*, busy *busy_flag,*, qnum *qnum,*, nt *src_net*

Long Syntax: LLC.060 LLC FLOW ON Data, state *state,*, busy *busy_flag,*, tr_queue_num *qnum,*, network *src_net*

Description: Flow on data to get the exact status of the application sending flow on

---

LLC.061

Level: C-INFO

Short Syntax: LLC.061 LLC FLOW OFF, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, dlci *dlci*, nt *dst_net*

Long Syntax: LLC.061 LLC FLOW OFF, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, dlci *dlci*, nt *dst_net*

Description: Flow off is being sent from origin to destination net.

LLC.062

Level: C-INFO

Short Syntax: LLC.062 LLC FLOW OFF Data, st *state,*, busy *busy_flag,*, qnum *qnum,*, nt *src_net*

Long Syntax: LLC.062 LLC FLOW OFF Data, state *state,*, busy *busy_flag,*, tr_queue_num *qnum,*, network *src_net*

Description: Flow off data to get the exact status of the application sending flow off

---

LLC.063

Level: C-INFO

Short Syntax: LLC.063 Invalid LPDU, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, dlci *dlci*, lpdu_type *lpdu_type* nt *dst_net*

Long Syntax: LLC.063 Invalid LPDU, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, dlci *dlci*, lpdu_type *lpdu_type* nt *dst_net*

Description: Invalid LPDU is being sent from origin to destination net.

LLC.064

Level: C-INFO

Short Syntax: LLC.064 Dest Matching SAP Not Found, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, nt *network*

Long Syntax: LLC.064 Dest Matching SAP Not Found, *src_mac*-> *dst_mac*, saps *src_sap*-> *dst_sap*, network *network*

Description: Sap structure for the destination sap has not been found.

LLC.065

Level: C-INFO

Short Syntax: LLC.065 Validate Sap Failed, sap *src_sap*, sap_pb *sap_pb* nt *network*

Long Syntax: LLC.065 Validate Sap Failed, sap *src_sap*, sap parm block *sap_pb* network *network*

Description: Validate sap Failed for the described sap

LLC.066

Level: C-INFO

Short Syntax: LLC.066 Init Sap Failed, sap *src_sap*, sap_pb *sap_pb* nt *network*

Long Syntax: LLC.066 Init Sap Failed, sap *src_sap*, sap parm block *sap_pb* network *network*

Description: Init sap Failed for the described sap

LLC.067

Level: C-INFO

Short Syntax: LLC.067 Open Station Failed, reason *reason*, sap *src_sap*, stn_pb *stn_pb* nt *network*

Long Syntax: LLC.067 Open Station Failed, reason *reason*, sap *src_sap*, station parm block *stn_pb* network *network*

Description: Open Station Failed for the described sap

LLC.068

Level: U-INFO

Short Syntax: LLC.068 SABME frame rejected, *src_mac*-> *dest_mac*, saps *src_sap*-> *dst_sap*, dlci *dlci*, nt *network*

Long Syntax: LLC.068 SABME frame rejected, *src_mac*-> *dest_mac*, saps *src_sap*-> *dst_sap*, dlci *dlci*, network *network*

Description: SABME frame refused by the local application within the router. This frame came in as a connection indication passive request.

Cause: The frame was not the type the local application wanted to handle.

Action: None.

LLC.069

Level: C-INFO

Short Syntax: LLC.069 SAP Added For Frame Relay, sap *sap*, sap_cb *sap_cb*, dlci *dlci*, nt *network*

Long Syntax: LLC.069 SAP Added For Frame Relay, sap *sap*, sap_cb *sap_cb*, dlci *dlci*, network *network*

Description: Open SAP processing for frame relay has been successful and SAP has been added to the SAP tables.

LLC.070

Level: C-INFO

Short Syntax: LLC.070 SAP Added, sap *sap*, sap_cb *sap_cb*, nt *network*

Long Syntax: LLC.070 SAP Added, sap *sap*, sap_cb *sap_cb*, nt *network*

Description: Open SAP processing has been successful and SAP has been added to the SAP tables.

LLC.071

Level: C-INFO

Short Syntax: LLC.071 SCB Found, scb *scb*, sap *sap*, src_mac *src_mac*, dest_mac *dest_mac*, dlci *dlci*, nt *network*

Long Syntax: LLC.071 SCB Found, scb *scb*, sap *sap*, src_mac *src_mac*, dest_mac *dest_mac*, dlci *dlci*, network *network*

Description: Session Control Block found. This is normal processing. For frame relay, dlci is listed. For non frame relay networks, the dlci number is meaningless.

Chapter 72. LSA Channel Network Interface (LSA)

This chapter describes LSA Channel Network Interface (LSA) messages. For information on message content and how to use the message, refer to the Introduction.

LSA.001

Level: CI-ERROR

Short Syntax: LSA.001 LSA id_check error *error_code*, (nt *network*)

Long Syntax: LSA.001 LSA id_check error *error_code*, (network *network*)

Description: There is no corresponding u_cep_id or u_sap_id to send a response to.

LSA.002

Level: CI-ERROR

Short Syntax: LSA.002 LSA Error, no mem alloc for SAP CB, (nt *network*)

Long Syntax: LSA.002 LSA Error, unable to allocate memory for SAP CB, (network *network*)

Description: Unable to allocate memory for a SAP CB.

LSA.003

Level: CI-ERROR

Short Syntax: LSA.003 LSA Error, no room to alloc mem for LSCB *sap_cb_ptr*, (nt *network*)

Long Syntax: LSA.003 LSA Error, no room available to allocate memory for LSCB *sap_cb_ptr*, (network *network*)

Description: No room in table to allocate an additional LSCB.

LSA.004

Level: CI-ERROR

Short Syntax: LSA.004 LSA dl_close_sap error *llc_rc* from LLC, (nt *network*)

Long Syntax: LSA.004 LSA dl_close_sap error *llc_rc* from LLC, (network *network*)

Description: LLC detected an error when LSA attempted to close a SAP.

LSA.005

Level: CI-ERROR

Short Syntax: LSA.005 LSA dl_close_station error *llc_rc* from LLC, (nt *network*)

Long Syntax: LSA.005 LSA dl_close_station error *llc_rc* from LLC, (network *network*)

Description: LLC detected an error when LSA attempted to close a link station.

LSA.006

Level: CI-ERROR

Short Syntax: LSA.006 LSA dl_data_request error *llc_rc* from LLC, (nt *network*)

Long Syntax: LSA.006 LSA dl_data_request error *llc_rc* from LLC, (network *network*)

Description: LLC detected an error when LSA attempted to send a Type2 data frame.

LSA.007

Level: CI-ERROR

Short Syntax: LSA.007 LSA header_check error *error_code*, (nt *network*)

Long Syntax: LSA.007 LSA header_check error *error_code*, (network *network*)

Description: An error was found while checking the header of an inbound LSA primitive.

LSA.008

Level: CE-ERROR

Short Syntax: LSA.008 LSA stn *p_cep_id* on SAP *sap_value* terminated with rc *vtamrc*, (nt *network*)

Long Syntax: LSA.008 LSA closing link station *p_cep_id* on SAP *sap_value* with rc *vtamrc*, (network *network*)

Description: LSA has closed a link station on a VTAM SAP for this network interface.

LSA.009

Level: C-INFO

Short Syntax: LSA.009 LSA disabling int for VTAM host user *host_user*, (nt *network*)

Long Syntax: LSA.009 LSA disabling interface for VTAM host user *host_user*, (network *network*)

Description: LSA has disabled a VTAM user

connection because the subchannel is offline.

LSA.010

Level: C-INFO

Short Syntax: LSA.010 LSA closing SAP *sap_value*, (nt *network*)

Long Syntax: LSA.010 LSA closing SAP *sap_value*, (network *network*)

Description: LSA has closed a VTAM SAP for this network interface.

LSA.011

Level: C-INFO

Short Syntax: LSA.011 LSA stn *p_cep_id* on SAP *sap_value* closed by VTAM, (nt *network*)

Long Syntax: LSA.011 LSA link station *p_cep_id* on SAP *sap_value* closed by VTAM, (network *network*)

Description: LSA has closed a link station on a VTAM SAP for this network interface.

LSA.012

Level: C-INFO

Short Syntax: LSA.012 LSA stn *p_cep_id* on SAP *sap_value* terminated with rc *vtamrc*, (nt *network*)

Long Syntax: LSA.012 LSA closing link station *p_cep_id* on SAP *sap_value* with rc *vtamrc*, (network *network*)

Description: LSA has closed a link station on a VTAM SAP for this network interface.

LSA.013

Level: CI-ERROR

Short Syntax: LSA.013 LSA Error, invalid p_cep_id or p_sap_id *cookie*, (nt *network*)

Long Syntax: LSA.013 LSA Error, invalid p_cep_id or p_sap_id *cookie*, (network *network*)

Description: An LLC event or data frame was received with an invalid "cookie".

LSA.014

Level: CI-ERROR

Short Syntax: LSA.014 LSA Error, event rcvd from wrong LAN, (nt *network*)

Long Syntax: LSA.014 LSA Error, event received from wrong LAN, (network *network*)

Description: An LLC event or data frame was received from the wrong LAN.

LSA.015

Level: CI-ERROR

Short Syntax: LSA.015 LSA Error, LLC event *event_type* received in state *vlan_status*, (nt *network*)

Long Syntax: LSA.015 LSA Error, LLC event *event_type* received in state *vlan_status*, (network *network*)

Description: LLC event occurred before LSA virtual interface was online.

LSA.016

Level: CI-ERROR

Short Syntax: LSA.016 LSA Error, SABME rcvd for statn *station* in state *status*, (nt *network*)

Long Syntax: LSA.016 LSA Error, SABME received for station *station* in state *status*, (network *network*)

Description: SABME received on connected station.

LSA.017

Level: CI-ERROR

Short Syntax: LSA.017 LSA Error, unxpctd Conn Confirm rcvd for stn *station* in state *status*, (nt *network*)

Long Syntax: LSA.017 LSA Error, unexpected Connect Confirm received for link station *station* in link state *status*, (network *network*)

Description: An unexpected Connect Confirm was received from LLC.

LSA.018

Level: CI-ERROR

Short Syntax: LSA.018 LSA Error, unable to allocate LSCB for SAP *sap_id*, (nt *network*)

Long Syntax: LSA.018 LSA Error, unable to allocate LSCB for SAP *sap_id*, (network *network*)

Description: Unable to find a free LSCB for this SAP.

LSA.019

Level: CI-ERROR

Short Syntax: LSA.019 LSA Error, invalid route info len *route_inf_len* rcvd, (nt *network*)

Long Syntax: LSA.019 LSA Error, invalid routing information length *route_inf_len* received, (network *network*)

Description: Invalid routing information length received.

LSA.020

Level: CI-ERROR

Short Syntax: LSA.020 LSA Error, event type *event_type*, invalid cause code *cause_code* rcvd, (nt *network*)

Long Syntax: LSA.020 LSA Error, event type *event_type*, unexpected cause code *cause_code* received, (network *network*)

Description: Invalid cause code received.

LSA.021

Level: CI-ERROR

Short Syntax: LSA.021 LSA Error, unexpected event type *event_type* rcvd, (nt *network*)

Long Syntax: LSA.021 LSA Error, unexpected event type *event_type* received, (network *network*)

Description: Unknown event type received from LLC.

LSA.022

Level: CE-ERROR

Short Syntax: LSA.022 LSA Error, invalid LAN type *lan_type* or LAN num *lan_num*, (nt *network*)

Long Syntax: LSA.022 LSA Error, invalid LAN type *lan_type* or LAN number *lan_num*, (network *network*)

Description: LAN type or LAN number is invalid.

LSA.023

Level: CI-ERROR

Short Syntax: LSA.023 LSA Error, virt adapt not init, stat is *virt_adap_stat*, (nt *network*)

Long Syntax: LSA.023 LSA Error, virtual adapter not initialized, status is *virt_adap_stat*, (network *network*)

Description: The virtual adapter status is not ENABLED.

LSA.024

Level: CI-ERROR

Short Syntax: LSA.024 LSA Error, frame rcvd with unknwn id *identifier*, (nt *network*)

Long Syntax: LSA.024 LSA Error, frame reveived with unknown identifier *identifier*, (network *network*)

Description: A frame was received with an unknown p_sap_id or p_cep_id.

LSA.025

Level: CI-ERROR

Short Syntax: LSA.025 LSA Error, cntrllr len *controller_len* should be *t2_len*, (nt *network*)

Long Syntax: LSA.025 LSA Error, controller length *controller_len* should be *t2_len*, (network *network*)

Description: The controller length is invalid.

LSA.026

Level: CI-ERROR

Short Syntax: LSA.026 LSA Error, XID poll/final *cmd_resp* or cmd/resp *poll_final* error, (nt *network*)

Long Syntax: LSA.026 LSA Error, XID poll/final *cmd_resp* or command/response field value *poll_final* is incorrect, (network *network*)

Description: The poll/final field contains an invalid value or incorrect state, or the cmd/resp field is invalid.

LSA.027

Level: CI-ERROR

Short Syntax: LSA.027 LSA Error, invalid routing info len *route_info_len*, (nt *network*)

Long Syntax: LSA.027 LSA Error, invalid routing information length of *route_info_len*, (network *network*)

Description: The routing information length is invalid.

LSA.028

Level: CI-ERROR

Short Syntax: LSA.028 LSA Error, frame len *frame_size* exceeded max *frame_max*, (nt *network*)

Long Syntax: LSA.028 LSA Error, frame length of *frame_size* exceeded maximum of *frame_max*, (network *network*)

Description: The frame size exceeded the maximum.

LSA.029

Level: CI-ERROR

Short Syntax: LSA.029 LSA Error, invalid SSAP *ssap* for Test/XID, (nt *network*)

Long Syntax: LSA.029 LSA Error, invalid SSAP *ssap* for Test/XID, (network *network*)

Description: The Source SAP is invalid for a Test/XID.

LSA.030

Level: CI-ERROR

Short Syntax: LSA.030 LSA Error, invalid SAP CB ptr *sap_cb*, (nt *network*)

Long Syntax: LSA.030 LSA Error, invalid SAP CB pointer *sap_cb*, (network *network*)

Description: The SAP CB pointer is invalid.

LSA.031

Level: CI-ERROR

Short Syntax: LSA.031 LSA Error, unexpected ret code 0x *ret_code* from LLC call to *func_name*, (nt *network*)

Long Syntax: LSA.031 LSA Error, unexpected return code 0x *ret_code* from LLC call to *func_name*, (network *network*)

Description: LLC has returned an error code to the LSA net handler.

LSA.032

Level: CI-ERROR

Short Syntax: LSA.032 LSA Error, prim type *primitive* vtam code *vtam_code*, parm *parm* (nt *network*)

Long Syntax: LSA.032 LSA Error, primitive type *primitive* vtam code *vtam_code*, parameter *parm* (network *network*)

Description: An error occurred processing the VTAM request/response. The specified parameter was responsible.

LSA.033

Level: C-INFO

Short Syntax: LSA.033 LSA enabling int for VTAM host user *host_user*, (nt *network*)

Long Syntax: LSA.033 LSA enabling interface for VTAM host user *host_user*, (network *network*)

Description: LSA has enabled a VTAM user connection.

LSA.034

Level: C-INFO

Short Syntax: LSA.034 LSA opening SAP *sap_value*, p_sap_id *p_sap_id* (nt *network*)

Long Syntax: LSA.034 LSA opening SAP *sap_value*, p_sap_id *p_sap_id* (network *network*)

Description: LSA has opened a VTAM SAP for this network interface.

LSA.035

Level: C-INFO

Short Syntax: LSA.035 LSA stn *p_cep_id* on SAP *sap_value* opened by VTAM, (nt *network*)

Long Syntax: LSA.035 LSA link station *p_cep_id* on SAP *sap_value* opened by VTAM, (network *network*)

Description: LSA has opened a link station on a VTAM SAP for this network interface.

LSA.036

Level: CI-ERROR

Short Syntax: LSA.036 LSA Event, prim type *primitive* vtam code *vtam_code*, (nt *network*)

Long Syntax: LSA.036 LSA Event, primitive type *primitive* vtam code *vtam_code*, (network *network*)

Description: An non-error event occurred processing the VTAM request/response.

LSA.037

Level: CI-ERROR

Short Syntax: LSA.037 LSA Error, MAC adap not enabled, (nt *network*)

Long Syntax: LSA.037 LSA Error, MAC adapter is not enabled, (network *network*)

Description: The MAC adapter is not enabled.

LSA.038

Level: CI-ERROR

Short Syntax: LSA.038 LSA Error, out of host user blocks, (nt *network*)

Long Syntax: LSA.038 LSA Error, out of host user blocks, (network *network*)

Description: There are no more host user blocks.

LSA.039

Level: CI-ERROR

Short Syntax: LSA.039 LSA Error, unknwn or unexpect req/resp *primitive* rcvd, (nt *network*)

Long Syntax: LSA.039 LSA Error, unknown or unexpected request/response *primitive* received, (network *network*)

Description: An unknown or unexpected primitive was received.

LSA.040

Level: CI-ERROR

Short Syntax: LSA.040 LSA Error, invalid req/resp *primitive* for statn *station_status* stat *network/)*

Long Syntax: LSA.040 LSA Error, invalid request/response *primitive* for station *station_status* in status *network/)*

Description: The req/response is invalid for this stations status.

LSA.041

Level: CI-ERROR

Short Syntax: LSA.041 LSA dl_open_station error 0x *llc_rc* from LLC, (nt *network*)

Long Syntax: LSA.041 LSA dl_open_station error 0x *llc_rc* from LLC, (network *network*)

Description: LLC detected an error when LSA attempted to open a link station.

LSA.042

Level: C-INFO

Short Syntax: LSA.042 LSA stn *p_cep_id* on SAP *sap_value* conn est, (nt *network*)

Long Syntax: LSA.042 LSA link station *p_cep_id* on SAP *sap_value* connection established, (network *network*)

Description: LSA has established an LLC connection with a remote link station.

LSA.043

Level: C-INFO

Short Syntax: LSA.043 LSA stn *p_cep_id* on SAP *sap_value* conn rej by *rej_end*, (nt *network*)

Long Syntax: LSA.043 LSA link station *p_cep_id* on SAP *sap_value* connection rejected by *rej_end*, (network *network*)

Description: An LLC connection to a remote link station was rejected.

LSA.044

Level: C-INFO

Short Syntax: LSA.044 LSA net *netnum* rcvd netup from net *lan_netnum* (nt *network*)

Long Syntax: LSA.044 LSA net *netnum* received netup from net *lan_netnum* (network *network*)

Description: LSA net handler received netup from attached downstream LAN.

LSA.045

Level: C-INFO

Short Syntax: LSA.045 LSA net *netnum* MAC addr *macaddr* set (nt *network*)

Long Syntax: LSA.045 LSA net *netnum* MAC address set to *macaddr* (network *network*)

Description: LSA net handler received netup from attached downstream LAN.

LSA.046

Level: P-TRACE

Short Syntax: LSA.046 LSA user data to base net handler (nt *network*)

Long Syntax: LSA.046 LSA user data sent to the base net handler (network *network*)

Description: The LSA net handler sent user data to the base net handler.

LSA.047

Level: P-TRACE

Short Syntax: LSA.047 LSA user data from base net handler (nt *network*)

Long Syntax: LSA.047 LSA user data received from the base net handler (network *network*)

Description: The LSA net handler received user data from the base net handler.

LSA.048

Level: P-TRACE

Short Syntax: LSA.048 LSA prim *prim_code* to base net handler (nt *network*)

Long Syntax: LSA.048 LSA primitive *prim_code* sent to the base net handler (network *network*)

Description: The LSA net handler sent data to the base net handler. This data is primitives that contain primitives use by LSA to run the connection to the host; not user data running over the connection.

LSA.049

Level: P-TRACE

Short Syntax: LSA.049 LSA prim *prim_code* from base net handler (nt *network*)

Long Syntax: LSA.049 LSA primitive *prim_code* received from the base net handler (network *network*)

Description: The LSA net handler received data from the base net handler. This data is primitives that contain primitives use by LSA to run the connection to the host; not user data running over the connection.

LSA.050

Level: C-INFO

Short Syntax: LSA.050 *in_out* flow *on_off* LSA stn *p_cep_id* on SAP *sap_value*, (nt *network*)

Long Syntax: LSA.050 *in_out* flow turned *on_off* for LSA link station *p_cep_id* on SAP *sap_value*, (network *network*)

Description: VTAM or LLC has turned flow control on/off for an LSA link station.

LSA.051

Level: CE-ERROR

Short Syntax: LSA.051 FRMR *in_out* for LSA stn *p_cep_id* on SAP *sap_value*, (nt *network*)

Long Syntax: LSA.051 Frame reject *in_out* for LSA link station *p_cep_id* on SAP *sap_value*, (network *network*)

Description: A frame reject was sent or received for an LSA link station.

LSA.053

Level: C-INFO

Short Syntax: LSA.053 LSA rcvd *event* event for SAP/stn *sap_cep* (nt *network*)

Long Syntax: LSA.053 LSA received *event* event from LLC for SAP/link station *sap_cep* (network *network*)

Description: The LSA net handler received an event notification from the LLC application.

LSA.054

Level: CI-ERROR

Short Syntax: LSA.054 LSA frame not sent to host rc *rc* (nt *network*)

Long Syntax: LSA.054 LSA frame not sent to host - return code *rc* (network *network*)

Description: The LSA net handler could not send a frame to the host.

LSA.055

Level: C-INFO

Short Syntax: LSA.055 APPN Loopback net installed (nt *network*)

Long Syntax: LSA.055 APPN Loopback net installed (nt *network*)

Description: The APPN net handler for LLC loopback has been installed.

LSA.056

Level: C-INFO

Short Syntax: LSA.056 APPN Loopback net init complete (nt *network*)

Long Syntax: LSA.056 APPN Loopback net initialization complete (nt *network*)

Description: The APPN net handler for LLC loopback has been initialized.

LSA.057

Level: C-INFO

Short Syntax: LSA.057 LSA net *netnum* disabled by user (nt *network*)

Long Syntax: LSA.057 LSA net *netnum* disabled by user (nt *network*)

Description: The LSA net handler disable routine has been invoked.

LSA.058

Level: CI-ERROR

Short Syntax: LSA.058 LSA can't get IORB for *cause* (nt *network*)

Long Syntax: LSA.058 LSA unable to get IORB for *cause* (nt *network*)

Description: The LSA net handler could not get an IORB. An event may have occurred which will not be reported to VTAM.

LSA.059

Level: C-INFO

Short Syntax: LSA.059 LSA net passed self-test (nt *network*)

Long Syntax: LSA.059 LSA net passed self-test (nt *network*)

Description: The LSA net handler passed its self-test routine. Both the channel adapter and downstream LAN adapter have gone netup.

LSA.060

Level: C-INFO

Short Syntax: LSA.060 LSA net *netnum* netdwn by LAN net *lan_netnum* netdwn (nt *network*)

Long Syntax: LSA.060 LSA net *netnum* went netdown because LAN net *lan_netnum* went netdown (nt *network*)

Description: The LSA net handler went netdown because the downstream LAN adapter went netdown.

LSA.061

Level: CI-ERROR

Short Syntax: LSA.061 LSA LLC can't find MAC *macaddr* SAP *sap* for prim *prim* (nt *network*)

Long Syntax: LSA.061 LSA LLC can't find MAC *macaddr* SAP *sap* for VTAM primitive *prim* (nt *network*)

Description: LLC could not find the specified SAP open for the MAC address. This error only occurs for LLC loopback.

Panic lsanomem

Short Syntax: lsanomem: LSA handler no memory

Description: An LSA handler cannot allocate memory for control block(s).

Action: Contact customer service.

Panic lsansram

Short Syntax: lsansram: LSA SRAM not found

Description: The SRAM record for an LSA net handler could not be found.

Action: Contact customer service.

Panic lsanolan

Short Syntax: lsanolan: LSA target LAN not defined

Description: LSA target LAN not defined.

Action: Contact customer service.

Panic lsanolsa

Short Syntax: lsanolsa: APPN loopback activated with no LSA net defined

Description: APPN loopback activated with no LSA net defined.

Action: Contact customer service.

Panic lsabcall

Short Syntax: lsabcall: bad call to routine

Description: An invalid call was made to a routine.

Action: Contact customer service.

Chapter 73. MAC Filtering (MCF)

This chapter describes MAC Filtering (MCF) messages. For information on message content and how to use the message, refer to the Introduction.

MCF.001

Level: P-TRACE

Short Syntax: MCF.001 MCF enbl

Long Syntax: MCF.001 MAC Filtering enabled

Description: The MAC Filtering database has been enabled.

MCF.002

Level: P-TRACE

Short Syntax: MCF.002 MCF dsbl

Long Syntax: MCF.002 MAC Filtering disabled

Description: The MAC Filtering database has been disabled.

MCF.003

Level: UI-ERROR

Short Syntax: MCF.003 MCF init-err no mem

Long Syntax: MCF.003 MAC Filtering no memory for initialization

Description: The MAC Filtering database initialization has failed to allocate memory for the MAC Filter Control structures.

Cause: Insufficient memory to support this configuration.

Action: Change configuration to reduce memory consumption. May require additional physical memory.

MCF.004

Level: UI-ERROR

Short Syntax: MCF.004 MCF init-err bd ifc nmbr - *filter_intf*

Long Syntax: MCF.004 MAC Filtering bad interface number *filter_intf* given in initialization

Description: The MAC Filtering database initialization has a non-existent interface configured with a filter.

Cause: The user has configured a trap for an interface which does not exist.

Action: Delete the erroneous trap or add the interface to which it is assigned.

MCF.005

Level: UI-ERROR

Short Syntax: MCF.005 MCF init-err gen flt db

Long Syntax: MCF.005 MAC Filtering database initialization error

Description: The MAC Filtering database initialization has encountered an error in creating the filter database.

Cause: Insufficient memory to support this configuration.

Action: Change configuration to reduce memory consumption. May require additional physical memory.

MCF.006

Level: U-TRACE

Short Syntax: MCF.006 MCF add filt at *name* ok

Long Syntax: MCF.006 MAC Filtering initialized filter at *name* successfully

Description: The MAC Filter configured on at the given direction and interface has been successfully initialized and is in effect.

MCF.007

Level: U-TRACE

Short Syntax: MCF.007 flt *filter* exc frm *source*-> *dest*, nt *network* int *intname*/ *intnum*

Long Syntax: MCF.007 MAC Filter *filter* excludes frame *source*-> *dest*, network *network* interface *intname*/ *intnum*

Description: The specified MAC Filter has matched a frame on the given direction and interface. The frame was excluded from further processing.

MCF.008

Level: U-TRACE

Short Syntax: MCF.008 flt *filter* inc frm *source*-> *dest*, nt *network* int *intname*/ *intnum*

Long Syntax: MCF.008 MAC Filter *filter* includes frame *source*-> *dest*, network *network* interface *intname*/ *intnum*

Description: The specified MAC Filter has matched a frame on the given direction and interface. The frame

was included in further processing.

MCF.009

Level: U-TRACE

Short Syntax: MCF.009 flt *filter* tag(*tag*) frm *source*-> *dest*, nt *network* int *intname*/ *intnum*

Long Syntax: MCF.009 MAC Filter *filter* tags(*tag*) frame *source*-> *dest*, network *network* interface *intname*/ *intnum*

Description: The specified MAC Filter has matched a frame on the given direction and interface. The frame was filtered according to the configured action.

Panic mcfimem

Short Syntax: MCF init fail, no mem

Description: The MAC Filtering initialization failed to allocate sufficient memory to complete initialization.

Action: Contact customer service.

Chapter 74. MMC

This chapter describes MMC messages. For information on message content and how to use the message, refer to the Introduction.

MMC.001

Level: C-INFO

Short Syntax: MMC.001 Dscrd ATM pkt on int *net_num*, int is dwn.

Long Syntax: MMC.001 Discard ATM packet on interface *net_num*, because interface is down.

Description: ATM packet discarded

MMC.002

Level: UI-ERROR

Short Syntax: MMC.002 switch_receive_frame. Unspprted msg (type= *type*) rcvd

Long Syntax: MMC.002 switch_receive_frame. Unsupported message type *type* received

Description: Unsupported message received at switch_receive_frame()

MMC.003

Level: UI-ERROR

Short Syntax: MMC.003 *func_name*: int *net_num* is not regstrd

Long Syntax: MMC.003 *func_name*: interface *net_num* is not registered

Description: interface is not registered with device driver

MMC.004

Level: UI-ERROR

Short Syntax: MMC.004 *func_name*: int *net_num* is not open

Long Syntax: MMC.004 *func_name*: int *net_num* is not open

Description: interface has not been opened in the dev

MMC.005

Level: C-INFO

Short Syntax: MMC.005 Frame ACK timed out for int *net_num* msg type *msg_type*

Long Syntax: MMC.005 Frame ACK timed out for interface *net_num*, message type *msg_type*

Description: Frame acknolwedgement timer out in device driver.

MMC.006

Level: DEBUG

Short Syntax: MMC.006 Initializing cpu_enet_init()

Long Syntax: MMC.006 Initializing cpu_enet_init()

Description: The cpu_enet_init() routine entered

MMC.007

Level: DEBUG

Short Syntax: MMC.007 cpu_enet_ *type*_dscr_list: 0x *addr*

Long Syntax: MMC.007 cpu_enet_ *type*_dscr_list: 0x *addr*

Description: Print the CPU Enet TX/RX descriptor list

MMC.008

Level: C-INFO

Short Syntax: MMC.008 cpu_enet_rx_dequeue: Discrdng rx frame *value*

Long Syntax: MMC.008 cpu_enet_rx_dequeue: Discarding rx frame *value*

Description: The cpu_enet_rx_dequeue routine is discarding an ATM packet

MMC.009

Level: DEBUG

Short Syntax: MMC.009 Error dequeuing *type* bufs from enet

Long Syntax: MMC.009 Error dequeuing *type* buffers from enet

Description: Error dequeuing buffers in device driver

MMC.010

Level: DEBUG

Short Syntax: MMC.010 enet_receive_frame: valid *net_num* cell cnt = int , debug , for

Long Syntax: MMC.010 enet_receive_frame: valid *net_num*, cell cnt = , i/f , debug , for ″

Description: An iorb allocation failed

MMC.011

Level: CI-ERROR

Short Syntax: MMC.011 tx frame over enet failed

Long Syntax: MMC.011 transmit frame over enet failed

Description: Attempt to transmit Ethernet frame failed

MMC.012

Level: DEBUG

Short Syntax: MMC.012 ATM VCC INDEX 0x *vcc_index* LECID: *lec_index*

Long Syntax: MMC.012 ATM VCC INDEX 0x *vcc_index* LECID: *lec_index*

Description: Decodes an ATM -> CPU message

MMC.013

Level: CI-ERROR

Short Syntax: MMC.013 addAtmConnection failed on *net_num*

Long Syntax: MMC.013 addAtmConnection failed on *net_num*

Description: addAtmConnection failed on this interface

MMC.014

Level: C-INFO

Short Syntax: MMC.014 deleteAtmConnection failed - *rc connection_ea , rc2, connection_ae*

Long Syntax: MMC.014 deleteAtmConnection failed - *rc connection_ea , rc2, connection_ae*

Description: deleteAtmConnection failed

MMC.015

Level: CI-ERROR

Short Syntax: MMC.015 *func_name*: illegal dist type

Long Syntax: MMC.015 *func_name*: illegal distribution type

Description: Illegal distribution type

MMC.016

Level: DEBUG

Short Syntax: MMC.016 *string*

Long Syntax: MMC.016 *string*

Description: This is a generic string passed up from MMC

MMC.017

Level: CI-ERROR

Short Syntax: MMC.017 nt *net_num1* reliable msg buffer alloc failed

Long Syntax: MMC.017 nt *net_num1* reliable msg buffer allocation failed

Description: Reliable message buffer allocation failed\r\n

MMC.018

Level: CI-ERROR

Short Syntax: MMC.018 nt *net_num1 func_name*: unable to allocate buffer - dropping frame

Long Syntax: MMC.018 nt *net_num1 func_name*: unable to allocate buffer - dropping frame

Description: unable to allocate buffer

MMC.019

Level: C-INFO

Short Syntax: MMC.019 nt *net_num1* Frame ACK timed out - bcast msg type *msg_type*

Long Syntax: MMC.019 nt *net_num1* Frame ACK timed out - broadcast message type *msg_type*

Description: This message indicates that the frame ack failed for a reliable broadcast

---

MMC.020

Level: CI-ERROR

Short Syntax: MMC.020 nt *net_num1 func_name*: msg buf alloc failed

Long Syntax: MMC.020 nt *net_num1 func_name*: message buffer allocation failed

Description: Frame structure allocation failed\r\n

MMC.021

Level: C-INFO

Short Syntax: MMC.021 Resetting EPIF *epif_num*

Long Syntax: MMC.021 Resetting EPIF *epif_num*

Description: Resetting EPIF

MMC.022

Level: CI-ERROR

Short Syntax: MMC.022 Reliable Msg Queue full - dropping msg type: *msg_type*

Long Syntax: MMC.022 Reliable Msg Queue full - dropping msg type: *msg_type*

Description: Reliable Msg Queue full - dropping msg

MMC.023

Level: CI-ERROR

Short Syntax: MMC.023 Reliable Msg Entering Data Discard mode: *num_messages*

Long Syntax: MMC.023 Reliable Msg Entering Data Discard mode: *num_messages*

Description: The reliable message queue is entering data discard mode.

MMC.024

Level: CI-ERROR

Short Syntax: MMC.024 Reliable Msg Exiting Data Discard mode: *num_messages*

Long Syntax: MMC.024 Reliable Msg Exiting Data Discard mode: *num_messages*

Description: The reliable has become uncongested and is returning to normal mode.

Chapter 75. MPC Channel Network Interface (MPC)

This chapter describes MPC Channel Network Interface (MPC) messages. For information on message content and how to use the message, refer to the Introduction.

MPC.001

Level: UE-ERROR

Short Syntax: MPC.001 *file*(*line*): No IORB allocated (nt *network*)

Long Syntax: MPC.001 *file*(*line*): No IORB could be allocated (network *network*)

Description: MPC+ processing required an IORB that could not be obtained.

MPC.002

Level: P-TRACE

Short Syntax: MPC.002 *file*(*line*): netfout did not send data frame (nt *network*)

Long Syntax: MPC.002 *file*(*line*): netfout did not send out data frame (network *network*)

Description: The MPC+ Net Handler did a netfout that failed to send the frame out.

MPC.003

Level: UE-ERROR

Short Syntax: MPC.003 *file*(*line*): PDU invalid (nt *network*)

Long Syntax: MPC.003 *file*(*line*): PDU received was invalid (network *network*)

Description: The MPC+ PDU was invalid.

Cause: The MPC+ Net Handler did not like the PDU that was received over the channel.

Action: Contact Software Support.

MPC.004

Level: C-TRACE

Short Syntax: MPC.004 *file*(*line*): input *fsminput* curr stte *curr_state* new stte *new_state* actn *action* (nt *network*)

Long Syntax: MPC.004 *file*(*line*): input *fsminput* current state *curr_state* new state *new_state* action *action* (network *network*)

Description: Show the inputs to the MPC+ FSM that is given in the message.

MPC.005

Level: UE-ERROR

Short Syntax: MPC.005 *file*(*line*): SDU (*rutype_string*) invalid (nt *network*)

Long Syntax: MPC.005 *file*(*line*): SDU (*rutype_string*) was invalid (network *network*)

Description: The MPC+ SDU was invalid.

Cause: The MPC+ Net Handler did not like the SDU in the PDU that was received over the channel.

Action: Contact Software Support.

MPC.006

Level: UE-ERROR

Short Syntax: MPC.006 *file*(*line*): no *cbtype_string* CB available (nt *network*)

Long Syntax: MPC.006 *file*(*line*): no *cbtype_string* control block available (network *network*)

Description: Storage for a control block or its resources (i.e. IORBs) was not able to be obtained for MPC+ Net Handler.

MPC.007

Level: UE-ERROR

Short Syntax: MPC.007 *file*(*line*): *conntype_string_* equal user data (nt *network*)

Long Syntax: MPC.007 *file*(*line*): *conntype_string_* Virtual Circuit user data was equal (network *network*)

Description: The user_data on the virtual circuit was the same.

Cause: Both VTAM and the MPC+ Net Handler picked the same user data.

Action: If the connection type is CM, then try to bring up the MPC+ Group again. Hopefully , the random number in the user data will be different the next time.

Action: If the User connection, then try to modify the user data. Note-APPN connections use the control point names.

MPC.008

Level: UE-ERROR

Short Syntax: MPC.008 *file*(*line*): *cbtype_string* CB was not found for *cmdtype_string* (nt *network*)

Long Syntax: MPC.008 *file*(*line*): *cbtype_string* control block could not be located for *cmdtype_string* (network *network*)

Description: The MPC+ control block for the command (primitive/SDU) could not be located.

Cause: The control block was already freed because the resources have come down.

Action: Typically, No action is required.

MPC.009

Level: U-INFO

Short Syntax: MPC.009 *file*(*line*): *fsmtype_string* FSM invalid, input = *input* state = *state* (nt *network*)

Long Syntax: MPC.009 *file*(*line*): *fsmtype_string* FSM had invalid input, input = *input* state = *state* (network *network*)

Description: One the the MPC+ FSMs received an input that should not occur in the current state.

Action: Typically, No action is required. If the problem persists, contact Software Support.

MPC.010

Level: UE-ERROR

Short Syntax: MPC.010 *file*(*line*): Primitive (*primtype_string*) invalid (nt *network*)

Long Syntax: MPC.010 *file*(*line*): Primitive (*primtype_string*) was invalid (network *network*)

Description: The MPC+ primitive was invalid.

Cause: The MPC+ Net Handler did not like the primitive it received from other processing in the box.

Action: Contact Software Support.

MPC.011

Level: C-INFO

Short Syntax: MPC.011 *file*(*line*): Primitive (*primtype_string*) was a dup (nt *network*)

Long Syntax: MPC.011 *file*(*line*): Primitive (*primtype_string*) was a duplicate (network *network*)

Description: The MPC+ primitive was for a resource that was already active or in the process of becoming active.

Action: Typically, No action is required. If the problem persists, contact Software Support.

MPC.012

Level: P-TRACE

Short Syntax: MPC.012 *file*(*line*): *conntype_string* conn congested (nt *network*)

Long Syntax: MPC.012 *file*(*line*): *conntype_string* connection is congested (network *network*)

Description: The connection that the MPC+ PDU was received over was congested so the PDU was discarded.

Action: Typically, No action is required. If the problem persists, contact Software Support.

MPC.013

Level: UI-ERROR

Short Syntax: MPC.013 *file*(*line*): cmd (*commtype_string*) was unsupp (nt *network*)

Long Syntax: MPC.013 *file*(*line*): command (*commtype_string*) was unsupported (network *network*)

Description: The command from the Device Driver was unsupported.

Action: Contact Software Support.

MPC.015

Level: C-INFO

Short Syntax: MPC.015 *file*(*line*): subchnnl (*local_sc_num*) not expectng *cmd_string* cmd (nt *network*)

Long Syntax: MPC.015 *file*(*line*): local subchannel (*local_sc_num*) not expecting *cmd_string* command in current state (network *network*)

Description: An MPC+ subchannel received a command that was not expected in its current state. The command was ignored

Cause: VTAM resent the command that was already processed for the subchannel.

Action: Typically, No action is required. If the problem persists, contact Software Support.

MPC.016

Level: UI-ERROR

Short Syntax: MPC.016 *file*(*line*): timer (*timer_string*) popped when not running (nt *network*)

Long Syntax: MPC.016 *file*(*line*): timer (*timer_string*) popped when it was not currently running (network *network*)

Description: An MPC+ timer was running when the processing did not think it was running.

Action: Contact Software Support.

MPC.017

Level: UE-ERROR

Short Syntax: MPC.017 *file*(*line*): XID2(*xid2_type*) failed validation (nt *network*)

Long Syntax: MPC.017 *file*(*line*): XID2(*xid2_type*) failed was validation (network *network*)

Description: The MPC+ XID2 received failed its validation checks and will be consider bad.

Cause: VTAM may be using a downlevel version of MPC instead of HPDT MPC. Only HPDT MPC (i.e. MPC+) is supported.

Action: Configure VTAM to use HPDT MPC.

Cause: The random numbers in the XID2 exchange were the same in the MPC+ Net Handler and VTAM.

Action: Try to bring up the MPC Group again. Hopefully, different random number will be exchanged the next time.

MPC.018

Level: C-INFO

Short Syntax: MPC.018 *file*(*line*): dup. PDU was received (nt *network*)

Long Syntax: MPC.018 *file*(*line*): A duplicate PDU was received (network *network*)

Description: The MPC+ Sequence Manager discarded a duplicate PDU that was recevied.

Action: No action is required.

MPC.019

Level: UI-ERROR

Short Syntax: MPC.019 *file*(*line*): *conn_string* connection cleaned up by Seq. Manager (nt *network*)

Long Syntax: MPC.019 *file*(*line*): *conn_string* connection was cleaned up by Sequence Manager (network *network*)

Description: The MPC+ Sequence Manager cleaned up the connection because of sequencing or acknowledgement problems.

Cause: Data got out of sequence and was not able to recover.

Action: The connections should come back and recover. If the problem continues to happen, then check that the sequence timer value for the connection is not too low. Increase the sequence timer value for the connection if it may be too low. The problem could have been due to delays in traffic that the sequence timer value was not high enough.

Cause: Data was not being acknowledged from VTAM in a timely matter.

Action: If data was still flowing, then may need to modify the sequence timer value for the connection.

MPC.020

Level: P-TRACE

Short Syntax: MPC.020 *file*(*line*): MPC+ *command_string* to base channel (nt *network*)

Long Syntax: MPC.020 *file*(*line*): MPC+ *command_string* sent to base channel (network *network*)

Description: The MPC+ Net Handler sent an MPC command or data to the base channel Net Handler.

MPC.021

Level: P-TRACE

Short Syntax: MPC.021 *file*(*line*): MPC+ *command_string* from base channel (nt *network*)

Long Syntax: MPC.021 *file*(*line*): MPC+ *command_string* received from base channel (network *network*)

Description: The MPC+ Net Handler received an MPC command or data from the base channel Net Handler.

MPC.022

Level: UE-ERROR

Short Syntax: MPC.022 *file*(*line*): *ru_string* invalid. *err_string*: 0x *err_data* (nt *network*)

Long Syntax: MPC.022 *file*(*line*): *ru_string* validation failed. *err_string*: 0x *err_data* (network *network*)

Description: Configuration type parameters failed validation.

Cause: Invalid data configured at this end or received from the other end

Action: Fix configuration.

MPC.023

Level: ALWAYS

Short Syntax: MPC.023 *file*(*line*): Disabled Net(*rea_string*). *err_string*: 0x *err_data* (nt *network*)

Long Syntax: MPC.023 *file*(*line*): Error: Disabled Network Interface (*rea_string*). *err_string*: 0x *err_data* (network *network*)

Description: Net Handler Interface disabled due to serious error

Cause: Storage allocation failure

Action: There is not currently enough storage for the configured resources. Storage may become available. To

attempt to bring back up the interface, issue test from the operator console.

Cause: Attempt by invalid protocol to use interface

Action: Probably a software error, check the configuration. If ok, contact customer service.

MPC.024

Level: C-INFO

Short Syntax: MPC.024 *file*(*line*): *event_string* IP Addr *IP_address* MPC+ nt *network*

Long Syntax: MPC.024 *file*(*line*): *event_string* IP Address *IP_address* on MPC+ network *network*

Description: An IP address has been added or deleted from the MPC+ Net Handler

---

MPC.025

Level: P-TRACE

Short Syntax: MPC.025 *file*(*line*): MPC+ user data to base channel (nt *network*)

Long Syntax: MPC.025 *file*(*line*): MPC+ user data sent to base channel (network *network*)

Description: The MPC+ Net Handler sent user data to the base channel Net Handler. A PDU may contain multiple user data packets. This message counts once per PDU, but displays one per packet in the PDU.

MPC.026

Level: P-TRACE

Short Syntax: MPC.026 *file*(*line*): MPC+ user data from base channel (nt *network*)

Long Syntax: MPC.026 *file*(*line*): MPC+ user data received from base channel (network *network*)

Description: The MPC+ Net Handler received user data from the base channel Net Handler. A PDU may contain multiple user data packets. This message counts once per PDU, but displays one per packet in the PDU.

MPC.027

Level: UE-ERROR

Short Syntax: MPC.027 *file*(*line*): Wrong protocol(*protocol1_string*) tried to use *protocol2_string* Exclusive Use MPC+ Group (nt *network*)

Long Syntax: MPC.027 *file*(*line*): Wrong protocol(*protocol1_string*) tried to use *protocol2_string* Exclusive Use MPC+ Group (network *network*)

Description: The MPC+ Group can not be used by the requesting protocol based on the configuration.

Action: Double check that configuration (Exclusive Use Enable) was correct.

MPC.028

Level: UE-ERROR

Short Syntax: MPC.028 *file*(*line*): *protocol1_string* Exclusive Use MPC+ Group already in use by *protocol2_string* (nt *network*)

Long Syntax: MPC.028 *file*(*line*): *protocol1_string* Exclusive Use MPC+ Group already in use by *protocol2_string* (network *network*)

Description: The MPC+ Group can not be used by the requesting protocol based on the configuration and the fact that another instance of the protocol is using it.

Action: Double check that configuration (Exclusive Use Enable) was correct.

MPC.029

Level: UE-ERROR

Short Syntax: MPC.029 *file*(*line*): Subchannel (0x *subnum*) READ or WRITE on both sides (nt *network*)

Long Syntax: MPC.029 *file*(*line*): Subchannel (0x *subnum*) is coded READ or coded WRITE on both sides (network *network*)

Description: The Subchannel list is either coded as READ on both sides of the channel or coded a WRITE on both sides of the channel.

Action: Double check that configuration (READ vs WRITE) was correct.

MPC.030

Level: UE-ERROR

Short Syntax: MPC.030 *file*(*line*): *cmdtype_string* was received for SC 0x *subnum* which is not part of this Net (nt *network*)

Long Syntax: MPC.030 *file*(*line*): *cmdtype_string* was received for subchannel 0x *subnum* which is not part of this MPC+ NET (network *network*)

Description: The Command listed was received for Subchannel that is not part of the Net handler

Action: Contact Software Support.

MPC.031

Level: C-INFO

Short Syntax: MPC.031 *file*(*line*): MPC+ nt *network* protocol down for protocol *prtcl*

Long Syntax: MPC.031 *file*(*line*): MPC+ network *network* protocol down for protocol *prtcl*

Description: The MPC+ Net Handler has receive a protocol down.

MPC.032

Level: C-INFO

Short Syntax: MPC.032 *file*(*line*): cmd (*commtype_string*) was discarded - net is disabled (nt *network*)

Long Syntax: MPC.032 *file*(*line*): command (*commtype_string*) was discarded - net is disabled (network *network*)

Description: The command was discarded because the net was disabled.

Action: Contact Software Support.

MPC.033

Level: UI-ERROR

Short Syntax: MPC.033 *file*(*line*): MPC+ discarded *protocolid_string* IORB due to no data (nt *network*)

Long Syntax: MPC.033 *file*(*line*): MPC+ discarded *protocolid_string* IORB due to no data (network *network*)

Description: The IORB was discarded because it did not contain any data.

Action: Contact Software Support.

MPC.034

Level: CE-ERROR

Short Syntax: MPC.034 *file*(*line*): *protocol_string* tried to use a non-Exclusive Use MPC+ Group (nt *network*)

Long Syntax: MPC.034 *file*(*line*): *protocol_string* tried to use a non-Exclusive Use MPC+ Group (network *network*)

Description: The MPC+ Group can not be used by the requesting protocol based on the configuration. This may not be an error. Some Host users (e.g. TCP/IP) may deliberately send down requests for multiple MPC protocols and do not expect them all to succeed.

Action: Double check that configuration (Exclusive Use Enable) was correct.

MPC.035

Level: C-INFO

Short Syntax: MPC.035 *file*(*line*): BF Decide: Loc Conn Token=0x *conn_string* BF= *bf* ThruPut= *thruput*(ms/pack) ElapTime= *elaptim*(ms) (nt *network*)

Long Syntax: MPC.035 *file*(*line*): Blocking Factor Decision: Local Connection Token=0x *conn_string* Blocking Factor= *bf* Thruput= *thruput*(millisec/packet) ElapsedTime= *elaptim*(millsec) (network *network*)

Description: Provides information on how the MPC+ blocking algorithm is behaving for the connection.

Action: None.

MPC.036

Level: C-INFO

Short Syntax: MPC.036 *file*(*line*): Block Push: Loc Conn Token=0x *conn_string* BF= *bf* PushCnt= *pushcnt* (nt *network*)

Long Syntax: MPC.036 *file*(*line*): Block Push: Local Connection Token=0x *conn_string* Blocking Factor= *bf* PushCnt= *pushcnt* (network *network*)

Description: Provides information on how the MPC+ blocking algorithm is behaving for the connection.

Action: None.

Panic mpcnomem

Short Syntax: mpcnomem: MPC+ Net Handler no memory

Description: An MPC+ Net Handler cannot allocate memory for control block(s).

Action: Contact customer service.

Panic mpcnsram

Short Syntax: mpcnsram: MPC+ channel SRAM not found

Description: The SRAM record for an MPC+ channel Net handler could not be found.

Action: Contact customer service.

Panic mpcnosub

Short Syntax: mpcnosub: subch not found

Description: The requested logical path and device address was not found in the channel handler subchannel table.

Action: Contact customer service.

Chapter 76. Multicast Address Resolution Protocol (MARS)

This chapter describes Multicast Address Resolution Protocol (MARS) messages. For information on message content and how to use the message, refer to the Introduction.

MARS.001

Level: U-INFO

Short Syntax: MARS.001 MARS Q ovf (destId= *destQueue*) for nt *network*

Long Syntax: MARS.001 MARS Queue overflow (destId= *destQueue*) net *network*

Description: A MARS packet was discarded, rather than being queued, because the queue of unprocessed MARS packets was too long. This means that MARS packets are arriving faster than they can be processed. Note that this event does not get counted in ELS, it is instead counted in the MARS console. The counters (kept per input network) can be read using the MARS>STATISTICS command, in the ''input packet overflows'' section.

Cause: This is often a symptom of a so-called ''MARS storm''. Some packets (usually an IP broadcast) arrive at hosts (usually a popular workstation) which do not recognize the destination address; they then attempt (in contravention of the Host specification) to forward the packet, but to do so they need the MARS mapping. Since they all receive the broadcast at the same time, they all attempt to forward the packet at the same time, and all do an MARS request at the same time.

Action: Prevail on the appropriate host manufacturer to bring their software into compliance with the specification. In the short term, it may be possible to disable the source of the packets, or cause it to use an address that the misbehaving hosts do recognize as a broadcast.

MARS.002

Level: UI-ERROR

Short Syntax: MARS.002 MARS Q dst is NULL (destId= *destQueue*) for nt *network ID*

Long Syntax: MARS.002 MARS queue destination is NULL (destId= *destQueue*) for net *network ID*

Description: A message was sent to the internal MARS processing routine with an invalid destination type or the destination queue was not initialized correctly.

MARS.003

Level: UI-ERROR

Short Syntax: MARS.003 ATM MARS marsSend net not sup or NULL channel detected (channel= *chaDest* nt *network ID*)

Long Syntax: MARS.003 ATM MARS marsSend net not supported or NULL channel detected (channel= *chaDest* net *network ID*)

Description: An outgoing MARS packet was received on a network which is not using MARS for address translation in any protocol or the channel for which the packet was to be sent on is NULL. check the information contained in the message to determine the cause of failure.

Cause: The gateway is misconfigured.

Action: Correct the configuration.

MARS.004

Level: UI-ERROR

Short Syntax: MARS.004 ATM MARS Rqst send failed rsn *reason_code* nt *network ID*

Long Syntax: MARS.004 ATM MARS transmission of request failed for reason *reason_code* net *network ID*

Description: An outgoing MARS request packet was dropped as the result of some problem internal problem. The reason_code gives the cause.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

MARS.005

Level: UI-ERROR

Short Syntax: MARS.005 ATM MARS marsSendControlList: Invalid input parms (listAddr= *listVal* dataAddr= *DataVal*)

Long Syntax: MARS.005 ATM MARS marsSendControlList: Invalid input parms (listAddr= *listVal* dataAddr= *DataVal*)

Description: A request to send a MARS packed on an outging control list was received. The input parameters for this request are invalid. Record the listAddr and dataAddr values and report problem if it continues.

MARS.006

Level: UI-ERROR

Short Syntax: MARS.006 ATM MARS *functionCall*: *msgDesc*.

Long Syntax: MARS.006 ATM MARS *functionCall*: *msgDesc*.

Description: An internal error occurred. The message contains where and why the error occurred.

MARS.008

Level: UI-ERROR

Short Syntax: MARS.008 ATM MARS marsSendMember: Invalid input parms (nodeAddr= *nodeVal* dataAddr= *DataVal*)

Long Syntax: MARS.008 ATM MARS marsSendMember: Invalid input parms (nodeAddr= *nodeVal* dataAddr= *DataVal*)

Description: A request to send a MARS packed on an outging channel was received. The input parameters for this request are invalid. Record the nodeAddr and dataAddr values and report problem if it continues.

MARS.009

Level: UI-ERROR

Short Syntax: MARS.009 ATM MARS marsControlListCreate: Invalid input parms (listAddr= *listAddr* nodeAddr= *nodeAddr* dataAddr= *DataAaddr*)

Long Syntax: MARS.009 ATM MARS marsControlListCreate: Invalid input parms (listAddr= *listAddr* nodeAddr= *nodeAddr* dataAddr= *DataAaddr*)

Description: A request to create a control list failed. The input parameters for this request are invalid. Record the listAddr, nodeAddr and dataAddr values and report problem if it continues.

MARS.010

Level: UI-ERROR

Short Syntax: MARS.010 ATM MARS PlaceCall Failure (rc= *retCode*): nt *network ID* for ATM addr = *atmAddr*

Long Syntax: MARS.010 ATM MARS PlaceCall Failure (rc= *retCode*): net *network ID* for destination ATM address = *atmAddr*

Description: While attempting to set up a SVC, the services of the device driver returned a value other than SUCCESS.

MARS.011

Level: U-TRACE

Short Syntax: MARS.011 ATM MARS PlaceCall Success: nt *network ID* for ATM addr = *atmAddr*

Long Syntax: MARS.011 ATM MARS PlaceCall Success: net *network ID* for ATM addr = *atmAddr*

Description: A call was successfully placed. This channel should show up on the new channel list. It has not yet been answered. When it is answered, a PlaceCallAck message will appear in the log.

MARS.012

Level: UI-ERROR

Short Syntax: MARS.012 ATM MARS marsControlListAddMember: Invalid input parms (listAddr= *listAddr* nodeAddr= *nodeAddr* dataAddr= *DataAaddr*)

Long Syntax: MARS.012 ATM MARS marsControlListAddMember: Invalid input parms (listAddr= *listAddr* nodeAddr= *nodeAddr* dataAddr= *DataAaddr*)

Description: A request to add a member to a control list failed. The input parameters for this request are invalid. Record the listAddr, nodeAddr and dataAddr values and report problem if it continues.

MARS.013

Level: UI-ERROR

Short Syntax: MARS.013 ATM MARS AddLeaf Failure (rc= *retCode*): nt *network ID* for ATM addr = *atmAddr*

Long Syntax: MARS.013 ATM MARS AddLeaf Failure (rc= *retCode*): net *network ID* for destination ATM address = *atmAddr*

Description: While attempting to add a leaf to an existing P2MP VC the services of the device driver returned a value other than SUCCESS.

MARS.014

Level: U-TRACE

Short Syntax: MARS.014 ATM MARS AddLeaf Success: vpi= *vpiVal* vci= *vciVal* nt *network ID* for ATM addr = *atmAddr*

Long Syntax: MARS.014 ATM MARS AddLeaf Success: vpi= *vpiVal* vci= *vciVal* net *network ID* for ATM addr = *atmAddr*

Description: An AddLeaf call was successfully placed. This leaf should show up on the channel list. It has not yet been answered. When it is answered, a AddLeafAck message will appear in the log.

MARS.015

Level: UI-ERROR

Short Syntax: MARS.015 ATM MARS marsControlListRemoveMember: Invalid input parms (listAddr= *listAddr* nodeAddr= *nodeAddr*)

Long Syntax: MARS.015 ATM MARS marsControlListRemoveMember: Invalid input parms (listAddr= *listAddr* nodeAddr= *nodeAddr*)

Description: A request to remove a member from a control list failed. The input parameters for this request are invalid. Record the listAddr and nodeAddr values and report problem if it continues.

MARS.016

Level: UI-ERROR

Short Syntax: MARS.016 ATM MARS HangupLeaf Failure (rc= *retCode*): vpi= *vpiVal* vci= *vciVal* for ATM addr = *atmAddr*

Long Syntax: MARS.016 ATM MARS HangupLeaf Failure (rc= *retCode*): vpi= *vpiVal* vci= *vciVal* for destination ATM address = *atmAddr*

Description: While attempting to remove a leaf from an existing P2MP VC the services of the device driver returned a value other than SUCCESS.

MARS.017

Level: U-TRACE

Short Syntax: MARS.017 ATM MARS HangupLeaf Success: vpi= *vpiVal* vci= *vciVal* for ATM addr = *atmAddr*

Long Syntax: MARS.017 ATM MARS HangupLeaf Success: vpi= *vpiVal* vci= *vciVal* for ATM addr = *atmAddr*

Description: A HangupLeaf call was successfully placed. This leaf should no longer show up on the channel list.

MARS.018

Level: UE-ERROR

Short Syntax: MARS.018 ATM MARS CloseDataPath failure(rc= *return_code* vpi= *vpiVal*, vci= *vciVal*) for ATM addr = *atmAddr*.

Long Syntax: MARS.018 ATM MARS CloseDataPath failure(rc= *return_code* vpi= *vpiVal*, vci= *vciVal*) for ATM address = *atmAddr*.

Description: When attempting to open up a data path with the specified parameters, a failure occured. The call will be hung up with the appropriate cause code.

MARS.019

Level: UI-ERROR

Short Syntax: MARS.019 ATM MARS HangupCall Failure (rc= *retCode*): vpi= *vpiVal* vci= *vciVal* for ATM addr = *atmAddr*

Long Syntax: MARS.019 ATM MARS HangupCall Failure (rc= *retCode*): vpi= *vpiVal* vci= *vciVal* for destination ATM address = *atmAddr*

Description: While attempting to remove a channel the services of the device driver returned a value other than SUCCESS.

MARS.020

Level: U-TRACE

Short Syntax: MARS.020 ATM MARS *functionCall*: *msgDesc*.

Long Syntax: MARS.020 ATM MARS *functionCall*: *msgDesc*.

Description: This is the action being performed by the MARS Server.

MARS.021

Level: UI-ERROR

Short Syntax: MARS.021 ATM MARS marsMcsCreateP2MPVC: Invalid input parms (mpp= *marsprt* channelp= *channel* grpaddr= *grpaddr* mep= *prtEnt*)

Long Syntax: MARS.021 ATM MARS marsMcsCreateP2MPVC: Invalid input parms (mpp= *marsprt* channelp= *channel* grpaddr= *grpaddr* mep= *prtEnt*)

Description: A request to create a data VC failed. The input parameters for this request are invalid. Record the mpp, channelp, grpaddr, and mep values and report problem if it continues.

MARS.022

Level: UI-ERROR

Short Syntax: MARS.022 ATM MARS *cmdType*: Invalid input parms (channelp= *channel* mnp= *marsnode*)

Long Syntax: MARS.022 ATM MARS *cmdType*: Invalid input parms (channelp= *channel* mnp= *marsnode*)

Description: A request to add/remove a member to a data P2MP VC failed. The input parameters for this request are invalid. Record the channelp and mnp values and report problem if it continues.

MARS.030

Level: U-INFO

Short Syntax: MARS.030 ATM MARS Net *devState* (dev num = *devNum*): ATM addr = *atmAddr*.

Long Syntax: MARS.030 ATM MARS Net *devState* (device number = *devNum*):ATM addr = *atmAddr*.

Description: This client has received a net up or down call as indicated in the message. All channels and calls will be cleared in the case of a Net Down call. Upon receiving a NetUp upcall, the interface will attempt to reestablish all calls.

MARS.031

Level: C-INFO

Short Syntax: MARS.031 ATM MARS AddrStateChg (*action*): ATM addr = *atmAddr*.

Long Syntax: MARS.031 ATM MARS AddrStateChg (*action*): ATM addr = *atmAddr*.

Description: This client has received an address state change from the switch. This means that the address ESI and SEL have been activated or deactivated depending on the message content. If activated the client can procede in setting up and receive calls to the switch. If deactivated all SVC connection will be cleared and but PVCs will remain operable.

MARS.032

Level: UE-ERROR

Short Syntax: MARS.032 ATM MARS AddrStateChg (*action*): ATM addr = *atmAddr*.

Long Syntax: MARS.032 ATM MARS AddrStateChg (*action*): ATM addr = *atmAddr*.

Description: This client has received an abnormal address state change from the switch. If refused a duplicate MAC address is already registered with the switch.

MARS.033

Level: C-INFO

Short Syntax: MARS.033 ATM MARS UNI Vers rcved: nt *network ID*

Long Syntax: MARS.033 ATM MARS UNI Vers rcved: net *network ID*

Description: This ATM client has received the UNI version supported from the switch.

MARS.034

Level: C-INFO

Short Syntax: MARS.034 ATM MARS Address Activation pending: nt *network ID*

Long Syntax: MARS.034 ATM MARS Address Activation pending: net *network ID*

Description: This client has initiated the sequence that registers the client ATM address with the switch. When the registration completes, another message of Address State change will be logged describing the status of the clients ATM address.

Action: No action required. This is normal processing.

MARS.035

Level: C-INFO

Short Syntax: MARS.035 ATM MARS Address Activation success: nt *network ID*

Long Syntax: MARS.035 ATM MARS Address Activation success: net *network ID*

Description: This client has been successful at activating an address.

MARS.036

Level: UI-ERROR

Short Syntax: MARS.036 ATM MARS GetAddrByHandle rc= *return_code*: nt *network ID*

Long Syntax: MARS.036 ATM MARS GetAddrByHandle rc= *return_code*: net *network ID*

Description: While attempting to get the address from the switch, an error was detected.

MARS.037

Level: UI-ERROR

Short Syntax: MARS.037 ATM MARS LlcOpenCallSap rc= *return_code*: nt *network ID*

Long Syntax: MARS.037 ATM MARS LlcOpenCallSap rc= *return_code*: net *network ID*

Description: While attempting to open a call sap, an

error was detected. A call sap is required in order to place or receive ATM calls to a remote destination.

MARS.038

Level: UI-ERROR

Short Syntax: MARS.038 ATM MARS atmMarsInit Registr failure (rc= *return_code*): nt *network ID*

Long Syntax: MARS.038 ATM MARS atmMarsInit Registr failure (rc= *return_code*): net *network ID*

Description: This client has failed to register as a user to the underlying device driver and net handler. This client will be inoperable.

Action: Reboot the router and contact the appropriate service personelle.

MARS.039

Level: C-INFO

Short Syntax: MARS.039 ATM MARS atmMarsInit Registr successfull: nt *network ID*

Long Syntax: MARS.039 ATM MARS atmMarsInit Registr successfull: net *network ID*

Description: This client has successfully registered with the underlying device driver and net handler. This is normal initialization.

MARS.041

Level: UI-ERROR

Short Syntax: MARS.041 ATM MARS atmMarsLecsListReport?:

Long Syntax: MARS.041 ATM MARS atmMarsLecsListReport?:

Description: An internal malfunction. The specified function was invoked on a classical MARS Server for which no such function is defined.

MARS.042

Level: U-INFO

Short Syntax: MARS.042 ATM MARS ReceiveCall (vpi= *vpiVal*, vci= *vciVal*) for ATM addr = *atmAddr*.

Long Syntax: MARS.042 ATM MARS ReceiveCall (vpi= *vpiVal*, vci= *vciVal*) for ATM address = *atmAddr*.

Description: A call was received by this client.

MARS.044

Level: UE-ERROR

Short Syntax: MARS.044 ATM MARS OpenDataPath failr(rc= *return_code* vpi= *vpiVal*, vci= *vciVal*) for ATM addr = *atmAddr*.

Long Syntax: MARS.044 ATM MARS OpenDataPath failr(rc= *return_code* vpi= *vpiVal*, vci= *vciVal*) for ATM address = *atmAddr*.

Description: When attempting to open up a data path with the specified parameters, a failure occured. The call will be hung up with the appropriate cause code.

MARS.045

Level: UE-ERROR

Short Syntax: MARS.045 ATM MARS atmRcvCallAck fail(rc= *return_code* vpi= *vpiVal*, vci= *vciVal*) for ATM addr = *atmAddr*.

Long Syntax: MARS.045 ATM MARS atmRcvCallAck fail(rc= *return_code* vpi= *vpiVal*, vci= *vciVal*) for ATM address = *atmAddr*.

Description: When attempting to acknowledge the incoming call, a failure occured.

Cause: The cause is an internal control block problem.

MARS.046

Level: C-INFO

Short Syntax: MARS.046 ATM MARS PlaceCallAck (vpi= *vpiNum*, vci= *vciNum*) for ATM addr = *atmAddr*.

Long Syntax: MARS.046 ATM MARS PlaceCallAck (vpi= *vpiNum*, vci= *vciNum*) for ATM address = *atmAddr*.

Description: A call that we have placed has been received and acknowledged by the remote destination. We will open up a data path to the remote side, and will begin transmitting and receiving on the VCC.

MARS.047

Level: UE-ERROR

Short Syntax: MARS.047 ATM MARS PlaceCallAck call parms mod.(vpi= *vpiNum*, vci= *vciNum*) for ATM addr = *atmAddr*.

Long Syntax: MARS.047 ATM MARS PlaceCallAck call parameters modified (vpi= *vpiNum*, vci= *vciNum*) for ATM address = *atmAddr*.

Description: A call that we have placed has been received and acknowledged by the remote destination but the original parms have been modified. The MARS server can not support modification of call parameters.

MARS.048

Level: U-INFO

Short Syntax: MARS.048 ATM MARS atmDisconnectCall: NULL CORRELATOR received

Long Syntax: MARS.048 ATM MARS atmDisconnectCall: NULL CORRELATOR received

Description: A call was released immediately before we received it.

MARS.049

Level: U-INFO

Short Syntax: MARS.049 ATM MARS DisconnectCall: (vpi= *vpiNum*, vci= *vciNum* type= *chanType*) for ATM addr = *atmAddr*.

Long Syntax: MARS.049 ATM MARS DisconnectCall: (vpi= *vpiNum*, vci= *vciNum* type= *chanType*) for ATM address = *atmAddr*.

Description: Either a call already active, or a call that we are placing has been released. The reason for the release is shown in additional ELS messages. This is a normal occurance. If the channel is required, we will reinitiate it. Control channels, for example are retried every 15 seconds until we connect to the server. The information in this message is the channel vpi/vci, and remote atm address of the channel that is being disconnected.

Cause: Either the network or the remote user has released the call.

MARS.050

Level: U-INFO

Short Syntax: MARS.050 ATM MARS DisconnectCall: rsn= *reason_code*, cause= *cause_code*, diagLen= *diag_len*, diagData[0]= *diag_data*

Long Syntax: MARS.050 ATM MARS DisconnectCall: rsn= *reason_code*, cause= *cause_code*, diagLen= *diag_len*, diagData[0]= *diag_data*

Description: The information in this message is the reason for which the call has been released.

MARS.052

Level: U-INFO

Short Syntax: MARS.052 ATM MARS DisconnectLeaf: rsn= *reason_code*, cause= *cause_code*, diagLen= *diag_len*, diagData[0]= *diag_data* vpi= *vcc_vpi*, vci= *vcc_vci*, LeafAtmAddr= *leaf_remote_atm_address*

Long Syntax: MARS.052 ATM MARS DisconnectLeaf: rsn= *reason_code*, cause= *cause_code*, diagLen= *diag_len*, diagData[0]= *diag_data* vpi= *vcc_vpi*, vci= *vcc_vci*, LeafAtmAddr= *leaf_remote_atm_address*

Description: The information in this message is the reason for which the leaf has been released. It also contains the channel vpi/vci for which this leaf was a member of along with the atm address of the leaf.

MARS.053

Level: U-INFO

Short Syntax: MARS.053 ATM MARS AddLeafAck: vpi= *vcc_vpi*, vci= *vcc_vci*, LeafAtmAddr= *leaf_remote_atm_address*

Long Syntax: MARS.053 ATM MARS AddLeafAck: vpi= *vcc_vpi*, vci= *vcc_vci*, LeafAtmAddr= *leaf_remote_atm_address*

Description: Confirms a successful addition of a new party to a point-to-multipoint call.

MARS.054

Level: UE-ERROR

Short Syntax: MARS.054 ATM MARS RcvFrame: Unknown *frameType* value= *protocolNum* nt *network ID*

Long Syntax: MARS.054 ATM MARS RcvFrame: Unknown *frameType* value= *protocolNum* net *network ID*

Description: A packet with an unknown protocol ID has been received off of the specified network. This may or may not be expected traffic. In any event, the packet will be discarded. No forwarding will occur.

MARS.055

Level: U-INFO

Short Syntax: MARS.055 ATM MARS This message is available for use

Long Syntax: MARS.055 ATM MARS This message is available for use

Description: This is only a placeholder.

MARS.056

Level: U-INFO

Short Syntax: MARS.056 ATM MARS This message is available for use

Long Syntax: MARS.056 ATM MARS This message is available for use

Description: This is only a placeholder.

MARS.057

Level: U-INFO

Short Syntax: MARS.057 ATM MARS This message is available for use

Long Syntax: MARS.057 ATM MARS This message is available for use

Description: This is only a placeholder.

MARS.058

Level: U-INFO

Short Syntax: MARS.058 ATM MARS joinMsg: MARS_JOIN for group *group* ignored. Registration pending.

Long Syntax: MARS.058 ATM MARS joinMsg: MARS_JOIN for group *group* ignored. Registration pending.

Description: ACK from add leaf for previous registration has not yet arrived.

MARS.059

Level: U-INFO

Short Syntax: MARS.059 ATM MARS joinMsg: MARS_JOIN for group *group* ignored. Not registered.

Long Syntax: MARS.059 ATM MARS joinMsg: MARS_JOIN for group *group* ignored. Not registered.

Description: Attempt to join a group but node has not previously registered.

MARS.060

Level: UI-ERROR

Short Syntax: MARS.060 ATM MARS remove_group: Removing a group but number of members = *numMembers*

Long Syntax: MARS.060 ATM MARS remove_group: Removing a group but number of members = *numMembers*

Description: A group is being removed however the number of members is not zero. This is an internal error indicating that a counter is incorrect. Group removal will continue.

MARS.061

Level: UI-ERROR

Short Syntax: MARS.061 ATM MARS mars_malloc: calloc of *numbytes* bytes failed, errno = *errno*

Long Syntax: MARS.061 ATM MARS mars_malloc: Attempt to calloc *numbytes* bytes has failed, errno = *errno*

Description: An attempt to obtain memory has failed.

MARS.062

Level: UI-ERROR

Short Syntax: MARS.062 ATM MARS mars_free: Address being freed is NULL

Long Syntax: MARS.062 ATM MARS mars_free: Address being freed is NULL

Description: The free storage subroutine is being called but the address is NULL. This is an internal error.

MARS.063

Level: UI-ERROR

Short Syntax: MARS.063 ATM MARS add_member: Adding a member but the group is NULL

Long Syntax: MARS.063 ATM MARS add_member: Adding a member but the group is NULL

Description: Attempting to add a member to a group but the group is NULL. This is an internal error.

MARS.064

Level: U-TRACE

Short Syntax: MARS.064 ATM MARS punch_mbr_holes: Group *group* needs to be hole punched

Long Syntax: MARS.064 ATM MARS punch_mbr_holes: Group *group* needs to be hole punched

Description: Exclude from the range those groups that the node is already a member of.

MARS.065

Level: U-INFO

Short Syntax: MARS.065 ATM MARS print_*functionCall*: 0x *proto* is an unknown protocol

Long Syntax: MARS.065 ATM MARS print_*functionCall*: 0x *proto* is an unknown protocol. It is ignored.

Description: A protocol was defined but not found in the server's protocol table. The protocol is ignored.

MARS.066

Level: U-INFO

Short Syntax: MARS.066 ATM MARS print_nodes: For protocol 0x *proto*, the number of active members = *mbrcnt*

Long Syntax: MARS.066 ATM MARS print_nodes: For protocol 0x *proto*, the number of active members = *mbrcnt*

Description: An informational message.

MARS.067

Level: U-INFO

Short Syntax: MARS.067 ATM MARS print_nodes: For protocol 0x *proto*, the number of removed members = *mbrcnt*

Long Syntax: MARS.067 ATM MARS print_nodes: For protocol 0x *proto*, the number of removed members = *mbrcnt*

Description: An informational message.

MARS.068

Level: U-INFO

Short Syntax: MARS.068 ATM MARS print_nodes: Server *serverAtmAddr* has joined these groups:

Long Syntax: MARS.068 ATM MARS print_nodes: Server *serverAtmAddr* has joined these groups:

Description: An informational message.

MARS.069

Level: U-INFO

Short Syntax: MARS.069 ATM MARS print_nodes: Host *hostAtmAddr* has joined these groups:

Long Syntax: MARS.069 ATM MARS print_nodes: Host *hostAtmAddr* has joined these groups:

Description: An informational message.

MARS.070

Level: U-INFO

Short Syntax: MARS.070 ATM MARS print_nodes: *group*

Long Syntax: MARS.070 ATM MARS print_nodes: *group*

Description: An informational message.

MARS.071

Level: U-INFO

Short Syntax: MARS.071 ATM MARS print_nodes: Server *serverAtmAddr* has not joined any groups

Long Syntax: MARS.071 ATM MARS print_nodes: Server *serverAtmAddr* has not joined any groups

Description: An informational message.

MARS.072

Level: U-INFO

Short Syntax: MARS.072 ATM MARS print_nodes: Host *hostAtmAddr* has not joined any groups

Long Syntax: MARS.072 ATM MARS print_nodes: Host *hostAtmAddr* has not joined any groups

Description: An informational message.

MARS.073

Level: U-INFO

Short Syntax: MARS.073 ATM MARS print_groups: Protocol = 0x *proto*

Long Syntax: MARS.073 ATM MARS print_groups: Protocol = 0x *proto*

Description: An informational message.

MARS.074

Level: U-INFO

Short Syntax: MARS.074 ATM MARS print_groups: Group *group* has these servers:

Long Syntax: MARS.074 ATM MARS print_groups: Group *group* has these servers:

Description: An informational message.

MARS.075

Level: U-INFO

Short Syntax: MARS.075 ATM MARS print_groups: Group *group* has these hosts:

Long Syntax: MARS.075 ATM MARS print_groups: Group *group* has these hosts:

Description: An informational message.

MARS.076

Level: U-INFO

Short Syntax: MARS.076 ATM MARS print_groups: Server *serverAtmAddr*

Long Syntax: MARS.076 ATM MARS print_groups: Server *serverAtmAddr*

Description: An informational message.

MARS.077

Level: U-INFO

Short Syntax: MARS.077 ATM MARS print_groups: Host *hostAtmAddr*

Long Syntax: MARS.077 ATM MARS print_groups: Host *hostAtmAddr*

Description: An informational message.

MARS.078

Level: U-INFO

Short Syntax: MARS.078 ATM MARS print_groups: Group *group* has no members

Long Syntax: MARS.078 ATM MARS print_groups:

Group *group* has no members

Description: An informational message.

MARS.079

Level: U-INFO

Short Syntax: MARS.079 ATM MARS mservMsg: MARS_MSERV for group *group* ignored. Registration pending.

Long Syntax: MARS.079 ATM MARS mservMsg: MARS_MSERV for group *group* ignored. Registration Pending.

Description: Add leaf ACK for previous registration has not yet arrived.

MARS.080

Level: U-TRACE

Short Syntax: MARS.080 ATM MARS *functionCall*: Hole punched pair = *group*

Long Syntax: MARS.080 ATM MARS *functionCall*: Hole punched pair = *group*

Description: An informational message.

MARS.081

Level: U-TRACE

Short Syntax: MARS.081 ATM MARS free_punset: Multi group = *group*

Long Syntax: MARS.081 ATM MARS free_punset: Multi group = *group*

Description: An informational message.

MARS.082

Level: UI-ERROR

Short Syntax: MARS.082 ATM MARS *functionCall*: Illegal ATM address.

Long Syntax: MARS.082 ATM MARS *functionCall*: Illegal ATM address.

Description: Subroutine marsChkAtmAddr indicated ATM address was illegal.

MARS.083

Level: U-INFO

Short Syntax: MARS.083 ATM MARS leaveMsg: MARS_LEAVE for group *group* ignored.

Long Syntax: MARS.083 ATM MARS leaveMsg: MARS_LEAVE for group *group* ignored.

Description: We must silently drop the message if the copy is not zero or the message contains more than one <min,max> pair.

MARS.084

Level: UI-ERROR

Short Syntax: MARS.084 ATM MARS leaveMsg: Undefined protocol (0x *proto*) in MARS_LEAVE, ignored.

Long Syntax: MARS.084 ATM MARS leaveMsg: Undefined protocol (0x *proto*) in MARS_LEAVE, ignored.

Description: Unknown protocol in message.

MARS.085

Level: U-INFO

Short Syntax: MARS.085 ATM MARS leaveMsg: MARS_LEAVE from host *hostAtmAddr* was not registered.

Long Syntax: MARS.085 ATM MARS leaveMsg: MARS_LEAVE from host *hostAtmAddr* was not registered.

Description: Unable to find the host in the table of nodes.

MARS.086

Level: U-TRACE

Short Syntax: MARS.086 ATM MARS leaveMsg: Processing MARS_LEAVE deregister from host *hostAtmAddr*

Long Syntax: MARS.086 ATM MARS leaveMsg: Processing MARS_LEAVE deregister from host *hostAtmAddr*

Description: An informational trace message.

MARS.087

Level: U-TRACE

Short Syntax: MARS.087 ATM MARS leaveMsg: Processing MARS_LEAVE for group *group*

Long Syntax: MARS.087 ATM MARS leaveMsg: Processing MARS_LEAVE for group *group*

Description: An informational trace message.

MARS.088

Level: U-INFO

Short Syntax: MARS.088 ATM MARS leaveMsg: MARS_LEAVE for group *group* ignored.

Long Syntax: MARS.088 ATM MARS leaveMsg: MARS_LEAVE for group *group* ignored.

Description: If this is an MCS or the cluster member has not previously registered then drop the message.

MARS.089

Level: U-INFO

Short Syntax: MARS.089 ATM MARS leaveMsg: MARS_LEAVE for group *group* not found.

Long Syntax: MARS.089 ATM MARS leaveMsg: MARS_LEAVE for group *group* not found.

Description: The leaving node is not a member of the specified group.

MARS.090

Level: U-TRACE

Short Syntax: MARS.090 ATM MARS cluster_leave: Group leave = *group*

Long Syntax: MARS.090 ATM MARS cluster_leave: Group leave = *group*

Description: The group contained within the MARS_LEAVE message.

MARS.091

Level: U-TRACE

Short Syntax: MARS.091 ATM MARS make_newmsg: Hole punched pair *group* to new msg.

Long Syntax: MARS.091 ATM MARS make_newmsg: Hole punched pair *group* to new msg.

Description: Results of hole punching.

MARS.092

Level: U-TRACE

Short Syntax: MARS.092 ATM MARS multi_group: Group *group* was MCS holepunched.

Long Syntax: MARS.092 ATM MARS multi_group: Group *group* was MCS holepunched.

Description: Results of hole punching for MCSs.

MARS.093

Level: U-TRACE

Short Syntax: MARS.093 ATM MARS multi_group: Hole punched pair *group* to original msg.

Long Syntax: MARS.093 ATM MARS multi_group: Hole punched pair *group* to original msg.

Description: Trace message.

MARS.094

Level: U-INFO

Short Syntax: MARS.094 ATM MARS cluster_join: group *group* gets layer3grp reset.

Long Syntax: MARS.094 ATM MARS cluster_join: group *group* gets layer3grp reset.

Description: Informational.

MARS.095

Level: U-INFO

Short Syntax: MARS.095 ATM MARS cluster_join: group *group* is an overlap, ignored.

Long Syntax: MARS.095 ATM MARS cluster_join: group *group* is an overlap, ignored.

Description: Informational.

MARS.096

Level: U-TRACE

Short Syntax: MARS.096 ATM MARS *functionCall*: Holepunching produced *ctr* pairs.

Long Syntax: MARS.096 ATM MARS *functionCall*: Holepunching produced *ctr* pairs.

Description: Trace message.

MARS.097

Level: U-TRACE

Short Syntax: MARS.097 ATM MARS *functionCall*: Hole punched pair = *group*

Long Syntax: MARS.097 ATM MARS *functionCall*: Hole punched pair = *group*

Description: Trace message.

MARS.098

Level: U-TRACE

Short Syntax: MARS.098 ATM MARS joinMsg: MARS_JOIN for group *group* ignored.

Long Syntax: MARS.098 ATM MARS joinMsg: MARS_JOIN for group *group* ignored.

Description: Trace message.

MARS.099

Level: UI-ERROR

Short Syntax: MARS.099 ATM MARS joinMsg: Undefined protocol (0x *proto*) in MARS_JOIN, ignored.

Long Syntax: MARS.099 ATM MARS joinMsg:

Undefined protocol (0x *proto*) in MARS_JOIN, ignored.

Description: Unknown protocol in message.

MARS.100

Level: U-INFO

Short Syntax: MARS.100 ATM MARS joinMsg: MARS_JOIN from host *hostAtmAddr* is duplicate registration.

Long Syntax: MARS.100 ATM MARS joinMsg: MARS_JOIN from host *hostAtmAddr* is duplicate registration.

Description: Duplicate join.

MARS.101

Level: U-TRACE

Short Syntax: MARS.101 ATM MARS joinMsg: Processing MARS_JOIN register from host *hostAtmAddr*

Long Syntax: MARS.101 ATM MARS joinMsg: Processing MARS_JOIN register from host *hostAtmAddr*

Description: Trace message.

MARS.102

Level: U-INFO

Short Syntax: MARS.102 ATM MARS joinMsg: Cluster registration has failed.

Long Syntax: MARS.102 ATM MARS joinMsg: Cluster registration has failed.

Description: Informational message.

MARS.103

Level: U-TRACE

Short Syntax: MARS.103 ATM MARS joinMsg: Processing MARS_JOIN for group *group*

Long Syntax: MARS.103 ATM MARS joinMsg: Processing MARS_JOIN for group *group*

Description: Trace message.

MARS.104

Level: U-INFO

Short Syntax: MARS.104 ATM MARS joinMsg: MARS_JOIN for group *group* ignored.

Long Syntax: MARS.104 ATM MARS joinMsg: MARS_JOIN for group *group* ignored.

Description: Informational message.

MARS.105

Level: U-INFO

Short Syntax: MARS.105 ATM MARS joinMsg: MARS_JOIN for group *group* is a duplicate.

Long Syntax: MARS.105 ATM MARS joinMsg: MARS_JOIN for group *group* is a duplicate.

Description: Informational message.

MARS.106

Level: U-TRACE

Short Syntax: MARS.106 ATM MARS multi_group: No holes were punched in *group*

Long Syntax: MARS.106 ATM MARS multi_group: No holes were punched in *group*

Description: Trace message.

MARS.107

Level: UI-ERROR

Short Syntax: MARS.107 ATM MARS cluster_join: No group specified in MARS_JOIN message

Long Syntax: MARS.107 ATM MARS cluster_join: No group specified in MARS_JOIN message

Description: Attempt to join a group but no group specified.

MARS.108

Level: U-TRACE

Short Syntax: MARS.108 ATM MARS cluster_join: Group *group* now has *ctr* members

Long Syntax: MARS.108 ATM MARS cluster_join: Group *group* now has *ctr* members

Description: Trace message.

MARS.109

Level: UI-ERROR

Short Syntax: MARS.109 ATM MARS cluster_register: *io_subroutine_name* rc = 0x *rc*

Long Syntax: MARS.109 ATM MARS cluster_register: *io_subroutine_name* rc = 0x *rc*

Description: I/O error return code.

MARS.110

Level: U-INFO

Short Syntax: MARS.110 ATM MARS joinMsg: Cluster join has failed.

Long Syntax: MARS.110 ATM MARS joinMsg: Cluster join has failed.

Description: Informational message.

MARS.111

Level: U-TRACE

Short Syntax: MARS.111 ATM MARS *functionCall*: Holepunching produced a NULL pair.

Long Syntax: MARS.111 ATM MARS *functionCall*: Holepunching produced a NULL pair.

Description: Trace message.

MARS.112

Level: UI-ERROR

Short Syntax: MARS.112 ATM MARS glrequestMsg: Undefined protocol (0x *proto*) in MARS_GROUPLIST_REQUEST, ignored

Long Syntax: MARS.112 ATM MARS glrequestMsg: Undefined protocol (0x *proto*) in MARS_GROUPLIST_REQUEST, ignored

Description: Unknown protocol in message.

MARS.113

Level: U-INFO

Short Syntax: MARS.113 ATM MARS glrequestMsg: MARS_GROUPLIST_REQUEST from host *hostAtmAddr* not registered

Long Syntax: MARS.113 ATM MARS glrequestMsg: MARS_GROUPLIST_REQUEST from host *hostAtmAddr* not registered

Description: Informational message.

MARS.114

Level: U-TRACE

Short Syntax: MARS.114 ATM MARS send_reply: Sending *ctr* protocol addresses

Long Syntax: MARS.114 ATM MARS send_reply: Sending *ctr* protocol addresses

Description: Trace message.

MARS.115

Level: U-TRACE

Short Syntax: MARS.115 ATM MARS glreply: Group request = *group*

Long Syntax: MARS.115 ATM MARS glreply: Group request = *group*

Description: Trace message.

MARS.116

Level: U-TRACE

Short Syntax: MARS.116 ATM MARS glreply: Group found = *group*

Long Syntax: MARS.116 ATM MARS glreply: Group found = *group*

Description: Trace message.

MARS.117

Level: U-TRACE

Short Syntax: MARS.117 ATM MARS glreply: Group found but member is not Layer 3

Long Syntax: MARS.117 ATM MARS glreply: Group found but member is not Layer 3

Description: Trace message.

MARS.118

Level: U-INFO

Short Syntax: MARS.118 ATM MARS mservMsg: MARS_MSERV for group *group* ignored

Long Syntax: MARS.118 ATM MARS mservMsg: MARS_MSERV for group *group* ignored

Description: Informational message.

MARS.119

Level: UI-ERROR

Short Syntax: MARS.119 ATM MARS mservMsg: Undefined protocol (0x *proto*) in MARS_MSERV, ignored.

Long Syntax: MARS.119 ATM MARS mservMsg: Undefined protocol (0x *proto*) in MARS_MSERV, ignored.

Description: Unknown protocol in message.

MARS.120

Level: U-INFO

Short Syntax: MARS.120 ATM MARS mservMsg: MARS_MSERV from server *serverAtmAddr* is duplicate registration

Long Syntax: MARS.120 ATM MARS mservMsg: MARS_MSERV from server *serverAtmAddr* is duplicate registration

Description: Informational message.

MARS.121

Level: U-TRACE

Short Syntax: MARS.121 ATM MARS mservMsg: Processing MARS_MSERV register from server *serverAtmAddr*

Long Syntax: MARS.121 ATM MARS mservMsg: Processing MARS_MSERV register from server *serverAtmAddr*

Description: Informational message.

MARS.122

Level: U-TRACE

Short Syntax: MARS.122 ATM MARS mservMsg: Processing MARS_MSERV for group *group*

Long Syntax: MARS.122 ATM MARS mservMsg: Processing MARS_MSERV for group *group*

Description: Trace message.

MARS.123

Level: U-INFO

Short Syntax: MARS.123 ATM MARS mservMsg: MARS_MSERV for group *group* ignored. Not registered.

Long Syntax: MARS.123 ATM MARS mservMsg: MARS_MSERV for group *group* ignored. Not registered.

Description: Informational message.

MARS.124

Level: U-INFO

Short Syntax: MARS.124 ATM MARS mservMsg: MARS_MSERV for group *group* is a duplicate

Long Syntax: MARS.124 ATM MARS mservMsg: MARS_MSERV for group *group* is a duplicate

Description: Informational message.

MARS.125

Level: U-INFO

Short Syntax: MARS.125 ATM MARS mservMsg: Server registration has failed

Long Syntax: MARS.125 ATM MARS mservMsg: Server registration has failed

Description: Informational message.

MARS.126

Level: U-INFO

Short Syntax: MARS.126 ATM MARS unservMsg: MARS_UNSERV for group *group* is a copy, ignored

Long Syntax: MARS.126 ATM MARS unservMsg: MARS_UNSERV for group *group* is a copy, ignored

Description: Informational message.

MARS.127

Level: UI-ERROR

Short Syntax: MARS.127 ATM MARS unservMsg: Undefined protocol (0x *proto*) in MARS_UNSERV, ignored

Long Syntax: MARS.127 ATM MARS unservMsg: Undefined protocol (0x *proto*) in MARS_UNSERV, ignored

Description: Unknown protocol in message.

MARS.128

Level: U-INFO

Short Syntax: MARS.128 ATM MARS unservMsg: MARS_UNSERV from server *serverAtmAddr* not registered

Long Syntax: MARS.128 ATM MARS unservMsg: MARS_UNSERV from server *serverAtmAddr* not registered

Description: Informational message.

MARS.129

Level: U-TRACE

Short Syntax: MARS.129 ATM MARS unservMsg: Processing MARS_UNSERV deregister from server *serverAtmAddr*

Long Syntax: MARS.129 ATM MARS unservMsg: Processing MARS_UNSERV deregister from server *serverAtmAddr*

Description: Trace message.

MARS.130

Level: U-TRACE

Short Syntax: MARS.130 ATM MARS unserv: Processing MARS_UNSERV for group *group*

Long Syntax: MARS.130 ATM MARS unserv: Processing MARS_UNSERV for group *group*

Description: Trace message.

MARS.131

Level: UE-ERROR

Short Syntax: MARS.131 ATM MARS msg_handler: Unrecognized tlv for *mars_message* message

Long Syntax: MARS.131 ATM MARS msg_handler:

Unrecognized tlv for *mars_message* message

Description: Drop message and give error message.

MARS.132

Level: U-INFO

Short Syntax: MARS.132 ATM MARS msg_handler: *mars_message* is an unexpected message, ignored

Long Syntax: MARS.132 ATM MARS msg_handler: *mars_message* is an unexpected message, ignored

Description: Informational message.

MARS.133

Level: U-TRACE

Short Syntax: MARS.133 ATM MARS send_bkups: Sending a redirect msg with *ctr* addrs

Long Syntax: MARS.133 ATM MARS send_bkups: Sending a redirect msg with *ctr* addrs

Description: Trace message.

MARS.134

Level: U-TRACE

Short Syntax: MARS.134 ATM MARS redirmap: Timer cancelled, all leafs on CCVC are gone

Long Syntax: MARS.134 ATM MARS redirmap: Timer cancelled, all leafs on CCVC are gone

Description: Trace message.

MARS.135

Level: UI-ERROR

Short Syntax: MARS.135 ATM MARS requestMsg: Undefined protocol (0x *proto*) in MARS_REQUEST, ignored

Long Syntax: MARS.135 ATM MARS requestMsg: Undefined protocol (0x *proto*) in MARS_REQUEST, ignored

Description: Unknown protocol in message.

MARS.136

Level: U-INFO

Short Syntax: MARS.136 ATM MARS requestMsg: MARS_REQUEST from unregistered host *hostAtmAddr*

Long Syntax: MARS.136 ATM MARS requestMsg: MARS_REQUEST from unregistered host *hostAtmAddr*

Description: Host has not previously registered. Ignore the message.

MARS.137

Level: U-TRACE

Short Syntax: MARS.137 ATM MARS requestMsg: Group = *group*

Long Syntax: MARS.137 ATM MARS requestMsg: Group = *group*

Description: Trace message.

MARS.138

Level: U-TRACE

Short Syntax: MARS.138 ATM MARS *functionCall*: Sending *mars_opcode* on *vctype*

Long Syntax: MARS.138 ATM MARS *functionCall*: Sending *mars_opcode* on *vctype*

Description: Trace message.

MARS.139

Level: UI-ERROR

Short Syntax: MARS.139 ATM MARS *functionCall*: *io_subroutine_name* rc = 0x *rc*

Long Syntax: MARS.139 ATM MARS *functionCall*: *io_subroutine_name* rc = 0x *rc*

Description: I/O error return code.

MARS.140

Level: U-TRACE

Short Syntax: MARS.140 ATM MARS marsTimerInit: Using default of *defaultValue* for redirect timer

Long Syntax: MARS.140 ATM MARS marsTimerInit: Using default of *defaultValue* for redirect timer

Description: Trace message.

MARS.141

Level: U-TRACE

Short Syntax: MARS.141 ATM MARS marsTimerInit: redirect timer from configuration = *timerValue*

Long Syntax: MARS.141 ATM MARS marsTimerInit: redirect timer from configuration = *timerValue*

Description: Trace message.

MARS.143

Level: U-TRACE

Short Syntax: MARS.143 ATM MARS marsListCleanUp: Purge of *typeVc* about to begin

Long Syntax: MARS.143 ATM MARS

marsListCleanUp: Purge of *typeVc* about to begin

Description: Trace message.

MARS.144

Level: U-INFO

Short Syntax: MARS.144 ATM MARS
marsInstanceCleanUp: Unknown instance

Long Syntax: MARS.144 ATM MARS
marsInstanceCleanUp: Unknown instance

Description: Cleanup halted.

Chapter 77. Multicast Forwarding Cache (MFC)

This chapter describes Multicast Forwarding Cache (MFC) messages. For information on message content and how to use the message, refer to the Introduction.

MFC.001

Level: UI-ERROR

Short Syntax: MFC.001 No buf for IGMP poll, ifc *IGMP_interface*

Long Syntax: MFC.001 No buffer to send IGMP Host Membership Query on interface *IGMP_interface*

Description: An IGMP Host Membership Query could not be sent out the specified interface, due to buffer shortages. No attempt will be made to send another one until the next poll interval elapses.

Cause: Not enough memory to support this configuration and traffic.

Action: Check memory statistics in GWCON to verify packet buffer level. Upgrade for more memory, or disable unnecesary forwarders/protocols or get more memory.

MFC.002

Level: UI-ERROR

Short Syntax: MFC.002 IGMP poll fails, ifc *IGMP_interface* rsn *failure_code*

Long Syntax: MFC.002 Can't send IGMP Host Membership Query on interface *IGMP_interface* reason: *failure_code*

Description: An IGMP Host Membership Query could not be sent out the specified interface, due to the specified reason. No attempt will be made to send another one until the next poll interval elapses.

Cause: The net handler for the interface identified failed to send the poll for the reason (code) indicated.

Action: Check the reason code issued with this message, and correct the problem.

MFC.003

Level: P-TRACE

Short Syntax: MFC.003 Rcvd IGMP Report *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: MFC.003 Received IGMP Host Membership Report *IP_source* -> *IP_destination*, net *network ID*

Description: An IGMP Host Membership Report has been received on the specified interface.

MFC.005

Level: UE-ERROR

Short Syntax: MFC.005 Bad IGMP xsum *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: MFC.005 Received bad IGMP checksum, *IP_source* -> *IP_destination* net *network ID*

Description: An IGMP message has been received having a bad IGMP checksum. The message is discarded.

MFC.007

Level: UE-ERROR

Short Syntax: MFC.007 Unexp IGMP Query *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: MFC.007 Unexpected IGMP Host Membership Query, *IP_source* -> *IP_destination* net *network ID*

Description: An IGMP Host Membership Query has been received on an interface where the receiving router itself is sending Host Membership Queries (i.e., the router itself is the Designated Router). This is unexpected. Host Membership Queries are ignored in any case.

MFC.008

Level: P-TRACE

Short Syntax: MFC.008 Rcvd IGMP Query *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: MFC.008 Received IGMP Host Membership Query, *IP_source* -> *IP_destination* net *network ID*

Description: An IGMP Host Membership Query has been received. These are ignored by multicast routers.

MFC.009

Level: P-TRACE

Short Syntax: MFC.009 Rcvd dgram *IP_source* -> *IP_destination*, from *receiving_interface*

Long Syntax: MFC.009 Received IP multicast

datagram, *IP_source* -> *IP_destination*, from *receiving_interface*

Description: An IP datagram has been received that has a class D address, indicating IP multicast. An attempt will be made to forward the datagram.

MFC.010

Level: P-TRACE

Short Syntax: MFC.010 Fwrd dgram *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: MFC.010 Forwarded IP multicast datagram, *IP_source* -> *IP_destination*, net *network ID*

Description: An IP datagram has been forwarded out the specified interface as a data-link multicast.

MFC.011

Level: P-TRACE

Short Syntax: MFC.011 Fwrd dgram *IP_source* -> *IP_destination*, nbr *IP_gw_address*

Long Syntax: MFC.011 Forwarded IP multicast datagram, *IP_source* -> *IP_destination*, neighbor *IP_gw_address*

Description: An IP datagram has been forwarded to a specific neighbor, as a data-link unicast.

MFC.012

Level: P-TRACE

Short Syntax: MFC.012 Local delivery, *IP_source* -> *IP_destination*

Long Syntax: MFC.012 Local delivery of multicast datagram, *IP_source* -> *IP_destination*

Description: An IP datagram has been delivered to one of the router's internal applications.

MFC.013

Level: UE-ERROR

Short Syntax: MFC.013 Bad IP option, *IP_source* -> *IP_destination*

Long Syntax: MFC.013 Multicast datagram discarded due to bad option, *IP_source* -> *IP_destination*

Description: An IP multicast datagram has been received, containing a bad IP option (misformatted or inappropriate for multicast). The datagram is discarded w/o returning an ICMP message.

MFC.014

Level: UE-ERROR

Short Syntax: MFC.014 Can't fwd *IP_source* -> *IP_destination*, rsn: *reason*

Long Syntax: MFC.014 Can't forward multicast *IP_source* -> *IP_destination*, due to reason : *reason*

Description: An IP multicast datagram has not been forwarded, due to the specified reason.

MFC.015

Level: P-TRACE

Short Syntax: MFC.015 Lcl orig *IP_source* -> *IP_destination*

Long Syntax: MFC.015 Locally originated multicast, *IP_source* -> *IP_destination*

Description: An IP datagram has been originated by one of the router's internal applications; an attempt is being made to forward it. Such datagrams are always forwarded out the interface associated with the packet source (if any), regardless of any other forwarding decision.

MFC.016

Level: UI-ERROR

Short Syntax: MFC.016 MARS Local Join failed, ifc *IGMP_interface* rsn *failure_code*

Long Syntax: MFC.016 Request to send MARS Join on interface *IGMP_interface* failed with reason code: *failure_code*

Description: An MARS Local Join Request could not be sent out the specified interface, due to the specified reason.

Cause: The MARS Client for the interface identified failed to send the request for the reason (code) indicated.

Action: Check the reason code issued with this message, and correct the problem.

MFC.017

Level: UI-ERROR

Short Syntax: MFC.017 MARS Local Leave failed, ifc *IGMP_interface* rsn *failure_code*

Long Syntax: MFC.017 Request to send MARS Leave on interface *IGMP_interface* failed with reason code: *failure_code*

Description: An MARS Local Leave Request could not be sent out the specified interface, due to the specified reason.

Cause: The MARS Client for the interface identified

failed to send the request for the reason (code) indicated.

Action: Check the reason code issued with this message, and correct the problem.

MFC.018

Level: P-TRACE

Short Syntax: MFC.018 Rcvd IGMP Leave *IP_source* -> *IP_destination*, group *IP_group* nt *network ID*

Long Syntax: MFC.018 Received IGMP Leave *IP_source* -> *IP_destination*, group *IP_group* net *network ID*

Description: An IGMP Leave message has been received on the specified interface.

MFC.019

Level: U-INFO

Short Syntax: MFC.019 Rcvd IGMPv2 Query in IGMPv1 mode *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: MFC.019 Received IGMPv2 Membership Query in IGMPv1 mode, *IP_source* -> *IP_destination* net *network ID*

Description: An IGMPv2 Host Membership Query has been received and the router is configured for IGMPv1.

MFC.020

Level: U-INFO

Short Syntax: MFC.020 Rcvd IGMPv1 Query in IGMPv2 mode *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: MFC.020 Received IGMPv1 Membership Query in IGMPv2 mode, *IP_source* -> *IP_destination* net *network ID*

Description: An IGMPv1 Host Membership Query has been received and the router is configured for IGMPv2.

Chapter 78. Multicast Forwarding Cache v6 (MFC6)

This chapter describes Multicast Forwarding Cache v6 (MFC6) messages. For information on message content and how to use the message, refer to the Introduction.

MFC6.001

Level: UI-ERROR

Short Syntax: MFC6.001 No buf for MLD poll, ifc *MLD_interface*

Long Syntax: MFC6.001 No buffer to send MLD Query on interface *MLD_interface*

Description: An MLD Query could not be sent out the specified interface, due to buffer shortages. No attempt will be made to send another one until the next poll interval elapses.

Cause: Not enough memory to support this configuration and traffic.

Action: Check memory statistics in GWCON to verify packet buffer level. Upgrade for more memory, or disable unnecesary forwarders/protocols or get more memory.

MFC6.002

Level: UI-ERROR

Short Syntax: MFC6.002 MLD poll fails, ifc *MLD_interface* rsn *failure_code*

Long Syntax: MFC6.002 Can't send MLD Query on interface *MLD_interface* reason: *failure_code*

Description: An MLD Query could not be sent out the specified interface, due to the specified reason. No attempt will be made to send another one until the next poll interval elapses.

Cause: The net handler for the interface identified failed to send the poll for the reason (code) indicated.

Action: Check the reason code issued with this message, and correct the problem.

MFC6.003

Level: P-TRACE

Short Syntax: MFC6.003 Rcvd MLD Report *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: MFC6.003 Received MLD Report *IP_source* -> *IP_destination*, net *network ID*

Description: An MLD Report has been received on the specified interface.

MFC6.004

Level: UE-ERROR

Short Syntax: MFC6.004 Unexp MLD Query *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: MFC6.004 Unexpected MLD Query, *IP_source* -> *IP_destination* net *network ID*

Description: An MLD Query has been received on an interface where the receiving router itself is sending Host Membership Queries (i.e., the router itself is the Designated Router). This is unexpected. Host Membership Queries are ignored in any case.

MFC6.005

Level: P-TRACE

Short Syntax: MFC6.005 Rcvd MLD Query *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: MFC6.005 Received MLD Query, *IP_source* -> *IP_destination* net *network ID*

Description: An MLD Query has been received. These are ignored by multicast routers.

MFC6.006

Level: P-TRACE

Short Syntax: MFC6.006 Rcvd dgram *IP_source* -> *IP_destination*, from *receiving_interface*

Long Syntax: MFC6.006 Received IP multicast datagram, *IP_source* -> *IP_destination*, from *receiving_interface*

Description: An IP datagram has been received that has a class D address, indicating IP multicast. An attempt will be made to forward the datagram.

MFC6.007

Level: P-TRACE

Short Syntax: MFC6.007 Fwrd dgram *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: MFC6.007 Forwarded IP multicast datagram, *IP_source* -> *IP_destination*, net *network ID*

Description: An IP datagram has been forwarded out the specified interface as a data-link multicast.

MFC6.008

Level: P-TRACE

Short Syntax: MFC6.008 Fwrd dgram *IP_source* -> *IP_destination*, nbr *IP_gw_address*

Long Syntax: MFC6.008 Forwarded IP multicast datagram, *IP_source* -> *IP_destination*, neighbor *IP_gw_address*

Description: An IP datagram has been forwarded to a specific neighbor, as a data-link unicast.

MFC6.009

Level: P-TRACE

Short Syntax: MFC6.009 Local delivery, *IP_source* -> *IP_destination*

Long Syntax: MFC6.009 Local delivery of multicast datagram, *IP_source* -> *IP_destination*

Description: An IP datagram has been delivered to one of the router's internal applications.

MFC6.010

Level: UE-ERROR

Short Syntax: MFC6.010 Bad IP option, *IP_source* -> *IP_destination*

Long Syntax: MFC6.010 Multicast datagram discarded due to bad option, *IP_source* -> *IP_destination*

Description: An IP multicast datagram has been received, containing a bad IP option (misformatted or inappropriate for multicast). The datagram is discarded w/o returning an ICMP message.

MFC6.011

Level: UE-ERROR

Short Syntax: MFC6.011 Can't fwd *IP_source* -> *IP_destination*, rsn: *reason*

Long Syntax: MFC6.011 Can't forward multicast *IP_source* -> *IP_destination*, due to reason : *reason*

Description: An IP multicast datagram has not been forwarded, due to the specified reason.

MFC6.012

Level: P-TRACE

Short Syntax: MFC6.012 Lcl orig *IP_source* -> *IP_destination*

Long Syntax: MFC6.012 Locally originated multicast, *IP_source* -> *IP_destination*

Description: An IP datagram has been originated by one of the router's internal applications; an attempt is being made to forward it. Such datagrams are always forwarded out the interface associated with the packet source (if any), regardless of any other forwarding decision.

MFC6.013

Level: P-TRACE

Short Syntax: MFC6.013 Rcvd MLD reg for *IP_source*

Long Syntax: MFC6.013 Received MLD Register Request for group *IP_source*

Description: An MLD Register Request has been received from an internal application.

MFC6.014

Level: P-TRACE

Short Syntax: MFC6.014 Rcvd MLD Done *IP_source* -> *IP_destination*, group *IP_group* nt *network ID*

Long Syntax: MFC6.014 Received MLD Done *IP_source* -> *IP_destination*, group *IP_group* net *network ID*

Description: An MLD Done message has been received on the specified interface.

MFC6.015

Level: UI-ERROR

Short Syntax: MFC6.015 Bad dp trie key delete, fnc : *function*

Long Syntax: MFC6.015 The deletion of a dp trie key failed in function : *function*

Description: A function attempted to delete a key from a dp trie data structure.

Cause: The specified key could not be found in the dp trie.

Action: Contact your IBM service representative.

Chapter 79. Multilink PPP (MLP)

This chapter describes Multilink PPP (MLP) messages. For information on message content and how to use the message, refer to the Introduction.

MLP.001

Level: P-TRACE

Short Syntax: MLP.001 MP Rcv *bytes* byt num= *seqno*,M= *M*,BE= *BE_bits* lng?= *long* nt *network*

Long Syntax: MLP.001 MP Rcv *bytes* byt num= *seqno*,M= *M*,BE= *BE_bits* lng?= *long* nt *network*

Description: Received an MP packet - this message reports the size (in bytes), the MP sequence number which is found in the MP header, the M value which is the minimum of the last sequence number received for each link in the MP bundle, the BE bits (0=neither, 1=End fragment, 2=Begin fragment, 3=Both Begin and End (full packet)), whether long sequence numbers are being received (1=Yes), and the net and interface that the packet was received on.

MLP.002

Level: CI-ERROR

Short Syntax: MLP.002 net *net* DISCARD (sequence less than expected) nrcv= *nrcv*,num= *seq*

Long Syntax: MLP.002 net *net* DISCARD (sequence less than expected) nrcv= *nrcv*,num= *seq*

Description: Discarding MP packet because the sequence number is less than that which is expected (less than nrcv or M). This may indicate that this packet was already "declared" lost.

MLP.003

Level: P-TRACE

Short Syntax: MLP.003 BAP Snd REQ= *type* (0=Cll,1=Cllbk,2=Drp)

Long Syntax: MLP.003 BAP Snd REQ= *type* (0=Cll,1=Cllbk,2=Drp)

Description: Sending BAP request

MLP.004

Level: CI-ERROR

Short Syntax: MLP.004 net *net* DISC seq= *seq*,nrecv= *nrecv* due to rcv buff shortage

Long Syntax: MLP.004 net *net* DISC seq= *seq*,nrecv= *nrecv* due to rcv buff shortage

Description: Discarding MP packet due to a link receive buffer shortage.

Action: If this occurs frequently, consider increasing link receive buffers.

MLP.005

Level: P-TRACE

Short Syntax: MLP.005 BAP Send CALL-STATUS= *status* (0=SUCC,255=FAIL)

Long Syntax: MLP.005 BAP Send CALL-STATUS= *status* (0=SUCC,255=FAIL)

Description: Sending a BAP Call status indication to indicate whether the link successfully joined the MP bundle (this includes LCP negotiation)

MLP.006

Level: CI-ERROR

Short Syntax: MLP.006 BAP: Inbound Req or Status Ind was not Acked

Long Syntax: MLP.006 BAP: Inbound Req or Status Ind was not Acked

Description: For some reason BAP chose not to ACK an incoming request from the peer. This could happen if there are not enough resources or we are not agreeing with our peers decision to add or drop bandwidth

MLP.007

Level: P-TRACE

Short Syntax: MLP.007 BAP Rcv Req= *theirReq* (0=Cll,1=Cllbk,2=Drp) ->COLLISION favp= *favpeer*

Long Syntax: MLP.007 BAP Rcv Req= *theirReq* (0=Cll,1=Cllbk,2=Drp) ->COLLISION favp= *favpeer*

Description: Recieved an inbound BAP request from our peer but have already sent another request in the meantime. This is a normal collision and will be resolved by the BACP favored peer.

MLP.008

Level: UI-ERROR

Short Syntax: MLP.008 BAP bd state-inbnd= *theirReq*(0=Cll,1=Cbk,2=Drp,3=CR,4=CBR,5=DR,6=S,7=SR)

Long Syntax: MLP.008 BAP bd state-inbnd= *theirReq*(0=Cll,1=Cbk,2=Drp,3=CR,4=CBR,5=DR,6=S,7=SR)

Description: Bad BAP state for inbound BAP packet

MLP.009

Level: P-TRACE

Short Syntax: MLP.009 BAP RCV RSP= *RespType*(0=Cl,1=Cbk,2=Drp,3=St) *Response*(0=AK,1=NK,2=RJ,3=FLN)

Long Syntax: MLP.009 BAP RCV RSP= *RespType*(0=Cl,1=Cbk,2=Drp,3=St) *Response*(0=AK,1=NK,2=RJ,3=FLN)

Description: Received an inbound BAP Response packet with a corresponding response code (ACK means go ahead with request, NAK means I understand and support your request but I don't want you to perform it now - try again later, REJECT means I do not understand/support your request, FULLNAK means I understand and support your request but I am limited by a resource condition of some kind (this could be the Maximum number of links configurable parameter) - do not send this request again until the total bandwidth of the MP bundle changes..

MLP.010

Level: C-INFO

Short Syntax: MLP.010 BOD Aprc= *Add*,ASpd= *AddS*,Dprc= *Drop*,DSpd= *DropS*,oSpd= *out*,iSpd= *in*

Long Syntax: MLP.010 BOD Aprc= *Add*,ASpd= *AddS*,Dprc= *Drop*,DSpd= *DropS*,oSpd= *out*,iSpd= *in*

Description: Checking bandwidth to determine if we need to add or drop a link. Total bandwidth, Add percentage, calculated add speed, drop percentage, calculated drop speed, outbound speed, and inbound speed are displayed. In order to drop both iSpd and oSpd have to drop below DSpd. In order to add either iSpd or oSpd must go above ASpd.

MLP.011

Level: C-INFO

Short Syntax: MLP.011 BAP BOD Drp lnk net= *net*,rem LD= *remLD*,loc LD= *locLD*

Long Syntax: MLP.011 BAP BOD Drp lnk net= *net*,rem LD= *remLD*,loc LD= *locLD*

Description: BAP is causing a link to be dropped - displays remote and local link discriminator.

MLP.012

Level: CI-ERROR

Short Syntax: MLP.012 BAP BOD - Avail *nettype* (0=Drp,1=Out,2=In) nt not found

Long Syntax: MLP.012 BAP BOD - Avail *nettype* (0=Drp,1=Out,2=In) net not found

Description: Normal error when there are not enough resources or a dial-circuit has not yet reset itself from a previous action.

MLP.013

Level: C-INFO

Short Syntax: MLP.013 BAP BOD Adding Net= *net* with *type* (0=Cll,1=Cllbk)

Long Syntax: MLP.013 BAP BOD Adding Net= *net* with *type* (0=Cll,1=Cllbk)

Description: BAP is adding a link

MLP.014

Level: CI-ERROR

Short Syntax: MLP.014 BAP BOD Can NOT check BOD requirements NOW!

Long Syntax: MLP.014 BAP BOD Can NOT check BOD requirements NOW!

Description: Normal error when some BAP process is occuring and the Bandwidth timer pops - bandwidth will not be checked in this interval.

MLP.015

Level: C-INFO

Short Syntax: MLP.015 BAP Peer wants to Drop our LD= *LD*

Long Syntax: MLP.015 BAP Peer wants to Drop our LD= *LD*

Description: Inbound BAP drop request from peer wanting to drop our link with the displayed link discriminator.

MLP.016

Level: P-TRACE

Short Syntax: MLP.016 BAP Rcv *theirReq* (Cll:1,2;CllBck:3,4;Drp:5,6;St:7,8->Req,Rsp)

Long Syntax: MLP.016 BAP Rcv *theirReq* (Cll:1,2;CllBck:3,4;Drp:5,6;St:7,8->Req,Rsp)

Description: received inbound BAP request

MLP.017

Level: C-INFO

Short Syntax: MLP.017 BAP - Our Available Phone number is *phoneNum*

Long Syntax: MLP.017 BAP - Our Available Phone number is *phoneNum*

Description: Found an available phone number to pass to our peer

MLP.018

Level: C-INFO

Short Syntax: MLP.018 BAP cll nt= *net*,olen= *len*,ud= *ud*,sntd= *sent*,offst= *offset*,Ph= *phone*

Long Syntax: MLP.018 BAP cll nt= *net*,olen= *len*,ud= *ud*,sntd= *sent*,offst= *offset*,Ph= *phone*

Description: BAP placing the call. Fields are displayed for which phone number will be used: original length of the phone number, number of unique digits, number of digits that were sent from the peer, the offset into the phone number to start copying, and the phone number.

MLP.019

Level: C-INFO

Short Syntax: MLP.019 MP Nt *net* removed from the bundle

Long Syntax: MLP.019 MP Nt *net* removed from the bundle

Description: MP remove link from bundle

MLP.020

Level: C-INFO

Short Syntax: MLP.020 MP add Nt *net* to the *type* (0=old,1=new) bundle

Long Syntax: MLP.020 MP add Nt *net* to the *type* (0=old,1=new) bundle

Description: MP add link to bundle

MLP.021

Level: P-TRACE

Short Syntax: MLP.021 MP nt *mpnet* Xmt *bytes* bytes num= *num*,shrt?= *shrt* on nt *linknet*

Long Syntax: MLP.021 MP nt *mpnet* Xmt *bytes* bytes num= *num*,shrt?= *shrt* on nt *linknet*

Description: MP XMIT a packet. Display the MP device, packet length MP sequence number, whether we are sending short sequence numbers (1=yes), and the link net.

MLP.022

Level: C-INFO

Short Syntax: MLP.022 BAP OPT: LNK TYPE: spd= *speed*, typ= *type* (1=ISDN,4=ANALOG)

Long Syntax: MLP.022 BAP OPT: LNK TYPE: spd= *speed*, typ= *type* (1=ISDN,4=ANALOG)

Description: BAP option

MLP.023

Level: C-INFO

Short Syntax: MLP.023 BAP OPT: PHONE:dig:unique= *ud*,snt= *ds*;num= *delta*,sub= *subaddr*

Long Syntax: MLP.023 BAP OPT: PHONE:dig:unique= *ud*,snt= *ds*;num= *delta*,sub= *subaddr*

Description: BAP option

MLP.024

Level: C-INFO

Short Syntax: MLP.024 BAP OPT: NO PHONE NUMBER NEEDED

Long Syntax: MLP.024 BAP OPT: NO PHONE NUMBER NEEDED

Description: BAP option

MLP.025

Level: C-INFO

Short Syntax: MLP.025 BAP OPT: REASON: *reason*

Long Syntax: MLP.025 BAP OPT: REASON: *reason*

Description: BAP option

MLP.026

Level: C-INFO

Short Syntax: MLP.026 BAP OPT: LINK DISCRIMINATOR: *ld*

Long Syntax: MLP.026 BAP OPT: LINK DISCRIMINATOR: *ld*

Description: BAP option

MLP.027

Level: C-INFO

Short Syntax: MLP.027 BAP OPT:STAT: *status*(0=SCC,17=BSY,255=FL)act= *action*(0=NO,1=RTRY)

Long Syntax: MLP.027 BAP OPT:STAT: *status*(0=SCC,17=BSY,255=FL)act= *action*(0=NO,1=RTRY)

Description: BAP option

MLP.028

Level: UE-ERROR

Short Syntax: MLP.028 BAP OPTION NOT RECOGNIZED

Long Syntax: MLP.028 BAP OPTION NOT RECOGNIZED

Description: BAP non-option

MLP.029

Level: UI-ERROR

Short Syntax: MLP.029 BAP error (inbound packet): no buffer

Long Syntax: MLP.029 BAP error (inbound packet): no buffer

Description: BAP tried to generate a packet to send a response and couldn't allocate a buffer. This could be a serious low memory problem.

MLP.030

Level: UE-ERROR

Short Syntax: MLP.030 BAP error (inbound packet): length mismatch

Long Syntax: MLP.030 BAP error (inbound packet): length mismatch

Description: BAP error

MLP.031

Level: UE-ERROR

Short Syntax: MLP.031 BAP error (inbound packet): bap_check failed

Long Syntax: MLP.031 BAP error (inbound packet): bap_check failed

Description: bap_check reported an error while processing an inbound BAP packet option.

MLP.032

Level: P-TRACE

Short Syntax: MLP.032 Sending BAP RESPONSE= *resp* (0=Ack)

Long Syntax: MLP.032 Sending BAP RESPONSE= *resp* (0=Ack)

Description: BAP response to an inbound request or status indication. (0=ACK, 1=NAK, 2=REJ, 3=FULLNAK).

MLP.033

Level: UE-ERROR

Short Syntax: MLP.033 BAP error (inbound packet): id mismatch

Long Syntax: MLP.033 BAP error (inbound packet): id mismatch

Description: inbound ID for a response or status indication does not match the one used for the initial request

MLP.034

Level: UE-ERROR

Short Syntax: MLP.034 BAP error: unique digits > digits sent

Long Syntax: MLP.034 BAP error: unique digits > digits sent

Description: cannot form a phone number to dial - unique digits is greater than the number of digits that were sent.

MLP.035

Level: P-TRACE

Short Syntax: MLP.035 mk favorite peer magic number *magic*

Long Syntax: MLP.035 making favorite peer magic number with value *magic*

Description: bacp_option built favpeer.

MLP.036

Level: UI-ERROR

Short Syntax: MLP.036 mk bacp unk *option*

Long Syntax: MLP.036 making unknown bacp option *option*

Description: bacp_option built an unrecognized option.

MLP.037

Level: C-INFO

Short Syntax: MLP.037 *state, routine_name,* nt *network ID*

Long Syntax: MLP.037 state = *state,*, called *routine_name,*, on nt *network ID*

Description: Called the specified cp routine.

MLP.038

Level: P-TRACE

Short Syntax: MLP.038 ck favorite peer mag 0x *magic_number*

Long Syntax: MLP.038 checking favorite peer magic number with value 0x *magic_number*

Description: bacp_check processed magic number.

MLP.039

Level: P-TRACE

Short Syntax: MLP.039 ck bacp unk *option*

Long Syntax: MLP.039 checking unknown bacp option *option*

Description: bacp_check processed an unrecognized option.

MLP.040

Level: UE-ERROR

Short Syntax: MLP.040 Bd bacp req hdr lngth, nt *network ID*

Long Syntax: MLP.040 Bad BACP request header length, on network *network ID*

Description: bacp_req got request with bad header length.

MLP.041

Level: UE-ERROR

Short Syntax: MLP.041 Bd bacp req opt *bacp_option,* shrt, nt *network ID*

Long Syntax: MLP.041 Bd BACP req opt *bacp_option,,* data too short, on net *network ID*

Description: bacp_req got request containing option with insufficient data.

MLP.042

Level: C-TRACE

Short Syntax: MLP.042 bacp req rslt: *bacp_rslt,,* opt *bacp_option,,* ln *opt_len,,* nt *network ID*

Long Syntax: MLP.042 bacp req rslt: *bacp_rslt,,* opt *bacp_option,,* ln *opt_len,,* nt *network ID*

Description: Result, so far, of processing one option.

MLP.043

Level: UE-ERROR

Short Syntax: MLP.043 lpbk nt *network ID*

Long Syntax: MLP.043 Excessive bacp magic number collisions on nt *network ID*

Description: Excessive magic number collisions while trying to configure link

MLP.044

Level: UE-ERROR

Short Syntax: MLP.044 Bd bacp ack id, exp *exp_id,* gt *got_id,,* nt *network ID*

Long Syntax: MLP.044 Bad bacp ack id, exp *exp_id,* got *got_id,,* on nt *network ID*

Description: bacp_ack got config ack with bad id.

MLP.045

Level: UE-ERROR

Short Syntax: MLP.045 Bd bacp ack lngth, nt *network ID*

Long Syntax: MLP.045 Bad bacp ack length, on network *network ID*

Description: bacp_ack got config ack with bad length.

MLP.046

Level: UE-ERROR

Short Syntax: MLP.046 msmtchd bacp ack, nt *network ID*

Long Syntax: MLP.046 mis-matched data in bacp ack, on network *network ID*

Description: bacp_ack got ack whose data doesn't match our request.

MLP.047

Level: UE-ERROR

Short Syntax: MLP.047 Bd bacp nak id, exp *exp_id,* gt *got_id,,* nt *network ID*

Long Syntax: MLP.047 Bad BACP nak id, expected *exp_id,,* got *got_id,,* on network *network ID*

Description: bacp_nak got nak with bad id.

MLP.048

Level: UE-ERROR

Short Syntax: MLP.048 Bd bacp nak lngth, nt *network ID*

Long Syntax: MLP.048 Bad BACP nak length, on network *network ID*

Description: bacp_nak got nak with bad length.

MLP.049

Level: UE-ERROR

Short Syntax: MLP.049 Bd bacp nak opt *bacp_option*,, nt *network ID*

Long Syntax: MLP.049 Bad BACP nak option = *bacp_option*,, on network *network ID*

Description: bacp_nak got nak containing out-of-range option.

MLP.050

Level: UE-ERROR

Short Syntax: MLP.050 out-ordr bacp nak opt *bacp_option*,, nt *network ID*

Long Syntax: MLP.050 Bad BACP nak option = *bacp_option*,, on network *network ID*

Description: bacp_nak got nak containing out-of-order option.

MLP.051

Level: UE-ERROR

Short Syntax: MLP.051 Bd bacp nak opt *bacp_option*, shrt, nt *network ID*

Long Syntax: MLP.051 Bad BACP nak option = *bacp_option*,, data too short, on network *network ID*

Description: bacp_nak got nak containing option with insufficient data.

MLP.052

Level: UE-ERROR

Short Syntax: MLP.052 Bd bacp rej id, exp *exp_id*, gt *got_id*,, nt *network ID*

Long Syntax: MLP.052 Bad bacp rej id, expected *exp_id*,, got *got_id*,, on network *network ID*

Description: bacp_ack got config ack with bad id.

MLP.053

Level: UE-ERROR

Short Syntax: MLP.053 Bd bacp rej lngth, nt *network ID*

Long Syntax: MLP.053 Bad BACP reject length, on network *network ID*

Description: bacp_rej got reject with bad length.

MLP.054

Level: UE-ERROR

Short Syntax: MLP.054 Bd bacp rej opt *bacp_option*,, nt *network ID*

Long Syntax: MLP.054 Bad BACP reject option = *bacp_option*,, on network *network ID*

Description: bacp_rej got reject containing out-of-range option.

MLP.055

Level: UE-ERROR

Short Syntax: MLP.055 out-ordr bacp rej opt *bacp_option*,, nt *network ID*

Long Syntax: MLP.055 Bad BACP reject option = *bacp_option*,, on network *network ID*

Description: bacp_rej got reject containing out-of-order option.

MLP.056

Level: C-INFO

Short Syntax: MLP.056 MP bundle removed (Nt *net*)

Long Syntax: MLP.056 MP bundle removed (Nt *net*)

Description: MP unbundle - all remaining links (if there are any) will be brought down.

MLP.057

Level: P-TRACE

Short Syntax: MLP.057 MP Nt *oldnet* SLW XMT shrt?= *short* frg *frag* (*bytes* byt) on nt *net*

Long Syntax: MLP.057 MP Nt *oldnet* SLW XMT shrt?= *short* frg *frag* (*bytes* byt) on nt *net*

Description: MP Slow XMIT. This is used for the BRS n_get transmit path. The parameters are the same as in MLP_21

MLP.058

Level: UI-ERROR

Short Syntax: MLP.058 Bad Dial nt *network ID*

Long Syntax: MLP.058 Bad Dialout MP link for nt *network ID*

Description: The Dialout MP link configured is not present or not a Dialout MP only link.

Cause: Configuration error.

Action: Configure dial circuit as a Dialout MP only link.

MLP.059

Level: UE-ERROR

Short Syntax: MLP.059 ERROR: Fixed link *link* with wrong MP nt *MPnet* - dropping link

Long Syntax: MLP.059 ERROR: Fixed link *link* with wrong MP nt *MPnet* - dropping link

Description: link is MP ONLY and found ia bundle which was not the configured bundle

Cause: Configuration error.

MLP.060

Level: UE-ERROR

Short Syntax: MLP.060 ERROR: MaxLinks exceeded MP nt *MPnet* - dropping nt *link*

Long Syntax: MLP.060 ERROR: MaxLinks exceeded MP nt *MPnet* - dropping nt *link*

Description: Max number of links for this bundle was exceeded

Cause: Configuration error.

MLP.061

Level: UE-ERROR

Short Syntax: MLP.061 ERROR: LCP or Auth on nt *link* mismatch w/ MP nt *MPnet* - dropping lnk

Long Syntax: MLP.061 ERROR: LCP or Auth on nt *link* mismatch w/ MP nt *MPnet* - dropping lnk

Description: LCP or Authentication negotiations do not match for this link

Cause: Configuration error.

MLP.062

Level: UI-ERROR

Short Syntax: MLP.062 ERROR: link nt *link* not the 1st link in bundle MP nt *MPnet* - dropping lnk

Long Syntax: MLP.062 ERROR: link nt *link* not the 1st link in bundle MP nt *MPnet* - dropping lnk

Description: Not the first link for this MP net

Cause: Configuration error.

MLP.063

Level: UI-ERROR

Short Syntax: MLP.063 ERROR: No MP bundle Net to use - dropping nt *link*

Long Syntax: MLP.063 ERROR: No MP bundle Net to use - dropping nt *link*

Description: No MP net to use for this MP session

Cause: Configuration error.

MLP.064

Level: UI-ERROR

Short Syntax: MLP.064 ERROR: Out of MP Buffers on MP nt *MPnet*

Long Syntax: MLP.064 ERROR: Out of MP Buffers on MP nt *MPnet*

Description: No More MP buffers on this MP net

Cause: Need more memory

MLP.065

Level: UI-ERROR

Short Syntax: MLP.065 ERROR: mp_netostart is performed on a non-MP net *MPnet*

Long Syntax: MLP.065 ERROR: mp_netostart is performed on a non-MP net *MPnet*

Description: mp_netostart (BRS) is performed on a non-MP net

Cause: Configuration error.

MLP.066

Level: C-INFO

Short Syntax: MLP.066 >>>>->> mp_init_prvq nt *MPnet* w/ *numbuffs* bufs, *numbytes* bytes!

Long Syntax: MLP.066 >>>>->> mp_init_prvq nt *MPnet* w/ *numbuffs* bufs, *numbytes* bytes!

Description: MP intialization

MLP.067

Level: C-INFO

Short Syntax: MLP.067 mp_slftst : net = *MPnet*

Long Syntax: MLP.067 mp_slftst : net = *MPnet*

Description: MP self test

MLP.068

Level: C-INFO

Short Syntax: MLP.068 >>>>->> performing n_up for DOD nt *MPnet*

Long Syntax: MLP.068 >>>>->> performing n_up for DOD nt *MPnet*

Description: Normal MP self test for Dial-On-Demand MP net.

MLP.069

Level: C-INFO

Short Syntax: MLP.069 >>>>->> performing n_down for FIXED INBOUND nt *MPnet*

Long Syntax: MLP.069 >>>>->> performing n_down for FIXED INBOUND nt *MPnet*

Description: Normal MP self test for fixed inbound MP circuit.

MLP.070

Level: UI-ERROR

Short Syntax: MLP.070 MP self test nt *MPnet* - bad state

Long Syntax: MLP.070 MP self test nt *MPnet* - bad state

Description: Bad MP self test

MLP.071

Level: C-INFO

Short Syntax: MLP.071 >>>>->> performing n_down since NETDOWN nt *MPnet*

Long Syntax: MLP.071 >>>>->> performing n_down since NETDOWN nt *MPnet*

Description: MP self test - Base net is still in NETDOWN state.

MLP.072

Level: C-INFO

Short Syntax: MLP.072 >>>>->> performing n_up for DOD nt *MPnet*

Long Syntax: MLP.072 >>>>->> performing n_up for DOD nt *MPnet*

Description: Normal MP self test for Dial-On-Demand MP net.

MLP.073

Level: C-INFO

Short Syntax: MLP.073 >>>>->> performing n_down for FIXED INBOUND nt *MPnet*

Long Syntax: MLP.073 >>>>->> performing n_down for FIXED INBOUND nt *MPnet*

Description: Normal MP self test for fixed inbound MP circuit.

MLP.074

Level: C-INFO

Short Syntax: MLP.074 >>>>->> performing n_down since callout failure nt *MPnet*

Long Syntax: MLP.074 >>>>->> performing n_down since callout failure nt *MPnet*

Description: MP self test - dial-circuit callour failed will try again

MLP.075

Level: C-INFO

Short Syntax: MLP.075 >>>>->> performing n_down bad link state and not calling nt *MPnet*

Long Syntax: MLP.075 >>>>->> performing n_down bad link state and not calling nt *MPnet*

Description: dial-circuit net seems to be in wierd state

MLP.076

Level: C-TRACE

Short Syntax: MLP.076 Idle timer expired MP nt *MPnet* - MP circuit down

Long Syntax: MLP.076 Idle timer expired MP nt *MPnet* - MP circuit down

Description: MP maintenance - idle timer expired for Dial-On-Demand MP circuit.

MLP.077

Level: CI-ERROR

Short Syntax: MLP.077 BAP: NO Available Phone Number

Long Syntax: MLP.077 BAP: NO Available Phone Number

Description: BAP Can't pass a phone number - there are none to get. Either a resource condition or one (or more dial-circuits) have not yet reset themselves from a previous action.

MLP.078

Level: UE-ERROR

Short Syntax: MLP.078 BAP: unknown PHONE DELTA sub option

Long Syntax: MLP.078 BAP: unknown PHONE DELTA sub option

Description: unknown PHONE DELTA sub option

MLP.079

Level: UE-ERROR

Short Syntax: MLP.079 BAP: unknown BAP option *option*

Long Syntax: MLP.079 BAP: unknown BAP option *option*

Description: unknown BAP option

MLP.080

Level: UE-ERROR

Short Syntax: MLP.080 BAP: FAILED BAP NEGOTIATIONS nt *MPnet*

Long Syntax: MLP.080 BAP: FAILED BAP NEGOTIATIONS nt *MPnet*

Description: Failed BAP negotiations - LINK TYPE OR PHONE DELTA NOT SUPPLIED

MLP.081

Level: C-TRACE

Short Syntax: MLP.081 BAP: Place call..tmp_addr= *temp*, dst_addr= *dst*

Long Syntax: MLP.081 BAP: Place call..tmp_addr= *temp*, dst_addr= *dst*

Description: call placed - tmp_addrs will be used to place the call.

MLP.082

Level: C-TRACE

Short Syntax: MLP.082 BACP OPEN nt *MPnet*: fav= *favpeer* (0=N,1=Y) loc= *local*,rem= *remote*

Long Syntax: MLP.082 BACP OPEN nt *MPnet*: fav= *favpeer* (0=N,1=Y) loc= *local*,rem= *remote*

Description: BACP opened for MP net - displayed are the favored peer values.

MLP.083

Level: UE-ERROR

Short Syntax: MLP.083 ERROR: Unknown BAP pkt type *type*

Long Syntax: MLP.083 ERROR: Unknown BAP pkt type *type*

Description: Received unknown BAP packet type

MLP.084

Level: UI-ERROR

Short Syntax: MLP.084 BAP: Unable to drop MP lnk from MP nt *MPnet*

Long Syntax: MLP.084 BAP: Unable to drop MP lnk from MP nt *MPnet*

Description: Link drop timeout period elapsed before link could be brought down

MLP.085

Level: UE-ERROR

Short Syntax: MLP.085 ERROR: No MP on static MP Link nt *linkNet*

Long Syntax: MLP.085 ERROR: No MP on static MP Link nt *linkNet*

Description: MRRU was not negotiated successfully on a MP ONLY link

MLP.086

Level: UE-ERROR

Short Syntax: MLP.086 ERROR: bad Endpt Disc on static MP Lnk nt *linkNet*

Long Syntax: MLP.086 ERROR: bad Endpt Disc on static MP Lnk nt *linkNet*

Description: Endpoint Discriminator on an MP ONLY link was different than the bundle

MLP.087

Level: UE-ERROR

Short Syntax: MLP.087 ERROR: nt *linkNet* parameter mismatch with MP bundle

Long Syntax: MLP.087 ERROR: nt *linkNet* parameter mismatch with MP bundle

Description: link d not negotiate same parameters as the MP bundle

MLP.088

Level: UE-ERROR

Short Syntax: MLP.088 ERROR: nt *linkNet* did not neg lnk disc on MP nt *MPnet* -BAP

Long Syntax: MLP.088 ERROR: nt *linkNet* did not neg lnk disc on MP nt *MPnet* -BAP

Description: link net did not negotiate link discriminator for bundle running BAP

MLP.089

Level: UE-ERROR

Short Syntax: MLP.089 ERROR: nt *MPnet* rcv BAP packet in bapCLOSED state

Long Syntax: MLP.089 ERROR: nt *MPnet* rcv BAP packet in bapCLOSED state

Description: BAP packet was received on a circuit not running BAP

MLP.090

Level: UE-ERROR

Short Syntax: MLP.090 ERROR: nt *MPnet* rcvd a BACP pkt in CLOSED state

Long Syntax: MLP.090 ERROR: nt *MPnet* rcvd a BACP pkt in CLOSED state

Description: BACP packet received in closed state

MLP.091

Level: UI-ERROR

Short Syntax: MLP.091 ERROR: BRS enabled on an MP slave net *linkNet*

Long Syntax: MLP.091 ERROR: BRS enabled on an MP slave net *linkNet*

Description: BRS enabled on an MP link net - packet will be dropped

---

MLP.092

Level: C-INFO

Short Syntax: MLP.092 BAP: no subaddress found

Long Syntax: MLP.092 BAP: no subaddress found

Description: BAP: no subaddress found - not going to pass one. This is probably because multiport is not supported or is not currently being used.

MLP.093

Level: C-INFO

Short Syntax: MLP.093 BAP - Our Available subaddress is *subAddr*

Long Syntax: MLP.093 BAP - Our Available subaddress is *subAddr*

Description: Found a subaddress and will pass it in our BAP call-response or BAP callback-request.

MLP.094

Level: UI-ERROR

Short Syntax: MLP.094 WARNING: nt *link* already added to MP nt *MPnet* - continuing

Long Syntax: MLP.094 WARNING: nt *link* already added to MP nt *MPnet* - continuing

Description: pppblk already was added to MP net - we will proceed

Cause: Probably dropped packet.

MLP.095

Level: UI-ERROR

Short Syntax: MLP.095 Leased MP net with WRS/DOD nt *network ID*

Long Syntax: MLP.095 Leased MP net with WRS/DOD nt *network ID*

Description: The MP net contains leased link(s) and has WRS.

Cause: Configuration error.

Action: disable WRS.

MLP.096

Level: UI-ERROR

Short Syntax: MLP.096 Disabling BAP on Leased MP nt *network ID*

Long Syntax: MLP.096 Disabling BAP on Leased MP nt *network ID*

Description: The MP net is ″leased″ (contains at least one leased circuit) and is enabled for BAP. This is an illegal configuration. BAP will be disabled and BOD will not use BAP.

Cause: Configuration error.

Action: disable BAP on leased MP circuit.

MLP.097

Level: UI-ERROR

Short Syntax: MLP.097 Disabling BAP on MP nt *network ID* w/ bad ctrl net

Long Syntax: MLP.097 Disabling BAP on MP nt *network ID* w/ bad ctrl net

Description: BAP only works when the Ctrl net is a valid ISDN dial-circuit which is not I.430, I.431.

Cause: Configuration error.

Action: disable BAP on MP circuit.

MLP.098

Level: UI-ERROR

Short Syntax: MLP.098 Disabling BOD ON INBOUND MP nt *network ID*

Long Syntax: MLP.098 Disabling BOD ON INBOUND MP nt *network ID*

Description: BOD cannot work on an INBOUND MP circuit (unless BAP is used).

Cause: Disable BOD.

Action: disable BOD on MP circuit.

MLP.099

Level: C-INFO

Short Syntax: MLP.099 BOD: Dropping nt *linknet* from MP nt *mpnet*

Long Syntax: MLP.099 BOD: Dropping nt *linknet* from MP nt *mpnet*

Description: BOD dropped a link

MLP.100

Level: C-INFO

Short Syntax: MLP.100 BOD: Attempting to Add nt *linknet* to MP nt *mpnet*

Long Syntax: MLP.100 BOD: Attempting to Add nt *linknet* to MP nt *mpnet*

Description: BOD adding a link

MLP.101

Level: UI-ERROR

Short Syntax: MLP.101 WARNING: slftst Nt *MPnet* - bundle sequences are off

Long Syntax: MLP.101 WARNING: slftst Nt *MPnet* - bundle sequences are off

Description: Bundle sequence numbers are off. This is caused by 30 consecutive drops due to receiving a packet which is less than expected. This is a recovery mechanism.

Cause: This can be caused by one link in a bundle restarting faster than the other link(s) can come down. This results in one bundle never reestablishing itself and using sequence numbers from the old connection.

MLP.102

Level: UI_ERROR

Short Syntax: MLP.102 ERROR: n_speed is 0 or very small,MP net *MPnet* - dropping net *link*

Long Syntax: MLP.102 ERROR: n_speed is 0 or very small,MP net *MPnet* - dropping net *link*

Description: n_speed was not configured on one of the links attempting to join the MP bundle.

MLP.103

Level: CI-ERROR

Short Syntax: MLP.103 net *net* DISC num= *seqno*,nrcv= *nrcv*,M= *M* due timeout

Long Syntax: MLP.103 net *net* DISC num= *seqno*,nrcv= *nrcv*,M= *M* due timeout

Description: Discarding MP packet because the MP timeout period expired while waiting for a packet/fragment

---

MLP.104

Level: CE-ERROR

Short Syntax: MLP.104 net *net* DISC (packet dropped in transit) M= *M*,num= *seq*,nr= *nr*

Long Syntax: MLP.104 net *net* DISC (packet dropped in transit) M= *M*,num= *seq*,nr= *nr*

Description: Discarding MP packet because a packet was dropped. This was determined by calculating M, the minimum of all links' last received sequence number. M has surpassed these dropped packets, indicating that they will never be received.

MLP.105

Level: CE-ERROR

Short Syntax: MLP.105 packets dropped in transit net *net*, new M= *M*,nrecv= *nrecv*

Long Syntax: MLP.105 packets dropped in transit net *net*, new M= *M*,nrecv= *nrecv*

Description: At least one packet was dropped in transit or discarded because M surpassed fragments/packets that will never be received (MLP_104 logged). This ELS message displays the new nrecv (next receive sequence number).

Chapter 80. Multiprotocol Over ATM Client (MPOA)

This chapter describes Multiprotocol Over ATM Client (MPOA) messages. For information on message content and how to use the message, refer to the Introduction.

MPOA.001

Level: UI-ERROR

Short Syntax: MPOA.001 Critical memory allocation failure! (interface # *net_number*)

Long Syntax: MPOA.001 Critical memory allocation failure! (interface # *net_number*)

Description: A critical memory allocation failure has occured on the mentioned interface number, which will impede normal MPC operation.

Cause: The box is running low on dynamically allocatable memory.

MPOA.002

Level: CI-ERROR

Short Syntax: MPOA.002 Non-critical memory allocation failure! (interface # *net_number*)

Long Syntax: MPOA.002 Non-critical memory allocation failure! (interface # *net_number*)

Description: A non-critical memory allocation failure has occured the mentioned interface number. This will not impede MPC operation but may reduce the performance somewhat.

Cause: The box is running low on dynamically allocatable memory.

MPOA.003

Level: UE-ERROR

Short Syntax: MPOA.003 Unrecog frame recvd (dropped)! (0x *word_1* 0x *word_2* 0x *word_3*)

Long Syntax: MPOA.003 Unrecognized frame received (dropped)! (0x *word_1* 0x *word_2* 0x *word_3*)

Description: A frame was received with an unrecognized/ unsupported LLC SNAP encoding (the first 3 words of the frame are printed). The frame was hence discarded.

MPOA.004

Level: UE-ERROR

Short Syntax: MPOA.004 Bad NHRP/MPOA msg recvd (dropped)! (0x *word_1* 0x *word_2* 0x *word_3*)

Long Syntax: MPOA.004 Bad NHRP/MPOA message received (dropped)! (0x *word_1* 0x *word_2* 0x *word_3*)

Description: A corrupted NHRP/MPOA message was received (based on checks like header checksum) (the first 3 words of the frame are printed). The frame was hence discarded.

MPOA.005

Level: UE-ERROR

Short Syntax: MPOA.005 NHRP Error ind. msg recvd! (intf # *intf_num*, code *error_code*)

Long Syntax: MPOA.005 NHRP Error indication message received! (intf # *intf_num* code *error_code*)

Description: A NHRP error indication message was received

MPOA.006

Level: UI-ERROR

Short Syntax: MPOA.006 Internal ATM downcall failed (*function_name*, return: *error_string*)

Long Syntax: MPOA.006 Internal ATM downcall failed (*function_name*, return: *error_string*)

Description: An internal downcall to the ATM driver failed which normally shouldn't.

MPOA.007

Level: U-INFO

Short Syntax: MPOA.007 Couldn't send MPOA Ctrl msg to *atm_address*, rc *return_code*

Long Syntax: MPOA.007 Couldn't send MPOA Ctrl msg to *atm_address*, rc *return_code*

Description: The MPOA Client could not send an MPOA control frame to the specified ATM address. This could be because there is no usable VCC to that address.

MPOA.008

Level: C-INFO

Short Syntax: MPOA.008 Sending MPOA NAK *atm_address*, cie code *cie_code*

Long Syntax: MPOA.008 Sending MPOA NAK *atm_address*, cie code *cie_code*

Description: The MPOA Client is initiating the sending of a NAK message to an MPOA device with the specified ATM address. The specified CIE code is being used.

MPOA.009

Level: C-INFO

Short Syntax: MPOA.009 Recvd Imposition Rqst on (*vpi/ vci*)

Long Syntax: MPOA.009 Recvd Imposition Rqst on (*vpi/ vci*)

Description: The MPC has received an MPOA cache imposition request message on the specified VCC.

MPOA.010

Level: UE-ERROR

Short Syntax: MPOA.010 Parsing error on recvd msg (*error_string*/rc *return_code*)

Long Syntax: MPOA.010 Parsing error on recvd msg (*error_string*/rc *return_code*)

Description: There was an error when parsing a received MPOA control frame.

MPOA.011

Level: UE-ERROR

Short Syntax: MPOA.011 Imposition rqst failure (*error_string*/rc *return_code*)

Long Syntax: MPOA.011 Imposition rqst failure (*error_string*/rc *return_code*)

Description: A failure occured when processing a received MPOA Cache Imposition request message. The error string gives some more information.

MPOA.012

Level: UE-ERROR

Short Syntax: MPOA.012 Recvd Imposition rqst from unknown MPS (VCC *vpi/ vci*)

Long Syntax: MPOA.012 Recvd Imposition rqst from unknown MPS (VCC *vpi/ vci*)

Description: The MPC has received an imposition request on the specified VCC but has not currently learnt that the sender is an MPS.

MPOA.013

Level: C-INFO

Short Syntax: MPOA.013 Free MPOA tag given out (*tag*)

Long Syntax: MPOA.013 Free MPOA tag given out (*tag*)

Description: A free MPOA tag has been assigned to an egress cache entry.

MPOA.014

Level: C-INFO

Short Syntax: MPOA.014 MPOA tag value freed (*tag*)

Long Syntax: MPOA.014 Previously assigned MPOA tag value freed back (*tag*)

Description: A previously assigned MPOA tag value has been freed back.

MPOA.015

Level: C-INFO

Short Syntax: MPOA.015 New egress cache entry created (*ipAddress/ ipAddressMask*, x *cacheID*, *entryTypeString*)

Long Syntax: MPOA.015 New egress cache entry created (*ipAddress/ ipAddressMask*, x *cacheID entryTypeString*)

Description: A new egress cache entry has been created for the specified protocol address and mask pair. The cacheID and the type of the entry are also printed.

MPOA.016

Level: U-INFO

Short Syntax: MPOA.016 Cntrl frm too large to send! (type x *MPOAFrameType*, max *maxFrameSize*)

Long Syntax: MPOA.016 Control frame too large to send! (type x *MPOAFrameType*, max *maxFrameSize*)

Description: The MPOA Client prepared a control frame which was longer than the maximum sized control frame that can be sent. Hence, unable to send this frame. The MPOA packet type code is printed alongwith the maximum allowable frame size.

MPOA.017

Level: U-INFO

Short Syntax: MPOA.017 Imposition rqst unsuccessful (*errString*, x *errCode*)

Long Syntax: MPOA.017 Imposition rqst unsuccessful (*errString*, x *errCode*)

Description: A received cache imposition request was unsuccessful in creating an active usable egress cache entry. An error string and internal error code describing the problem are printed.

MPOA.018

Level: C-INFO

Short Syntax: MPOA.018 Imposn rfrsh for exstng ntry recvd (*ipAddress*/ *ipAddressMask*, x *cacheID*, *entryTypeString*)

Long Syntax: MPOA.018 Imposition refresh for existing entry received (*ipAddress*/ *ipAddressMask*, x *cacheID*, *entryTypeString*)

Description: An MPOC Cache imposition request was received to refresh an existing entry in the egress cache. The destination protocol address, mask, cache ID and type of entry are displayed.

MPOA.019

Level: U-INFO

Short Syntax: MPOA.019 Recvd imposn rqst had dll mismatch!(*ipAddress*/ *ipAddressMask*)

Long Syntax: MPOA.019 Recvd imposn rqst had dll mismatch!(*ipAddress*/ *ipAddressMask*)

Description: An imposition request received for an existing e-cache entry had a different DLL than the one provided on the previous request. The MPC will internally delete the old entry and create a new one with the new information.

MPOA.020

Level: C-INFO

Short Syntax: MPOA.020 Imposn rqst valid (*ipAddress*/ *ipAddressMask*, rqst ID: x *requestID*)

Long Syntax: MPOA.020 Imposition request valid (*ipAddress*/ *ipAddressMask*, rqst ID: x *requestID*)

Description: A valid cache imposition request has been received for the specified destination address/mask pair. The request ID in the message is also printed.

MPOA.021

Level: C-INFO

Short Syntax: MPOA.021 Processing imposn rqst for new ntry (*ipAddress*/ *ipAddressMask*, x *cacheID*, *entryTypeString*)

Long Syntax: MPOA.021 Processing imposition request for a new entry (*ipAddress*/ *ipAddressMask*, x *cacheID*, *entryTypeString*)

Description: An MPOA Cache imposition request for a new egress cache entry is being processed. The destination protocol address, mask, cache ID and type of entry are displayed.

MPOA.022

Level: U-INFO

Short Syntax: MPOA.022 Recvd imposn rqst, ntry type mismatch (old *ipAddress*/ new *ipAddressMask*)

Long Syntax: MPOA.022 Recvd imposn rqst, entry type mismatch (old *ipAddress*/ new *ipAddressMask*)

Description: An imposition request received for an existing e-cache entry had a different entry type than the existing one. The existing entry will internally be deleted and a new one created.

MPOA.023

Level: UE-ERROR

Short Syntax: MPOA.023 Recvd imposn rqst, prtcl addr mismatch (old *oldIpAddressPrefix*/ new *newIpAddressPrefix*)

Long Syntax: MPOA.023 Recvd imposn rqst, prtcl addr mismatch (old *oldIpAddressPrefix*/ new *newIpAddressPrefix*)

Description: An imposition request received for an existing e-cache entry had a different destination protocol address prefix than the existing one. The existing entry will internally be deleted and a new one created. Indicates a likely error in the E-MPS logic.

MPOA.024

Level: UE-ERROR

Short Syntax: MPOA.024 Recvd imposn rqst had src atm addr mismatch (*ipAddress*/ *ipAddressMask*)

Long Syntax: MPOA.024 Recvd imposn rqst had src atm addr mismatch (*ipAddress*/ *ipAddressMask*)

Description: An imposition request received for an existing e-cache entry had a different source ATM address than the existing one. The existing entry will internally be deleted and a new one created. Indicates a likely error in the E-MPS logic.

MPOA.025

Level: UI-ERROR

Short Syntax: MPOA.025 Internal error (function: *functionName*)

Long Syntax: MPOA.025 Internal error (function: *functionName*)

Description: An internal error occured (in the indicated function). This is purely for debugging purposes.

MPOA.026

Level: UE-ERROR

Short Syntax: MPOA.026 Recvd imposn rfrsh for purging ntry (*ipAddress*/ *ipAddressMask*)

Long Syntax: MPOA.026 Recvd imposn rfrsh for purging ntry (*ipAddress*/ *ipAddressMask*)

Description: An imposition request refresh was received for an entry which was in the process of being purged. Indicates a likely error in the E-MPS logic.

MPOA.027

Level: C-INFO

Short Syntax: MPOA.027 Egress cache entry refreshed (*ipAddress*/ *ipAddressMask*, id x *cacheID*, *entryTypeString*)

Long Syntax: MPOA.027 Egress cache entry refreshed (*ipAddress*/ *ipAddressMask*, id x *cacheID*, *entryTypeString*)

Description: An existing egress cache entry has been refreshed for the specified protocol address and mask pair. The cacheID and the type of the entry are also printed.

MPOA.028

Level: C-INFO

Short Syntax: MPOA.028 Cache id based e-mps purge recvd (x *cacheID*)

Long Syntax: MPOA.028 Cache id based e-mps purge recvd (x *cacheID*)

Description: An e-mps purge was received for a single egress cache entry. The cache ID of this entry is printed.

MPOA.029

Level: C-INFO

Short Syntax: MPOA.029 Prtcl addr based e-mps purge recvd (*ipAddress*/ *ipAddressMask*)

Long Syntax: MPOA.029 Prtcl addr based e-mps purge recvd (*ipAddress*/ *ipAddressMask*)

Description: An e-mps purge was received for a range of protocol addresses. This range is specified by the protocol address and address mask (the range could also indicate only one address).

MPOA.030

Level: U-INFO

Short Syntax: MPOA.030 Cache id based e-mps purge. No match! (x *cacheID*)

Long Syntax: MPOA.030 Cache id based e-mps purge. No match! (x *cacheID*)

Description: No matching cache entry was found for an e-mps cache purge request containing a cache ID.

MPOA.031

Level: C-INFO

Short Syntax: MPOA.031 Emps purge: purging egress cache entry (id x *cacheID*)

Long Syntax: MPOA.031 Emps purge: purging egress cache entry (id x *cacheID*)

Description: An egress cache entry is being purged because of a received e-mps purge request. The cacheID of the entry is displayed.

MPOA.032

Level: C-INFO

Short Syntax: MPOA.032 Aging out egress cache entry (id x *cacheID*)

Long Syntax: MPOA.032 Aging out egress cache entry (id x *cacheID*)

Description: An egress cache entry is being aged out because its holding time has expired. The cacheID of the entry is displayed.

MPOA.033

Level: U-INFO

Short Syntax: MPOA.033 Initiating e-mpc purge request (id x *cacheID*)

Long Syntax: MPOA.033 Initiating e-mpc purge request (id x *cacheID*)

Description: An e-mpc initiated purge request procedure is being started for the specified egress cache entry.

MPOA.034

Level: U-INFO

Short Syntax: MPOA.034 Error preparing e-mpc purge request (id x *cacheID*, err code *errorCode*)

Long Syntax: MPOA.034 Error preparing e-mpc purge request (id x *cacheID*, err code *errorCode*)

Description: An e-mpc initiated purge request message could not be prepared because of a local error (e.g. memory allocation failure).

MPOA.035

Level: U-INFO

Short Syntax: MPOA.035 Retrying e-mpc purge request (id x *cacheID*)

Long Syntax: MPOA.035 Retrying e-mpc purge request (id x *cacheID*)

Description: An e-mpc initiated purge request procedure is being retried for the specified egress cache entry. This is because either the earlier request could not be sent out or no valid reply was received for it.

MPOA.036

Level: U-INFO

Short Syntax: MPOA.036 Exhausted e-mpc purge request retries. entry deleted (id x *cacheID*)

Long Syntax: MPOA.036 Exhausted e-mpc purge request retries. entry deleted (id x *cacheID*)

Description: Multiple attempts at an e-mpc initiated purge of an egress cache entry failed. The maximum retry time has been exceeded and the entry is being locally deleted.

MPOA.037

Level: C-INFO

Short Syntax: MPOA.037 Received empc purge reply (vpi/vci *vpi/ vci*)

Long Syntax: MPOA.037 Received empc purge reply (vpi/vci *vpi/ vci*)

Description: An empc purge reply was received by this MPC on the specified VCC.

MPOA.038

Level: C-INFO

Short Syntax: MPOA.038 eMPC purge reply without dll hdr (vpi/vci *vpi/ vci*)

Long Syntax: MPOA.038 eMPC purge reply without dll hdr (vpi/vci *vpi/ vci*)

Description: An empc purge reply was received by this MPC on the specified VCC without an MPOA DLL Header extension. The packet will be dropped without further processing.

MPOA.039

Level: UE-ERROR

Short Syntax: MPOA.039 eMPC purge reply: no match on cache ID (id x *cacheId*)

Long Syntax: MPOA.039 eMPC purge reply: no match on cache ID (id x *cacheId*)

Description: An empc purge reply was received by this MPC but the cache ID in the message (printed) does not match any existing entry imposed by the same MPS. The packet will be dropped without further processing.

MPOA.040

Level: UE-ERROR

Short Syntax: MPOA.040 eMPC purge reply: mps mismatch (id x *cacheId*)

Long Syntax: MPOA.040 eMPC purge reply: mps mismatch (id x *cacheId*)

Description: An empc purge reply was received by this MPC but the matching cache entry was imposed by a different MPS. The packet will be dropped without further processing. The cacheId in the received message is printed.

MPOA.041

Level: UE-ERROR

Short Syntax: MPOA.041 eMPC purge reply recvd for non purging entry (id x *cacheId*)

Long Syntax: MPOA.041 eMPC purge reply recvd for non purging entry (id x *cacheId*)

Description: An empc purge reply was received by this MPC but the matching cache entry is not being purged at this time. The packet will be dropped without further processing. The cacheId in the received message is printed.

MPOA.042

Level: C-INFO

Short Syntax: MPOA.042 eMPC purge reply deleted entry (id x *cacheId*)

Long Syntax: MPOA.042 eMPC purge reply deleted entry (id x *cacheId*)

Description: A valid empc purge reply was received by this MPC for an entry whose cache ID is printed and the entry is being deleted.

MPOA.043

Level: UE-ERROR

Short Syntax: MPOA.043 eMPC purge reply recvd as a NAK (vpi/vci *vpi/ vci*)

Long Syntax: MPOA.043 eMPC purge reply recvd as a NAK (vpi/vci *vpi/ vci*)

Description: An empc purge reply was received by this MPC on the specified VCC but indicated a NAKed response. The packet will be dropped without further processing.

MPOA.044

Level: C-INFO

Short Syntax: MPOA.044 eMPC recvd LEC down! (lec *lecIntfNum*)

Long Syntax: MPOA.044 eMPC recvd LEC down! (lec *lecIntfNum*)

Description: The empc received an (internal) message indicating that the specified LEC has been disabled. e-cache entries imposed over this LEC will be disabled and/or purged.

MPOA.045

Level: C-INFO

Short Syntax: MPOA.045 eMPC recvd LAN destn unreg (lec *lecIntfNum*)

Long Syntax: MPOA.045 eMPC recvd LAN destn unreg (lec *lecIntfNum*)

Description: The empc received an (internal) message indicating that the specified LEC has deregistered a LAN destination. e-cache entries imposed over this LEC for this LAN destination will be disabled and/or purged.

MPOA.046

Level: C-INFO

Short Syntax: MPOA.046 eMPC recvd mps down (MPS *mpsAtmAddr*)

Long Syntax: MPOA.046 eMPC recvd mps down (MPS *mpsAtmAddr*)

Description: The empc received an (internal) message indicating that an MPS went down. The ATM address of the MPS is printed. e-cache entries imposed by this MPS will be invalidated and data plane purges may be sent for each such entry.

MPOA.047

Level: C-INFO

Short Syntax: MPOA.047 eMPC recvd mps mac addr invalid (MPS *mpsAtmAddr*) (MAC *macAddr*)

Long Syntax: MPOA.047 eMPC recvd mps mac addr invalid (MPS *mpsAtmAddr*) (MAC *macAddr*)

Description: The empc received an (internal) message indicating that an MPS MAC address association is no longer valid. e-cache entries containing DLL information with this MAC address as the source MAC address will be invalidated and data plane purges may be sent for each such entry.

MPOA.048

Level: C-INFO

Short Syntax: MPOA.048 eMPC deleting imposed e-cache entry (addr/mask *addr/ addrMask*, cache id x *cacheId*)

Long Syntax: MPOA.048 eMPC deleting imposed e-cache entry (addr/mask *addr/ addrMask*, cache id x *cacheId*)

Description: An externally imposed egress MPC cache entry is being deleted. The protocol address, address mask and cache ID are printed.

MPOA.049

Level: C-INFO

Short Syntax: MPOA.049 Deleting intern. derived e-cache entry (addr/mask *addr/ addrMask*, cache id x *cacheId*)

Long Syntax: MPOA.049 Deleting intern. derived e-cache entry (addr/mask *addr/ addrMask*, cache id x *cacheId*)

Description: An internally derived egress MPC cache entry is being deleted. The protocol address, address mask and cache ID are printed.

MPOA.050

Level: C-INFO

Short Syntax: MPOA.050 eMPC: Nontag 1483 next hop ambiguity! (new entry cache id x *cacheId*)

Long Syntax: MPOA.050 eMPC: Nontag 1483 next hop ambiguity! (new entry cache id x *cacheId*)

Description: The next hop ambiguity case was encountered for this imposition request. This leads to two different e-cache entries with the same source ATM address and destination protocol addresses. In this case, the already existing entry will be overwritten with the new information. The cacheId of the new entry is printed.

MPOA.051

Level: C-INFO

Short Syntax: MPOA.051 eMPC purge reply rqst id mismatch! (msg cache id x *cacheId*)

Long Syntax: MPOA.051 eMPC purge reply rqst id mismatch! (msg cache id x *cacheId*)

Description: An E-MPC initiated Purge Reply was received for a matching egress cache entry which was being purged, however the request ID in the reply did not match that sent out so the reply will be ignored.

MPOA.052

Level: UE-ERROR

Short Syntax: MPOA.052 Bad tag in recvd pkt! (*receivedTag*)

Long Syntax: MPOA.052 Bad tag in recvd pkt! (*receivedTag*)

Description: An MPOA Tagged packet was received with an invalid tag. The tag in the pkt is printed.

MPOA.053

Level: UE-ERROR

Short Syntax: MPOA.053 Destn prot addr mismatch in recvd pkt! (dstn *pktDstnProtAddr*, entry *entryProtAddr/ entryAddrMask*)

Long Syntax: MPOA.053 Destn prot addr mismatch in recvd pkt! (dstn *pktDstnProtAddr*, entry *entryProtAddr/ entryAddrMask*)

Description: The tag based lookup for a received MPOA tagged packet matched to an e-cache entry for a different destination protocol address range than the destination protocol address in the packet. The destn address in the packet, and the address/mask combination of the e-cache entry are printed.

MPOA.054

Level: UE-ERROR

Short Syntax: MPOA.054 Src atm addr mismatch in recvd pkt! (entry *entryAtmAddr*, vpi/vci *vpi/ vci*)

Long Syntax: MPOA.054 Src atm addr mismatch in recvd pkt! (entry *entryAtmAddr*, vpi/vci *vpi/ vci*)

Description: The tag based lookup for a received MPOA tagged packet matched to an e-cache entry for a different source ATM address than that where the packet came from. The src atm address in the packet, and the vpi/vci of the VCC the pkt came on are printed.

MPOA.055

Level: U-INFO

Short Syntax: MPOA.055 Tagged pkt: matching entry inactive! (entry *entryProtAddress/ entryProtAddressMask*, state *entryState*)

Long Syntax: MPOA.055 Tagged pkt: matching entry inactive! (entry *entryProtAddress/ entryProtAddressMask*, state *entryState*)

Description: The tag based lookup for a received MPOA tagged packet matched to an e-cache entry which was not in an active state. The protocol address and address mask of the matching entry are printed alongwith the state of the entry.

MPOA.056

Level: UI-ERROR

Short Syntax: MPOA.056 Bridge unable to deliver data packet (rc *returnCode*)

Long Syntax: MPOA.056 Bridge unable to deliver data packet (rc *returnCode*)

Description: The bridge was unable to successfully bridge the received data packet. The return code from the bridge is printed.

MPOA.057

Level: C-INFO

Short Syntax: MPOA.057 Recvd tagged data pkt! (tag *receivedTag*, vpi/vci /)

Long Syntax: MPOA.057 Recvd tagged data pkt! (tag *receivedTag*, vpi/vci /)

Description: An MPOA Tagged packet was received. The tag in the pkt is printed.

MPOA.058

Level: C-INFO

Short Syntax: MPOA.058 Recvd 1483 IP data pkt! (*pktDstIpAddr*)

Long Syntax: MPOA.058 Recvd 1483 IP data pkt! (*pktDstIpAddr*)

Description: An MPOA Nontagged 1483 IP packet was received. The destination IP address in the pkt is printed.

MPOA.059

Level: U-INFO

Short Syntax: MPOA.059 1483 pkt: matching entry inactive! (entry *entryProtAddress/ entryProtAddressMask*, state *entryState*)

Long Syntax: MPOA.059 1483 pkt: matching entry inactive! (entry *entryProtAddress/ entryProtAddressMask*, state *entryState*)

Description: The tag based lookup for a received MPOA non-tagged 1483 packet matched to an e-cache entry which was not in an active state. The protocol address and address mask of the matching entry are printed alongwith the state of the entry.

MPOA.060

Level: C-INFO

Short Syntax: MPOA.060 1483 pkt hash cache miss! (*pktDstIpAddr*)

Long Syntax: MPOA.060 1483 pkt hash cache miss! (*pktDstIpAddr*)

Description: A received MPOA Nontagged 1483 IP packet caused a hash array miss. The destination IP address in the pkt is printed.

MPOA.061

Level: P-TRACE

Short Syntax: MPOA.061 mpoa client ATM contrl frame trace

Long Syntax: MPOA.061 mpoa client ATM contrl frame trace

Description: Packet trace for MPOA client ATM control frame.

MPOA.062

Level: P-TRACE

Short Syntax: MPOA.062 mpoa client ATM data frame trace

Long Syntax: MPOA.062 mpoa client ATM data frame trace

Description: Packet trace for MPOA client ATM data frame.

MPOA.063

Level: P-TRACE

Short Syntax: MPOA.063 mpoa client LAN frame trace

Long Syntax: MPOA.063 mpoa client LAN frame trace

Description: Packet trace for MPOA client LAN frame.

MPOA.064

Level: U-INFO

Short Syntax: MPOA.064 Bad recvd pkt: sending DPP rqst! (*pktDstIpAddr*)

Long Syntax: MPOA.064 Bad recvd pkt: sending DPP rqst! (*pktDstIpAddr*)

Description: A Data Plane Purge Request is being sent because of a received packet for which no matching egress cache entry was found. The destination protocol address in the pkt is printed.

MPOA.065

Level: UI-ERROR

Short Syntax: MPOA.065 MPC create failed! (intf *netNum*, rc *returnCode*)

Long Syntax: MPOA.065 MPC create failed! (intf *netNum*, rc *returnCode*)

Description: The initial MPOA client creation process failed on the specified ATM interface number with the specified return code. This indicates an internal error such as a memory allocation failure or some failure involving the ATM interface. The MPOA client will not come up unless the cause of the failure is fixed.

MPOA.066

Level: U-INFO

Short Syntax: MPOA.066 No MPC sram record. Will use defaults! (*netNum*)

Long Syntax: MPOA.066 No MPC sram record. Will use defaults! (*netNum*)

Description: No SRAM configuration record was found for the MPOA client. The client will hence come up with a default set of parameters.

MPOA.067

Level: UE-ERROR

Short Syntax: MPOA.067 Mismatched MPC sram record. Will use defaults! (*netNum*)

Long Syntax: MPOA.067 Mismatched MPC sram record. Will use defaults! (*netNum*)

Description: An SRAM configuration record was found for the MPOA client but for a different ATM interface than the one coming up. The existing SRAM record configuration parameters will hence be ignored and the client will hence come up with a default set of parameters. This could indicate a misconfiguration.

MPOA.068

Level: C-INFO

Short Syntax: MPOA.068 Matching MPC sram record found, initializing. (*netNum*)

Long Syntax: MPOA.068 Matching MPC sram record found, initializing. (*netNum*)

Description: A matching SRAM configuration record was found for the MPOA client. The client will now try to come up with the configured set of parameters.

MPOA.069

Level: U-INFO

Short Syntax: MPOA.069 Updated MPC line rate/bw config parms! (*netNum*)

Long Syntax: MPOA.069 Updated MPC line rate/bw config parms! (*netNum*)

Description: The line rate and/or bandwidth related configuration paramaters in an MPC configuration

record were incorrect in relation to the line rate of the ATM interface on which the MPC is coming up. These were updated to values which are correct in relation to interface line rate. This may indicate a configuration error by the user.

MPOA.070

Level: UE-ERROR

Short Syntax: MPOA.070 Recvd frame too large to bridge! (*lecNetNum*)

Long Syntax: MPOA.070 Recvd frame too large to bridge! (*lecNetNum*)

Description: A received frame was too large for the egress LEC to be bridged. The net number of the egress LEC is printed. This could because of an external error at the ingress or because of a misconfiguration of mtus between the e-mps and the e-mpc.

MPOA.071

Level: U-INFO

Short Syntax: MPOA.071 eMPC: purge reply from unknown mps (*vpi*, *vci*)

Long Syntax: MPOA.071 eMPC: purge reply from unknown mps (*vpi*, *vci*)

Description: An e-mpc initiated purge reply was received on the specified VCC but the other end of this VCC is not currently known to be an MPS.

MPOA.072

Level: U-INFO

Short Syntax: MPOA.072 MPC (*atmIntfNum*) STARTING

Long Syntax: MPOA.072 MPC (*atmIntfNum*) STARTING

Description: MPC instance is starting operation.

MPOA.073

Level: UI-ERROR

Short Syntax: MPOA.073 MPC (*atmIntfNum*) TERMINATING: *errString*

Long Syntax: MPOA.073 MPC (*atmIntfNum*) TERMINATING: *errString*

Description: MPC instance is terminating operation due to error.

MPOA.074

Level: U-INFO

Short Syntax: MPOA.074 MPC (*atmIntfNum*) DELETED

Long Syntax: MPOA.074 MPC (*atmIntfNum*) DELETED

Description: MPC instance is being deleted.

MPOA.075

Level: U-INFO

Short Syntax: MPOA.075 MPC (*atmIntfNum*) STOPPED

Long Syntax: MPOA.075 MPC (*atmIntfNum*) STOPPED

Description: Operation of MPC instance is being stopped.

MPOA.076

Level: U-INFO

Short Syntax: MPOA.076 MPC (*atmIntfNum*) RESTARTING

Long Syntax: MPOA.076 MPC (*atmIntfNum*) RESTARTING

Description: Operation of MPC instance is being restarted.

MPOA.077

Level: UI-ERROR

Short Syntax: MPOA.077 MPC (*atmIntfNum*) => DOWN: ATM user reg failed: *errString*

Long Syntax: MPOA.077 MPC (*atmIntfNum*) => DOWN: ATM user reg failed: *errString*

Description: Registration as user of ATM interface failed. MPC going down as result.

MPOA.078

Level: UI-ERROR

Short Syntax: MPOA.078 MPC (*atmIntfNum*) => DOWN: ATM set user name failed: *errString*

Long Syntax: MPOA.078 MPC (*atmIntfNum*) => DOWN: ATM set user name failed: *errString*

Description: ATM interface call to set user name failed. MPC going down as result.

MPOA.079

Level: U-INFO

Short Syntax: MPOA.079 MPC (*atmIntfNum*): waiting for ATM net UP

Long Syntax: MPOA.079 MPC (*atmIntfNum*): waiting for ATM net UP

Description: MPC operation suspended until ATM net comes up.

MPOA.080

Level: UE-ERROR

Short Syntax: MPOA.080 MPC (*atmIntfNum*): ATM net DOWN

Long Syntax: MPOA.080 MPC (*atmIntfNum*): ATM net DOWN

Description: ATM net has gone done. MPC operation will be suspended until ATM net comes back up.

MPOA.081

Level: U-INFO

Short Syntax: MPOA.081 MPC (*atmIntfNum*): ATM net UP

Long Syntax: MPOA.081 MPC (*atmIntfNum*): ATM net UP

Description: ATM net has come up. MPC operation will be continued.

MPOA.082

Level: UI-ERROR

Short Syntax: MPOA.082 MPC (*atmIntfNum*) => DOWN: ATM addr activation failed: *errString*

Long Syntax: MPOA.082 MPC (*atmIntfNum*) => DOWN: ATM addr activation failed: *errString*

Description: ATM interface call to initiate activation of MPC's ATM address (i.e., registration with switch) failed. MPC going down as result.

MPOA.083

Level: U-INFO

Short Syntax: MPOA.083 MPC (*atmIntfNum*): waiting for ATM addr activation

Long Syntax: MPOA.083 MPC (*atmIntfNum*): waiting for ATM addr activation

Description: Activation of MPC's ATM address has been initiated. MPC operation will be suspended until activation completes.

MPOA.084

Level: UE-ERROR

Short Syntax: MPOA.084 MPC (*atmIntfNum*): ATM addr activation timed out: retrying

Long Syntax: MPOA.084 MPC (*atmIntfNum*): ATM addr activation timed out: retrying

Description: No response has been received in response to request to activate MPC's ATM address. Activation request will be retried.

MPOA.085

Level: UE-ERROR

Short Syntax: MPOA.085 MPC (*atmIntfNum*): ATM addr rejected by switch: retry timer started

Long Syntax: MPOA.085 MPC (*atmIntfNum*): ATM addr rejected by switch: retry timer started

Description: Request to activate MPC's ATM address was rejected by switch. Activation request will be retried after short delay.

MPOA.086

Level: UE-ERROR

Short Syntax: MPOA.086 MPC (*atmIntfNum*): ATM addr deactivated: reactivating

Long Syntax: MPOA.086 MPC (*atmIntfNum*): ATM addr deactivated: reactivating

Description: MPC's ATM address was deactivated. Reactivation will be initiated.

MPOA.087

Level: U-INFO

Short Syntax: MPOA.087 MPC (*atmIntfNum*): timer expired, retrying ATM addr activation

Long Syntax: MPOA.087 MPC (*atmIntfNum*): timer expired, retrying ATM addr activation

Description: ATM address activation retry timer has expired. Request to activate MPC's ATM address will be retried now.

MPOA.088

Level: UI-ERROR

Short Syntax: MPOA.088 MPC (*atmIntfNum*): unexpected ATM addr activation timer expiration

Long Syntax: MPOA.088 MPC (*atmIntfNum*): unexpected ATM addr activation timer expiration

Description: ATM address activation retry timer has expired unexpectedly.

MPOA.089

Level: U-INFO

Short Syntax: MPOA.089 MPC (*atmIntfNum*): ATM addr activated

Long Syntax: MPOA.089 MPC (*atmIntfNum*): ATM addr activated

Description: MPC's ATM address has been activated. MPC operation will now continue.

MPOA.090

Level: UI-ERROR

Short Syntax: MPOA.090 MPC (*atmIntfNum*) => DOWN: err reading ATM addr: *errString*

Long Syntax: MPOA.090 MPC (*atmIntfNum*) => DOWN: err reading ATM addr: *errString*

Description: ATM interface call to read MPC's ATM address failed. MPC going down as result.

MPOA.091

Level: UI-ERROR

Short Syntax: MPOA.091 MPC (*atmIntfNum*) => DOWN: err reading UNI version: *errString*

Long Syntax: MPOA.091 MPC (*atmIntfNum*) => DOWN: err reading UNI version: *errString*

Description: ATM interface call to read UNI version failed. MPC going down as result.

MPOA.092

Level: U-INFO

Short Syntax: MPOA.092 MPC (*atmIntfNum*): waiting for UNI version report

Long Syntax: MPOA.092 MPC (*atmIntfNum*): waiting for UNI version report

Description: MPC operation suspended until UNI version is reported by ATM subsystem.

MPOA.093

Level: U-INFO

Short Syntax: MPOA.093 MPC (*atmIntfNum*): UNI version reported

Long Syntax: MPOA.093 MPC (*atmIntfNum*): UNI version reported

Description: Version of UNI being run reported by ATM subsystem.

MPOA.094

Level: UI-ERROR

Short Syntax: MPOA.094 MPC (*atmIntfNum*) => DOWN: err opening LLC Call SAP: *errString*

Long Syntax: MPOA.094 MPC (*atmIntfNum*) => DOWN: err opening LLC Call SAP: *errString*

Description: ATM interface primitive to open SAP for receiving LLC calls failed. MPC going down as result.

MPOA.095

Level: UI-ERROR

Short Syntax: MPOA.095 MPC (*atmIntfNum*) => DOWN: err opening LANE Call SAP: *errString*

Long Syntax: MPOA.095 MPC (*atmIntfNum*) => DOWN: err opening LANE Call SAP: *errString*

Description: ATM interface primitive to open SAP for receiving LANE calls failed. MPC going down as result.

MPOA.096

Level: UI-ERROR

Short Syntax: MPOA.096 MPC (*atmIntfNum*) => DOWN: err opening ATM Frame SAP: *errString*

Long Syntax: MPOA.096 MPC (*atmIntfNum*) => DOWN: err opening ATM Frame SAP: *errString*

Description: ATM interface call to open SAP for transferring ATM frames failed. MPC going down as result.

MPOA.097

Level: UI-ERROR

Short Syntax: MPOA.097 MPC (*atmIntfNum*) => DOWN: registration with LEC failed: *errString*

Long Syntax: MPOA.097 MPC (*atmIntfNum*) => DOWN: registration with LEC failed: *errString*

Description: Call to register as user of LEC interface failed. MPC going down as result.

MPOA.098

Level: UI-ERROR

Short Syntax: MPOA.098 MPC (*atmIntfNum*) => DOWN: egress init failed: *errString*

Long Syntax: MPOA.098 MPC (*atmIntfNum*) => DOWN: egress init failed: *errString*

Description: Call to initialize egress MPC functions failed. MPC going down as result.

MPOA.099

Level: UI-ERROR

Short Syntax: MPOA.099 MPC: recv unexpected LECS addr list report

Long Syntax: MPOA.099 MPC: recv unexpected LECS addr list report

Description: Received unexpected LECS address list report from ATM subsystem.

MPOA.100

Level: UI-ERROR

Short Syntax: MPOA.100 MPC: recv unexpected ATM disconnect leaf message

Long Syntax: MPOA.100 MPC: recv unexpected ATM disconnect leaf message

Description: Received unexpected disconnect leaf message from ATM subsystem.

MPOA.101

Level: UI-ERROR

Short Syntax: MPOA.101 MPC: recv unexpected ATM add leaf ack message

Long Syntax: MPOA.101 MPC: recv unexpected ATM add leaf ack message

Description: Received unexpected add leaf acknowledgement message from ATM subsystem.

MPOA.102

Level: U-INFO

Short Syntax: MPOA.102 MPC (*atmIntfNum*): LEC (*lecIntfNum*) UP

Long Syntax: MPOA.102 MPC (*atmIntfNum*): LEC (*lecIntfNum*) UP

Description: A MPC LEC has become operational.

MPOA.103

Level: UI-ERROR

Short Syntax: MPOA.103 MPC (*atmIntfNum*): LEC (*lecIntfNum*) IGNORED: cntrl blk alloc failed

Long Syntax: MPOA.103 MPC (*atmIntfNum*): LEC (*lecIntfNum*) IGNORED: cntrl blk alloc failed

Description: Ignoring MPC LEC that has become operational because control block allocation failed.

MPOA.104

Level: UI-ERROR

Short Syntax: MPOA.104 MPC (*atmIntfNum*): LEC (*lecIntfNum*) IGNORED: ELAN ID database insertion failed

Long Syntax: MPOA.104 MPC (*atmIntfNum*): LEC (*lecIntfNum*) IGNORED: ELAN ID database insertion failed

Description: Ignoring MPC LEC that has become operational because ELAN ID database insertion failed.

MPOA.105

Level: UE-ERROR

Short Syntax: MPOA.105 MPC (*atmIntfNum*): LEC (*lec1IntfNum*) IGNORED: same ELAN ID as LEC (*lec2IntfNum*), but different ELAN type

Long Syntax: MPOA.105 MPC (*atmIntfNum*): LEC (*lec1IntfNum*) IGNORED: same ELAN ID as LEC (*lec2IntfNum*), but different ELAN type

Description: Ignoring MPC LEC that has become operational because the LEC is associated with the same ELAN ID as another MPC LEC, but the LECs are for different types of ELANs.

MPOA.106

Level: U-INFO

Short Syntax: MPOA.106 MPC (*atmIntfNum*): not accepting shortcuts for ELAN ID *elanId*: multiple Ethernet LECs on same ELAN

Long Syntax: MPOA.106 MPC (*atmIntfNum*): not accepting shortcuts for ELAN ID *elanId*: multiple Ethernet LECs on same ELAN

Description: The MPC is not accepting shortcuts for an ELAN because there are multiple local Ethernet LECs on the ELAN.

MPOA.107

Level: U-INFO

Short Syntax: MPOA.107 MPC (*atmIntfNum*): resuming shortcut acceptance for ELAN ID *elanId*

Long Syntax: MPOA.107 MPC (*atmIntfNum*): resuming shortcut acceptance for ELAN ID *elanId*

Description: The MPC has resumed accepting shortcuts for an ELAN.

MPOA.108

Level: U-INFO

Short Syntax: MPOA.108 MPC (*atmIntfNum*): LEC (*lecIntfNum*) DOWN

Long Syntax: MPOA.108 MPC (*atmIntfNum*): LEC (*lecIntfNum*) DOWN

Description: A MPC LEC is no longer operational.

MPOA.109

Level: U-INFO

Short Syntax: MPOA.109 MPC (*atmIntfNum*): LEC (*lecIntfNum*) registered route descriptor (x *rd*)

Long Syntax: MPOA.109 MPC (*atmIntfNum*): LEC (*lecIntfNum*) registered route descriptor (x *rd*)

Description: A MPC LEC has registered a route descriptor with the LES.

MPOA.110

Level: UI-ERROR

Short Syntax: MPOA.110 MPC (*atmIntfNum*): route descriptor (x *rd*) IGNORED: cntrl blk alloc failed

Long Syntax: MPOA.110 MPC (*atmIntfNum*): route descriptor (x *rd*) IGNORED: cntrl blk alloc failed

Description: The MPC is ignoring a route descriptor that was registered with the LES by a MPC LEC because allocation of a control block failed.

MPOA.111

Level: U-INFO

Short Syntax: MPOA.111 MPC (*atmIntfNum*): LEC (*lecIntfNum*) deregistered route descriptor (x *rd*)

Long Syntax: MPOA.111 MPC (*atmIntfNum*): LEC (*lecIntfNum*) deregistered route descriptor (x *rd*)

Description: A MPC LEC has deregistered a route descriptor with the LES.

MPOA.112

Level: CE-ERROR

Short Syntax: MPOA.112 MPC (*atmIntfNum*): ignored bad MPOA Device TLV recv by LEC (*lecIntfNum*)

Long Syntax: MPOA.112 MPC (*atmIntfNum*): ignored bad MPOA Device TLV recv by LEC (*lecIntfNum*)

Description: The MPC has ignored a bad MPOA Device TLV received via a LEC.

MPOA.113

Level: UI-ERROR

Short Syntax: MPOA.113 MPC (*atmIntfNum*): MPS IGNORED: ATM addr database insertion failed: LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.113 MPC (*atmIntfNum*): MPS IGNORED: ATM addr database insertion failed: LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Description: The MPC is ignoring a discovered MPS because insertion of a database entry for the MPS's ATM address failed.

MPOA.114

Level: UI-ERROR

Short Syntax: MPOA.114 MPC (*atmIntfNum*): MPS IGNORED: cntrl blk alloc failed: LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.114 MPC (*atmIntfNum*): MPS IGNORED: cntrl blk alloc failed: LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Description: The MPC is ignoring a discovered MPS because allocation of a control block failed.

MPOA.115

Level: U-INFO

Short Syntax: MPOA.115 MPC (*atmIntfNum*): MPS DISCOVERED: LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.115 MPC (*atmIntfNum*): MPS DISCOVERED: LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Description: The MPC has discovered a MPS.

MPOA.116

Level: U-INFO

Short Syntax: MPOA.116 MPC (*atmIntfNum*): MPS DELETED: *reasonString*: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.116 MPC (*atmIntfNum*): MPS DELETED: *reasonString*: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC has deleted the control block for a discovered MPS.

MPOA.117

Level: U-INFO

Short Syntax: MPOA.117 MPC (*atmIntfNum*): MPS MAC Addr (x *macAddr*) DISCOVERED: LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.117 MPC (*atmIntfNum*): MPS MAC Addr (x *macAddr*) DISCOVERED: LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Description: The MPC has discovered a MAC address associated with a MPS.

MPOA.118

Level: U-INFO

Short Syntax: MPOA.118 MPC (*atmIntfNum*): refreshed MPS MAC addr (x *macAddr*): LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.118 MPC (*atmIntfNum*): refreshed MPS MAC Addr (x *macAddr*): LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Description: The association of a MAC address with a MPS has been refeshed.

MPOA.119

Level: UI-ERROR

Short Syntax: MPOA.119 MPC (*atmIntfNum*): MPS MAC addr (x *macAddr*) IGNORED: cntrl blk alloc failed: LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.119 MPC (*atmIntfNum*): MPS MAC addr (x *macAddr*) IGNORED: cntrl blk alloc failed: LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Description: The MPC has ignored discovery of a MAC address associated with a MPS because allocation of a control block failed.

MPOA.120

Level: UI-ERROR

Short Syntax: MPOA.120 MPC (*atmIntfNum*): MPS MAC addr (x *macAddr*) IGNORED: bridge reg failed: LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.120 MPC (*atmIntfNum*): MPS MAC addr (x *macAddr*) IGNORED: bridge reg failed: LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Description: The MPC has ignored discovery of a MAC address associated with a MPS because registration of the MAC address with the bridge failed.

MPOA.121

Level: U-INFO

Short Syntax: MPOA.121 MPC (*atmIntfNum*): MPS MAC addr (x *macAddr*) DELETED: LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.121 MPC (*atmIntfNum*): MPS MAC addr (x *macAddr*) DELETED: LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Description: The MPC has deleted the control block for a MAC address associated with a MPS.

MPOA.122

Level: U-INFO

Short Syntax: MPOA.122 MPC (*atmIntfNum*): recv ' *tlvString*' config TLV from LECS: LEC (*lecIntfNum*): value = *parmValue*

Long Syntax: MPOA.122 MPC (*atmIntfNum*): recv ' *tlvString*' config TLV from LECS: LEC (*lecIntfNum*): value = *parmValue*

Description: The MPC has received a configuration TLV from the LECS.

MPOA.123

Level: U-INFO

Short Syntax: MPOA.123 MPC (*atmIntfNum*): recv ' *enable_disable_string* IP' config TLV from LECS: LEC (*lecIntfNum*)

Long Syntax: MPOA.123 MPC (*atmIntfNum*): recv ' *enable_disable_string* IP' config TLV from LECS: LEC (*lecIntfNum*)

Description: The MPC has received a configuration TLV from the LECS to control whether IP flow detection is enabled or disabled.

MPOA.124

Level: CE-ERROR

Short Syntax: MPOA.124 MPC (*atmIntfNum*): recv invalid ' *tlvString*' config TLV from LECS: LEC (*lecIntfNum*): value = *parmValue*

Long Syntax: MPOA.124 MPC (*atmIntfNum*): recv invalid ' *tlvString*' config TLV from LECS: LEC (*lecIntfNum*): value = *parmValue*

Description: The MPC has received an invalid configuration TLV from the LECS.

MPOA.125

Level: U-INFO

Short Syntax: MPOA.125 MPC (*atmIntfNum*): recv config TLV for unknown protocol from LECS: LEC (*lecIntfNum*): protocol short = x *shortProtocol*, long = x *longProtocol*

Long Syntax: MPOA.125 MPC (*atmIntfNum*): recv config TLV for unknown protocol from LECS: LEC (*lecIntfNum*): protocol short = x *shortProtocol*, long = x *longProtocol*

Description: The MPC has received a configuration TLV from the LECS for an unknown protocol.

MPOA.126

Level: U-INFO

Short Syntax: MPOA.126 MPC (*atmIntfNum*): recv unknown config TLV from LECS: LEC (*lecIntfNum*): type = x *tlvType*

Long Syntax: MPOA.126 MPC (*atmIntfNum*): recv unknown config TLV from LECS: LEC (*lecIntfNum*): type = x *tlvType*

Description: The MPC has received a configuration TLV from the LECS with an unknown type.

MPOA.127

Level: UE-ERROR

Short Syntax: MPOA.127 MPC (*atmIntfNum*): recv *msgTypeString* with no src ATM addr: Remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.127 MPC (*atmIntfNum*): recv *msgTypeString* with no src ATM addr: Remote ATM addr = x *remoteAtmAddr*

Description: The MPC has received a control message that does not contain a source ATM address.

MPOA.128

Level: C-INFO

Short Syntax: MPOA.128 MPC (*atmIntfNum*): recv MPS keep-alive: lifetime = *lifeTime*: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.128 MPC (*atmIntfNum*): recv MPS keep-alive: lifetime = *lifeTime*: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC has received a keep-alive message from a MPS.

MPOA.129

Level: UE-ERROR

Short Syntax: MPOA.129 MPC (*atmIntfNum*): recv keep-alive with no extension: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.129 MPC (*atmIntfNum*): recv keep-alive with no extension: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC received a keep-alive message with no extension from a MPS.

MPOA.130

Level: UE-ERROR

Short Syntax: MPOA.130 MPC (*atmIntfNum*): recv invalid MPS keep-alive: *errString*: MPS ATM addr = x *mpsAtmAddr*: bad value = *badValue*

Long Syntax: MPOA.130 MPC (*atmIntfNum*): recv invalid MPS keep-alive: *errString*: MPS ATM addr = x *mpsAtmAddr*: bad value = *badValue*

Description: The MPC received an invalid keep-alive message from a MPS.

MPOA.131

Level: C-INFO

Short Syntax: MPOA.131 MPC (*atmIntfNum*): recv keep-alive for unknown MPS: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.131 MPC (*atmIntfNum*): recv keep-alive for unknown MPS: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC received a keep-alive message for an unknown MPS.

MPOA.132

Level: C-INFO

Short Syntax: MPOA.132 MPC (*atmIntfNum*): recv purge request: previous-hop ATM addr = x *atmAddr*

Long Syntax: MPOA.132 MPC (*atmIntfNum*): recv purge request: previous-hop ATM addr = x *atmAddr*

Description: The MPC has received a purge request message.

MPOA.133

Level: CE-ERROR

Short Syntax: MPOA.133 MPC (*atmIntfNum*): initiating err ind: *errString*: next-hop ATM addr = x *atmAddr*

Long Syntax: MPOA.133 MPC (*atmIntfNum*): initiating err ind: *errString*: next-hop ATM addr = x *atmAddr*

Description: The MPC is initiating transmission of an error indication message.

MPOA.134

Level: C-INFO

Short Syntax: MPOA.134 MPC (*atmIntfNum*): cntrl frm not sent: *errString*: next-hop ATM addr = x *atmAddr*

Long Syntax: MPOA.134 MPC (*atmIntfNum*): cntrl frm not sent: *errString*: next-hop ATM addr = x *atmAddr*

Description: A control frame for whic transmission was initiated was not sent.

MPOA.135

Level: CE-ERROR

Short Syntax: MPOA.135 MPC (*atmIntfNum*): invalid CIE in purge req: *errString*: previous-hop ATM addr = x *atmAddr*

Long Syntax: MPOA.135 MPC (*atmIntfNum*): invalid CIE in purge req: *errString*: previous-hop ATM addr = x *atmAddr*

Description: The MPC received a purge request that contained an invalid CIE.

MPOA.136

Level: C-INFO

Short Syntax: MPOA.136 MPC (*atmIntfNum*): purging IP shortcut: IP addr = *ipAddr*, prefix len = *prefixLength*

Long Syntax: MPOA.136 MPC (*atmIntfNum*): purging IP shortcut: IP addr = *ipAddr*, prefix len = *prefixLength*

Description: The MPC has purged an IP shortcut.

MPOA.137

Level: C-INFO

Short Syntax: MPOA.137 MPC (*atmIntfNum*): initiating purge reply: next-hop ATM addr = x *atmAddr*

Long Syntax: MPOA.137 MPC (*atmIntfNum*): initiating purge reply: next-hop ATM addr = x *atmAddr*

Description: The MPC is initiating transmission of a purge reply message.

MPOA.138

Level: C-INFO

Short Syntax: MPOA.138 MPC (*atmIntfNum*): recv trigger for IP addr = *ipAddr*: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.138 MPC (*atmIntfNum*): recv trigger for IP addr = *ipAddr*: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC received a trigger request for an IP address.

MPOA.139

Level: CE-ERROR

Short Syntax: MPOA.139 MPC (*atmIntfNum*): recv invalid trigger: *errString*: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.139 MPC (*atmIntfNum*): recv invalid trigger: *errString*: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC received an invalid trigger request.

MPOA.140

Level: CE-ERROR

Short Syntax: MPOA.140 MPC (*atmIntfNum*): recv trigger for unknown MPS: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.140 MPC (*atmIntfNum*): recv trigger for unknown MPS: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC received a trigger request for an unknown MPS.

MPOA.141

Level: UI-ERROR

Short Syntax: MPOA.141 MPC (*atmIntfNum*): shortcut cntrl blk alloc failed: IP addr/mask = *ipAddr/ ipMask*

Long Syntax: MPOA.141 MPC (*atmIntfNum*): shortcut cntrl blk alloc failed: IP addr/mask = *ipAddr/ ipMask*

Description: Allocation of a shortcut control block failed.

MPOA.142

Level: C-INFO

Short Syntax: MPOA.142 MPC (*atmIntfNum*): shortcut cntrl blk DELETED: *ipAddr*: IP addr/mask = *ipMask/*

Long Syntax: MPOA.142 MPC (*atmIntfNum*): shortcut cntrl blk DELETED: *ipAddr*: IP addr/mask = *ipMask/*

Description: The MPC has deleted a shortcut route.

MPOA.143

Level: C-INFO

Short Syntax: MPOA.143 MPC (*atmIntfNum*): initiating resolution req: IP addr = *ipAddr*: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.143 MPC (*atmIntfNum*): initiating resolution req: IP addr = *ipAddr*: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC is initiating transmission of a resolution request for an IP address.

MPOA.144

Level: C-INFO

Short Syntax: MPOA.144 MPC (*atmIntfNum*): recv resolution reply: IP addr = *ipAddr/ ipMask*: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.144 MPC (*atmIntfNum*): recv

resolution reply: IP addr = *ipAddr/ ipMask*: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC received a resolution reply for an IP destination.

MPOA.145

Level: CE-ERROR

Short Syntax: MPOA.145 MPC (*atmIntfNum*): discarded resolution reply: *reasonString*: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.145 MPC (*atmIntfNum*): discarded resolution reply: *reasonString*: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC has discarded a resolution reply that it received.

MPOA.146

Level: CE-ERROR

Short Syntax: MPOA.146 MPC (*atmIntfNum*): recv invalid resolution reply: *errString*: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.146 MPC (*atmIntfNum*): recv invalid resolution reply: *errString*: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC has received an invalid resolution reply.

MPOA.147

Level: CE-ERROR

Short Syntax: MPOA.147 MPC (*atmIntfNum*): recv resolution NAK: *nakString*: IP addr = *ipAddr*: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.147 MPC (*atmIntfNum*): recv resolution NAK: *nakString*: IP addr = *ipAddr*: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC has received a negative acknowledgement in a resolution reply for an IP address.

MPOA.148

Level: CE-ERROR

Short Syntax: MPOA.148 MPC (*atmIntfNum*): ignored LANE ext: *errString*: IP addr = *ipAddr*: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.148 MPC (*atmIntfNum*): ignored LANE ext: *errString*: IP addr = *ipAddr*: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC is ignoring invalid LANE extensions in a resolution reply for an IP address.

MPOA.149

Level: C-INFO

Short Syntax: MPOA.149 MPC (*atmIntfNum*): converting route: IP addr/mask *errString/ ipAddr1* => *ipMask1/ ipAddr2*

Long Syntax: MPOA.149 MPC (*atmIntfNum*): converting route: IP addr/mask *errString/ ipAddr1* => *ipMask1/ ipAddr2*

Description: The MPC is converting an existing route based on the contents of a resolution reply that it received for an IP address. The conversion may be from host route to network route or vice versa.

MPOA.150

Level: C-INFO

Short Syntax: MPOA.150 MPC (*atmIntfNum*): route conversion failed: *reasonString*

Long Syntax: MPOA.150 MPC (*atmIntfNum*): route conversion failed: *reasonString*

Description: Conversion of an existing route to a new type failed. The failed conversion may have been host route to network route or vice versa.

MPOA.151

Level: C-INFO

Short Syntax: MPOA.151 MPC (*atmIntfNum*): shortcut associated with VCC: IP addr/mask = *ipAddr/ ipMask*

Long Syntax: MPOA.151 MPC (*atmIntfNum*): shortcut associated with VCC: IP addr/mask = *ipAddr/ ipMask*

Description: The MPC has associated a shortcut route with a VCC.

MPOA.152

Level: C-INFO

Short Syntax: MPOA.152 MPC (*atmIntfNum*): shortcut disassociated from VCC: IP addr/mask = *ipAddr/ ipMask*

Long Syntax: MPOA.152 MPC (*atmIntfNum*): shortcut disassociated from VCC: IP addr/mask = *ipAddr/ ipMask*

Description: The MPC has disassociated a shortcut route from a VCC.

MPOA.153

Level: C-INFO

Short Syntax: MPOA.153 MPC (*atmIntfNum*): updated shortcut: IP addr/mask *ipAddr/ ipMask*: encapsulation type = *encapsString*: MTU = *mtu*: holding time =

holdingTime secs: VCC state = *vccTypeString*: dest ATM addr = x *dstAtmAddr*

Long Syntax: MPOA.153 MPC (*atmIntfNum*): updated shortcut: IP addr/mask *ipAddr/ ipMask*: encapsulation type = *encapsString*: MTU = *mtu*: holding time = *holdingTime* secs: VCC state = *vccTypeString*: dest ATM addr = x *dstAtmAddr*

Description: The MPC has updated a shortcut route based on the contents of a resolution reply for an IP destination.

MPOA.154

Level: UI-ERROR

Short Syntax: MPOA.154 MPC (*atmIntfNum*): IP shortcut=>hold down state: *errString*: IP addr/mask = *ipAddr/ ipMask*

Long Syntax: MPOA.154 MPC (*atmIntfNum*): IP shortcut=>hold down state: *errString*: IP addr/mask = *ipAddr/ ipMask*

Description: The MPC has put an IP shortcut in hold down state.

MPOA.155

Level: UI-ERROR

Short Syntax: MPOA.155 MPC (*atmIntfNum*): VCC cntrl blk alloc failed: remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.155 MPC (*atmIntfNum*): VCC cntrl blk alloc failed: remote ATM addr = x *remoteAtmAddr*

Description: VCC not established because allocation of control block failed.

MPOA.156

Level: C-INFO

Short Syntax: MPOA.156 MPC (*atmIntfNum*): placing call for *typeOfVccString*: remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.156 MPC (*atmIntfNum*): placing call for *typeOfVccString*: remote ATM addr = x *remoteAtmAddr*

Description: The MPC has placed a call to establish an ATM VCC.

MPOA.157

Level: UI-ERROR

Short Syntax: MPOA.157 MPC (*atmIntfNum*): err placing call: *errString*: remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.157 MPC (*atmIntfNum*): err placing call: *errString*: remote ATM addr = x *remoteAtmAddr*

Description: An error occurred when the MPC invoked the ATM interface primitive for placing a call.

MPOA.158

Level: UI-ERROR

Short Syntax: MPOA.158 MPC (*atmIntfNum*) => DOWN: err placing call: *errString*

Long Syntax: MPOA.158 MPC (*atmIntfNum*) => DOWN: err placing call: *errString*

Description: A fatal error occurred when the MPC invoked the ATM interface primitive for placing a call.

MPOA.159

Level: CE-ERROR

Short Syntax: MPOA.159 MPC (*atmIntfNum*): AAL IE: not present or invalid AAL type (x *aalType*)

Long Syntax: MPOA.159 MPC (*atmIntfNum*): AAL IE: not present or invalid AAL type (x *aalType*)

Description: The MPC received a signalling message that was invalid because either it did not contain an AAL IE or the AAL type in the AAL IE was bad.

MPOA.160

Level: CE-ERROR

Short Syntax: MPOA.160 MPC (*atmIntfNum*): AAL IE: invalid *sduTypeString* max SDU size (*sduSize*)

Long Syntax: MPOA.160 MPC (*atmIntfNum*): AAL IE: invalid *sduTypeString* max SDU size (*sduSize*)

Description: The MPC received a signalling message that was invalid because the Max SDU Size in the AAL IE was bad.

MPOA.161

Level: CE-ERROR

Short Syntax: MPOA.161 MPC (*atmIntfNum*): invalid BLLI

Long Syntax: MPOA.161 MPC (*atmIntfNum*): invalid BLLI

Description: The MPC received a signalling message that was invalid because it contained a bad BLLI IE.

MPOA.162

Level: CE-ERROR

Short Syntax: MPOA.162 MPC (*atmIntfNum*): cell rate IE: *errString* (*badVal*)

Long Syntax: MPOA.162 MPC (*atmIntfNum*): cell rate IE: *errString* (*badVal*)

Description: The MPC received a signalling message that was invalid because the Cell Rate IE contained a bad value.

MPOA.163

Level: CE-ERROR

Short Syntax: MPOA.163 MPC (*atmIntfNum*): bearer IE: *errString* (x *badVal*)

Long Syntax: MPOA.163 MPC (*atmIntfNum*): bearer IE: *errString* (x *badVal*)

Description: The MPC received a signalling message that was invalid because the Broadband Bearer IE contained a bad value.

MPOA.164

Level: CE-ERROR

Short Syntax: MPOA.164 MPC (*atmIntfNum*): QoS IE: *errString* (x *badVal*)

Long Syntax: MPOA.164 MPC (*atmIntfNum*): QoS IE: *errString* (x *badVal*)

Description: The MPC received a signalling message that was invalid because the QoS IE contained a bad value.

MPOA.165

Level: CE-ERROR

Short Syntax: MPOA.165 MPC (*atmIntfNum*): calling party IE: *errString*

Long Syntax: MPOA.165 MPC (*atmIntfNum*): calling party IE: *errString*

Description: The MPC received a signalling message that was invalid because of an error with the Calling Party IE.

MPOA.166

Level: CE-ERROR

Short Syntax: MPOA.166 MPC (*atmIntfNum*): releasing placed call: *errString*: remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.166 MPC (*atmIntfNum*): releasing placed call: *errString*: remote ATM addr = x *remoteAtmAddr*

Description: The MPC is releasing a call that it placed due to an error.

MPOA.167

Level: UI-ERROR

Short Syntax: MPOA.167 MPC (*atmIntfNum*) => DOWN: err opening VCC data path: *errString*

Long Syntax: MPOA.167 MPC (*atmIntfNum*) => DOWN: err opening VCC data path: *errString*

Description: A fatal error occurred when the MPC invoked the ATM interface primitive for opening a VCC data path.

MPOA.168

Level: CE-ERROR

Short Syntax: MPOA.168 MPC (*atmIntfNum*): placed call failed: net down: remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.168 MPC (*atmIntfNum*): placed call failed: net down: remote ATM addr = x *remoteAtmAddr*

Description: A call placed by the MPC failed because the signalling interface to the network is down.

MPOA.169

Level: CE-ERROR

Short Syntax: MPOA.169 MPC (*atmIntfNum*): placed call failed: cause # *causeCode*: remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.169 MPC (*atmIntfNum*): placed call failed: cause # *causeCode*: remote ATM addr = x *remoteAtmAddr*

Description: A call placed by the MPC failed with specified cause code.

MPOA.170

Level: C-INFO

Short Syntax: MPOA.170 MPC (*atmIntfNum*): retrying placed call with bearer class C: remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.170 MPC (*atmIntfNum*): retrying placed call with bearer class C: remote ATM addr = x *remoteAtmAddr*

Description: A call placed by the MPC failed due to the Bearer Class, so the call is being retried with Bearer Class C.

MPOA.171

Level: C-INFO

Short Syntax: MPOA.171 MPC (*atmIntfNum*): placed call established: *typeOfVccString*: remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.171 MPC (*atmIntfNum*): placed call established: *typeOfVccString*: remote ATM addr = x *remoteAtmAddr*

Description: A call placed by the MPC has been established.

MPOA.172

Level: CE-ERROR

Short Syntax: MPOA.172 MPC (*atmIntfNum*): rejected received call: no calling party ATM addr

Long Syntax: MPOA.172 MPC (*atmIntfNum*): rejected received call: no calling party ATM addr

Description: The MPC rejected a received call because the calling party ATM address was not provided.

MPOA.173

Level: CE-ERROR

Short Syntax: MPOA.173 MPC (*atmIntfNum*): rejected received call: *errString*: remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.173 MPC (*atmIntfNum*): rejected received call: *errString*: remote ATM addr = x *remoteAtmAddr*

Description: The MPC rejected a received call for the specified reason.

MPOA.174

Level: UI-ERROR

Short Syntax: MPOA.174 MPC (*atmIntfNum*) (*errString*) => DOWN: err acking received call:

Long Syntax: MPOA.174 MPC (*atmIntfNum*) (*errString*) => DOWN: err acking received call:

Description: A fatal error occurred when the MPC invoked the ATM interface primitive for acknowledging a received call.

MPOA.175

Level: C-INFO

Short Syntax: MPOA.175 MPC (*atmIntfNum*): received call: *typeOfVccString*: remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.175 MPC (*atmIntfNum*): received call: *typeOfVccString*: remote ATM addr = x *remoteAtmAddr*

Description: The MPC a received call for the specified type of VCC.

MPOA.176

Level: C-INFO

Short Syntax: MPOA.176 MPC (*atmIntfNum*): call released: normal: *typeOfVccString*: remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.176 MPC (*atmIntfNum*): call released: normal: *typeOfVccString*: remote ATM addr = x *remoteAtmAddr*

Description: A call was released remotely with a normal cause code.

MPOA.177

Level: CE-ERROR

Short Syntax: MPOA.177 MPC (*atmIntfNum*): call released: net down: *typeOfVccString*: remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.177 MPC (*atmIntfNum*): call released: net down: *typeOfVccString*: remote ATM addr = x *remoteAtmAddr*

Description: A call was released because the signalling interface to the network went down.

MPOA.178

Level: CE-ERROR

Short Syntax: MPOA.178 MPC (*atmIntfNum*): call released: cause # *causeCode*: *typeOfVccString*: remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.178 MPC (*atmIntfNum*): call released: cause # *causeCode*: *typeOfVccString*: remote ATM addr = x *remoteAtmAddr*

Description: A call was released remotely with specified cause code.

MPOA.179

Level: C-INFO

Short Syntax: MPOA.179 MPC (*atmIntfNum*): starting VCC timer: *timerTypeString* = *timerValue* secs: remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.179 MPC (*atmIntfNum*): starting VCC timer: *timerTypeString* = *timerValue* secs: remote ATM addr = x *remoteAtmAddr*

Description: A timer associated with the VCC to the specified remote ATM address has been started.

MPOA.180

Level: C-INFO

Short Syntax: MPOA.180 MPC (*atmIntfNum*): VCC DELETED: *reasonString*: remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.180 MPC (*atmIntfNum*): VCC DELETED: *reasonString*: remote ATM addr = x *remoteAtmAddr*

Description: A VCC has been deleted for the specified reason.

MPOA.181

Level: UI-ERROR

Short Syntax: MPOA.181 MPC (*atmIntfNum*): ATM frame buf alloc failed: *errString*

Long Syntax: MPOA.181 MPC (*atmIntfNum*): ATM frame buf alloc failed: *errString*

Description: Allocation of a buffer for an ATM frame failed.

MPOA.182

Level: UI-ERROR

Short Syntax: MPOA.182 MPC (*atmIntfNum*) => DOWN: ATM frame buf alloc failed: *errString*

Long Syntax: MPOA.182 MPC (*atmIntfNum*) => DOWN: ATM frame buf alloc failed: *errString*

Description: Fatal error occured when MPC attempted to allocate a buffer for an ATM frame.

MPOA.183

Level: UI-ERROR

Short Syntax: MPOA.183 MPC (*atmIntfNum*) => DOWN: err reading burned-in MAC addr: *errString*

Long Syntax: MPOA.183 MPC (*atmIntfNum*) => DOWN: err reading burned-in MAC addr: *errString*

Description: Fatal error occured when MPC invoked the ATM interface primitive for reading the burned-in MAC address.

MPOA.184

Level: C-INFO

Short Syntax: MPOA.184 MPC (*atmIntfNum*): opened hardware path for LANE VCC: remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.184 MPC (*atmIntfNum*): opened hardware path for LANE VCC: remote ATM addr = x *remoteAtmAddr*

Description: The MPC opened a path for switching frames received on a LANE VCC in hardware.

MPOA.185

Level: UI-ERROR

Short Syntax: MPOA.185 MPC (*atmIntfNum*): err opening VCC hardware path: *errString* remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.185 MPC (*atmIntfNum*): err opening VCC hardware path: *errString* remote ATM addr = x *remoteAtmAddr*

Description: An error occurred when the MPC attempted to open a path for switching frames received on a LANE VCC in hardware.

MPOA.186

Level: C-INFO

Short Syntax: MPOA.186 Rcvd LANE data frame (vpi/vci *vpi/ vci*)

Long Syntax: MPOA.186 Rcvd LANE data frame (vpi/vci *vpi/ vci*)

Description: A LANE data frame was received by the empc. The vpi/vci of the VCC on which the frame was received are printed.

MPOA.187

Level: U-INFO

Short Syntax: MPOA.187 Unable to send data plane purge rqst (addr *protocolAddress*, rc *errorCode*)

Long Syntax: MPOA.187 Unable to send data plane purge rqst (addr *protocolAddress*, rc *errorCode*)

Description: The empc was unable to send out a data plane purge request message. The protocol address for which the DPP was being sent and an internal error code are printed,

---

MPOA.188

Level: U-INFO

Short Syntax: MPOA.188 LANE pkt recvd from invalid vcc, initiating hold down! (vpi/vci *vpi/ vci*)

Long Syntax: MPOA.188 LANE pkt recvd from invalid vcc, initiating hold down! (vpi/vci *vpi/ vci*)

Description: A packet was received from a LANE vcc though we are not supposed to be receiving on this VCC. (This happens typically with LANE extensions if we think the other end has inadvertently started learning from LANE shortcut packets and is sending us LANE packets). The received packet will be dropped and the ATM address of this VCC will be placed in a hold down state.

MPOA.189

Level: C-INFO

Short Syntax: MPOA.189 Recvd lane ready query (vpi/vci *vpi/ vci*)

Long Syntax: MPOA.189 Recvd lane ready query (vpi/vci *vpi/ vci*)

Description: A LANE Ready-Query message was received on the specified VCC. The MPC will try to respond with a LANE Ready-Ind packet.

MPOA.190

Level: C-INFO

Short Syntax: MPOA.190 Recvd lane encaps NHRP purge rqst (vpi/vci *vpi/ vci*)

Long Syntax: MPOA.190 Recvd lane encaps NHRP purge rqst (vpi/vci *vpi/ vci*)

Description: A NHRP Purge request message was received encapsulated as a LANE control frame on the specified VCC. The MPC will now initiate processing of this request.

MPOA.191

Level: UE-ERROR

Short Syntax: MPOA.191 Recvd invalid lane control frame, dropping! (vpi/vci *vpi/ vci*)

Long Syntax: MPOA.191 Recvd invalid lane control frame, dropping! (vpi/vci *vpi/ vci*)

Description: An invalid LANE control frame was received on the specified VCC The packet will be dropped.

MPOA.192

Level: U-INFO

Short Syntax: MPOA.192 Exhausted dpp retries (src atm addr *entryAtmAddr*)

Long Syntax: MPOA.192 Exhausted dpp retries (src atm addr *entryAtmAddr*)

Description: The empc exhausted the maximum number of retries for sending a MPOA data plane purge request to an ingress device for an entry. (Data Plane Purge Replies are only requested from IBM MPCs). The source ATM address of the entry (address of the ingress) is displayed and the entry will be deleted.

MPOA.193

Level: C-INFO

Short Syntax: MPOA.193 Recvd lane encaps NHRP purge rply ! (vpi/vci *vpi/ vci*)

Long Syntax: MPOA.193 Recvd lane encaps NHRP purge rply ! (vpi/vci *vpi/ vci*)

Description: A NHRP Purge reply message was received encapsulated as a LANE control frame on the specified VCC. The MPC will now initiate processing of this message.

MPOA.194

Level: U-INFO

Short Syntax: MPOA.194 Recvd bad NHRP purge rply ! (vpi/vci *vpi/ vci*)

Long Syntax: MPOA.194 Recvd bad NHRP purge rply ! (vpi/vci *vpi/ vci*)

Description: A NHRP Purge reply message was received on the specified VCC but had some format problems or was missing some needed information. The packet will be dropped.

MPOA.195

Level: U-INFO

Short Syntax: MPOA.195 Sending MPS death dpp rqst! (*destIpAddress/ destIpAddressMask*, id *cacheId*)

Long Syntax: MPOA.195 Sending MPS death dpp rqst! (*destIpAddress/ destIpAddressMask*, id *cacheId*)

Description: A data plane purge request is being sent for an entry because of detection of death of the MPS which imposed it. The protocol address and mask of the entry and the cache ID are displayed.

MPOA.196

Level: U-INFO

Short Syntax: MPOA.196 Sending MPS MAC invalid dpp rqst! (*destIpAddress/ destIpAddressMask*, id *cacheId*)

Long Syntax: MPOA.196 Sending MPS MAC invalid dpp rqst! (*destIpAddress/ destIpAddressMask*, id *cacheId*)

Description: A data plane purge request is being sent for an entry because the corresponding MPS MAC address is no longer valid. The protocol address and mask of the entry and the cache ID are displayed.

MPOA.197

Level: C-INFO

Short Syntax: MPOA.197 Recvd local lane frame (atm intf *atmIntfNum*)

Long Syntax: MPOA.197 Recvd local lane frame (atm intf *atmIntfNum*)

Description: The empc received a locally delivered LANE encapsulated data frame.

MPOA.198

Level: UE-ERROR

Short Syntax: MPOA.198 Recvd bad frame (x *frameBytes*)

Long Syntax: MPOA.198 Recvd bad frame (x *frameBytes*)

Description: The empc received a bad frame (one which could not be decoded correctly). The frame will be dropped.

MPOA.199

Level: UI-ERROR

Short Syntax: MPOA.199 local tagged pkt src atm mismatch (entry: *srcAtmAddrInEntry*)

Long Syntax: MPOA.199 local tagged pkt src atm mismatch (entry: *srcAtmAddrInEntry*)

Description: A locally deliverable tagged frame matched to an entry with the source ATM address different from ours. This could indicate an internal error in which the ingress and egress MPCs are out of synch.

MPOA.200

Level: U-INFO

Short Syntax: MPOA.200 No match for recvd dpp reply (*destIpAddr*/ *destIpAddrMask*)

Long Syntax: MPOA.200 No match for recvd dpp reply (*destIpAddr*/ *destIpAddrMask*)

Description: A Data Plane Purge reply was received (only valid from IBM MPC ingress devices) but did not match to any outstanding entry which is in the process of being purged. (The entry may have got internally) deleted before the purge reply was received).

MPOA.201

Level: UI-ERROR

Short Syntax: MPOA.201 No mem for derived entry (*atmIntfNum*)

Long Syntax: MPOA.201 No mem for derived entry (*atmIntfNum*)

Description: An internal memory allocation (for a ″derived″ type of egress cache entry) failed on receipt of a packet. The packet will be dropped and a Data Plane Purge request initiated. This indicates the box is running low on dynamically allocatable memory.

MPOA.202

Level: U-INFO

Short Syntax: MPOA.202 No match in Nontag 1483 net routes for pkt (*destProtAddr*)

Long Syntax: MPOA.202 No match in Nontag 1483 net routes for pkt (*destProtAddr*)

Description: No matching e-cache entry was found for the destination protocol address in a received packet in the Nontag 1483 Network routes database.

MPOA.203

Level: U-INFO

Short Syntax: MPOA.203 No match in Nontag 1483 host routes for pkt (*destProtAddr*)

Long Syntax: MPOA.203 No match in Nontag 1483 host routes for pkt (*destProtAddr*)

Description: No matching e-cache entry was found for the destination protocol address in a received packet in the Nontag 1483 host routes database.

MPOA.204

Level: C-INFO

Short Syntax: MPOA.204 Function *functionNameString*() called

Long Syntax: MPOA.204 Function *functionNameString*() called

Description: The (internal) function displayed was called (warning: most internal functions do NOT print this ELS, only a few do).

MPOA.205

Level: UE-ERROR

Short Syntax: MPOA.205 Dropping LANE data packet (*atmIntfNum*)

Long Syntax: MPOA.205 Dropping LANE data packet (*atmIntfNum*)

Description: The MPC is discarding a received LANE data packet either because there was no matching egress LEC for it or because the packet was incorrectly prepared. This could indicate an error in the sender of the packet in that it maye either have incorrect LANE header information or is incorrectly preparing the LANE data packet.

MPOA.206

Level: UE-ERROR

Short Syntax: MPOA.206 No matching LEC on imposition (dll: ring 1 x *firstRingNumber* ring 2 x *secondRingNumber* bridge x *bridgeNumber*)

Long Syntax: MPOA.206 No matching LEC on imposition (dll: ring 1 x *firstRingNumber* ring 2 x *secondRingNumber* bridge x *bridgeNumber*)

Description: The DLL hdr information provided in an MPOA Cache Imposition request did not match any ''first ring number, bridge number, second ring number'' triplet that the MPC is currently aware of. This could indicate an error in the DLL header information provided by the imposing e-mps.

MPOA.207

Level: C-INFO

Short Syntax: MPOA.207 Resetting src atm lane ext lec ptr ATM: x *atmAddress*

Long Syntax: MPOA.207 Resetting src atm lane ext lec ptr ATM: x *atmAddress*

Description: The LANE extensions lec egress LEC ptr for the specified source ATM address is being reset. After this, the Vccs from this atm address may use any egress LEC for establishing the hardware switched patch.

MPOA.208

Level: C-INFO

Short Syntax: MPOA.208 Setting src atm lane ext lec ptr ATM: x *atmAddress* Lec:

Long Syntax: MPOA.208 Setting src atm lane ext lec ptr ATM: x *atmAddress* Lec:

Description: The LANE extensions lec egress LEC ptr for the specified source ATM address is being set to point to the displayed LEC. Hardware switched paths can now established on VCCs with this remote ATM address.

MPOA.209

Level: U-INFO

Short Syntax: MPOA.209 MPC Tag array resized to *newSize*

Long Syntax: MPOA.209 MPC Tag array resized to *newSize*

Description: The internal tag array of the eMpc has been resized to the the value indicated. This is done only when the number of tagged cache entries in the egress cache is significantly large.

MPOA.210

Level: U-INFO

Short Syntax: MPOA.210 Bad data pkt dropped w/o dpp (dest: *destIpAddr*, vpi/vci *vpi/ vci*)

Long Syntax: MPOA.210 Bad data pkt dropped w/o dpp (dest: *destIpAddr*, vpi/vci *vpi/ vci*)

Description: A received data packet did not match to any valid egress cache entry but was dropped without sending a Data Plane Purge because a purge rate limit timer was in effect for the remote ATM address of the VCC on which the packet was received.

MPOA.211

Level: C-INFO

Short Syntax: MPOA.211 MPC (*atmIntfNum*): retrying placed call with pcr 25 MBps: remote ATM addr = x *remoteAtmAddr*

Long Syntax: MPOA.211 MPC (*atmIntfNum*): retrying placed call with pcr 25 MBps: remote ATM addr = x *remoteAtmAddr*

Description: A call placed by the MPC failed due to the cell rate being too high, so the call is being retried with a peak cell rate corresponding to 25 MBps.

MPOA.212

Level: UI-ERROR

Short Syntax: MPOA.212 MPC (*atmIntfNum*): shortcut cntrl blk alloc failed: IPX addr = x *ipxAddr*

Long Syntax: MPOA.212 MPC (*atmIntfNum*): shortcut cntrl blk alloc failed: IPX addr = x *ipxAddr*

Description: Allocation of a shortcut control block failed.

MPOA.213

Level: UI-ERROR

Short Syntax: MPOA.213 MPC (*atmIntfNum*): IPX shortcut=>hold down state: *errString*: IPX addr = x *ipxAddrPtr*

Long Syntax: MPOA.213 MPC (*atmIntfNum*): IPX shortcut=>hold down state: *errString*: IPX addr = x *ipxAddrPtr*

Description: The MPC has put an IPX shortcut in hold down state.

MPOA.214

Level: C-INFO

Short Syntax: MPOA.214 MPC (*atmIntfNum*): shortcut disassociated from VCC: IPX addr = x *ipxAddrPtr*

Long Syntax: MPOA.214 MPC (*atmIntfNum*): shortcut disassociated from VCC: IPX addr = x *ipxAddrPtr*

Description: The MPC has disassociated an IPX shortcut route from a VCC.

MPOA.215

Level: C-INFO

Short Syntax: MPOA.215 MPC (*atmIntfNum*): purging IPX shortcut: IPX addr = x *ipxAddrPtr*

Long Syntax: MPOA.215 MPC (*atmIntfNum*): purging IPX shortcut: IPX addr = x *ipxAddrPtr*

Description: The MPC has purged an IPX shortcut.

MPOA.216

Level: C-INFO

Short Syntax: MPOA.216 MPC (*atmIntfNum*): shortcut associated with VCC: IPX addr = x *ipxAddrPtr*

Long Syntax: MPOA.216 MPC (*atmIntfNum*): shortcut associated with VCC: IPX addr = x *ipxAddrPtr*

Description: The MPC has associated an IPX shortcut route with a VCC.

MPOA.217

Level: C-INFO

Short Syntax: MPOA.217 MPC (*atmIntfNum*): initiating resolution req: IPX addr = x *ipxAddrPtr*: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.217 MPC (*atmIntfNum*): initiating resolution req: IPX addr = x *ipxAddrPtr*: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC is initiating transmission of a resolution request for an IPX address.

MPOA.218

Level: C-INFO

Short Syntax: MPOA.218 MPC (*atmIntfNum*): shortcut cntrl blk DELETED: *ipxAddrPtr*: IPX addr = x

Long Syntax: MPOA.218 MPC (*atmIntfNum*): shortcut cntrl blk DELETED: *ipxAddrPtr*: IPX addr = x

Description: The MPC has deleted an IPX shortcut route.

MPOA.219

Level: CE-ERROR

Short Syntax: MPOA.219 MPC (*atmIntfNum*): ignored LANE ext: *errString*: IPX addr = x *ipxAddrPtr*: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.219 MPC (*atmIntfNum*): ignored LANE ext: *errString*: IPX addr = x *ipxAddrPtr*: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC is ignoring invalid LANE extensions in a resolution reply for an IPX address.

MPOA.220

Level: CE-ERROR

Short Syntax: MPOA.220 MPC (*atmIntfNum*): recv resolution NAK: *nakString*: IPX addr = x *ipxAddrPtr*: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.220 MPC (*atmIntfNum*): recv resolution NAK: *nakString*: IPX addr = x *ipxAddrPtr*: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC has received a negative acknowledgement in a resolution reply for an IPX address.

MPOA.221

Level: C-INFO

Short Syntax: MPOA.221 MPC (*atmIntfNum*): recv resolution reply: IPX addr = x *ipxAddrPtr*: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.221 MPC (*atmIntfNum*): recv resolution reply: IPX addr = x *ipxAddrPtr*: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC received a resolution reply for an IPX destination.

MPOA.222

Level: C-INFO

Short Syntax: MPOA.222 MPC (*atmIntfNum*): converting route: IPX addr/prefix x *errString*/ *ipxAddrPtr => prefix1*

Long Syntax: MPOA.222 MPC (*atmIntfNum*): converting route: IPX addr/prefix x *errString*/ *ipxAddrPtr => prefix1*

Description: The MPC is converting an existing route based on the contents of a resolution reply that it received for an IPX address. The conversion may be from host route to network route or vice versa.

MPOA.223

Level: C-INFO

Short Syntax: MPOA.223 MPC (*atmIntfNum*): updated shortcut: IPX addr x *ipxAddrPtr*: encapsulation type = *encapsString*: MTU = *mtu*: holding time = *holdingTime* secs: VCC state = *vccTypeString*: dest ATM addr = x *dstAtmAddr*

Long Syntax: MPOA.223 MPC (*atmIntfNum*): updated shortcut: IPX addr x *ipxAddrPtr*: encapsulation type = *encapsString*: MTU = *mtu*: holding time = *holdingTime* secs: VCC state = *vccTypeString*: dest ATM addr = x *dstAtmAddr*

Description: The MPC has updated a shortcut route

based on the contents of a resolution reply for an IPX destination.

MPOA.224

Level: C-INFO

Short Syntax: MPOA.224 MPC (*atmIntfNum*): recv trigger for IPX addr = x *ipxAddrPtr*: MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.224 MPC (*atmIntfNum*): recv trigger for IPX addr = x *ipxAddrPtr*: MPS ATM addr = x *mpsAtmAddr*

Description: The MPC received a trigger request for an IPX address.

MPOA.225

Level: C-INFO

Short Syntax: MPOA.225 eMPC deleting imposed e-cache entry (IPX networkNo x *ipxNetworkNumber*, cache id x *cacheId*)

Long Syntax: MPOA.225 eMPC deleting imposed e-cache entry (IPX networkNo x *ipxNetworkNumber*, cache id x *cacheId*)

Description: An externally imposed egress MPC cache entry (IPX Network) is being deleted.

MPOA.226

Level: C-INFO

Short Syntax: MPOA.226 eMPC deleting imposed e-cache entry (IPX address x *ipxAddress*, cache id x *cacheId*)

Long Syntax: MPOA.226 eMPC deleting imposed e-cache entry (IPX address x *ipxAddress*, cache id x *cacheId*)

Description: An externally imposed egress MPC cache entry (possibly network) is being deleted.

MPOA.227

Level: C-INFO

Short Syntax: MPOA.227 eMPC deleting intern. derived e-cache entry (IPX address x *ipxAddress*, cache id x *cacheId*)

Long Syntax: MPOA.227 eMPC deleting intern. derived e-cache entry (IPX address x *ipxAddress*, cache id x *cacheId*)

Description: An externally imposed egress MPC cache entry (possibly network) is being deleted.

MPOA.228

Level: U-INFO

Short Syntax: MPOA.228 Sending MPS death dpp rqst! (*destIpxAddress*, id *cacheId*)

Long Syntax: MPOA.228 Sending MPS death dpp rqst! (*destIpxAddress*, id *cacheId*)

Description: A data plane purge request is being sent for an entry because of detection of death of the MPS which imposed it. The protocol address and and the cache ID are displayed.

MPOA.229

Level: U-INFO

Short Syntax: MPOA.229 Bad data pkt dropped w/o dpp (dest: *destIpAxddr*, vpi/vci *vpi/ vci*)

Long Syntax: MPOA.229 Bad data pkt dropped w/o dpp (dest: *destIpAxddr*, vpi/vci *vpi/ vci*)

Description: A received data packet did not match to any valid egress cache entry but was dropped without sending a Data Plane Purge because a purge rate limit timer was in effect for the remote ATM address of the VCC on which the packet was received.

MPOA.230

Level: U-INFO

Short Syntax: MPOA.230 MPC (*atmIntfNum*) IP STOPPED

Long Syntax: MPOA.230 MPC (*atmIntfNum*) IP STOPPED

Description: IP Operation of MPC instance is being stopped.

MPOA.231

Level: U-INFO

Short Syntax: MPOA.231 MPC (*atmIntfNum*) IPX STOPPED

Long Syntax: MPOA.231 MPC (*atmIntfNum*) IPX STOPPED

Description: IPX Operation of MPC instance is being stopped.

MPOA.232

Level: U-INFO

Short Syntax: MPOA.232 No MPC protocol sram record. Will use defaults! (*netNum*)

Long Syntax: MPOA.232 No MPC protocol sram record. Will use defaults! (*netNum*)

Description: No SRAM protocol configuration record

was found for the MPOA client. The client will hence come up with a default set of parameters.

MPOA.233

Level: UE-ERROR

Short Syntax: MPOA.233 Mismatched MPC protocol sram record. Will use defaults! (*netNum*)

Long Syntax: MPOA.233 Mismatched MPC protocol sram record. Will use defaults! (*netNum*)

Description: An SRAM protocol configuration record was found for the MPOA client but for a different ATM interface than the one coming up. The existing SRAM protocol record configuration parameters will hence be ignored and the client will hence come up with a default set of parameters. This could indicate a misconfiguration.

MPOA.234

Level: U-INFO

Short Syntax: MPOA.234 Bad recvd pkt: sending DPP rqst! (*destIpxAddressPtr*)

Long Syntax: MPOA.234 Bad recvd pkt: sending DPP rqst! (*destIpxAddressPtr*)

Description: A Data Plane Purge Request is being sent because of a received IPX packet for which no matching egress cache entry was found. The destination protocol address in the pkt is printed.

MPOA.235

Level: U-INFO

Short Syntax: MPOA.235 Purge Reply recvd for disabled protocol! (IP addr/mask = *IpAddress/ IpAddressMask*, vpi/vci *vpi/ vci*)

Long Syntax: MPOA.235 Purge Reply recvd for disabled protocol! (IP addr/mask = *IpAddress/ IpAddressMask*, vpi/vci *vpi/ vci*)

Description: A NHRP Purge reply message was received on the specified VCC but the specified protocol has been disabled and all entries deleted. The packet will be dropped.

MPOA.236

Level: U-INFO

Short Syntax: MPOA.236 Purge Reply recvd for disabled protocol! (IPX addr = *ipxAddrPtr*, Prefix = *ipxPrefix*, vpi/vci *vpi/ vci*)

Long Syntax: MPOA.236 Purge Reply recvd for disabled protocol! (IPX addr = *ipxAddrPtr*, Prefix = *ipxPrefix*, vpi/vci *vpi/ vci*)

Description: A NHRP Purge reply message was received on the specified VCC but the specified protocol has been disabled and all entries deleted. The packet will be dropped.

MPOA.237

Level: U-INFO

Short Syntax: MPOA.237 No match for recvd IPX dpp reply (*ipxAddrPtr/ ipxPrefix*)

Long Syntax: MPOA.237 No match for recvd IPX dpp reply (*ipxAddrPtr/ ipxPrefix*)

Description: A Data Plane Purge reply was received (only valid from IBM MPC ingress devices) but did not match to any outstanding IPX entry which is in the process of being purged. (The entry may have got internally) deleted before the purge reply was received).

MPOA.238

Level: UE-ERROR

Short Syntax: MPOA.238 Destn prot addr mismatch in recvd pkt! (dstn *dstnIpxAddrPtr*, entry *entryIpxAddrPtr*, Prefix = *entryIpxPrefix*)

Long Syntax: MPOA.238 Destn prot addr mismatch in recvd pkt! (dstn *dstnIpxAddrPtr*, entry *entryIpxAddrPtr*, Prefix = *entryIpxPrefix*)

Description: The tag based lookup for a received MPOA tagged packet matched to an e-cache entry for a different IPX destination protocol address range than the destination protocol address in the packet. The IPX destn address in the packet, and the address/mask combination of the e-cache entry are printed.

MPOA.239

Level: U-INFO

Short Syntax: MPOA.239 Tagged pkt: matching entry inactive! (entry *ipxAddrPtr*, Prefix = *ipxPrefix*, state *entryState*)

Long Syntax: MPOA.239 Tagged pkt: matching entry inactive! (entry *ipxAddrPtr*, Prefix = *ipxPrefix*, state *entryState*)

Description: The tag based lookup for a received MPOA tagged packet matched to an e-cache entry which was not in an active state. The IPX protocol address and prefix length of the matching entry are printed along with the state of the entry.

MPOA.240

Level: C-INFO

Short Syntax: MPOA.240 Recvd 1483 IPX data pkt! (*ipxAddrPtr*)

Long Syntax: MPOA.240 Recvd 1483 IPX data pkt! (*ipxAddrPtr*)

Description: An MPOA Nontagged 1483 IPX packet was received. The destination IPX address in the pkt is printed.

MPOA.241

Level: C-INFO

Short Syntax: MPOA.241 1483 pkt hash cache miss! (*ipxAddrPtr*)

Long Syntax: MPOA.241 1483 pkt hash cache miss! (*ipxAddrPtr*)

Description: A received MPOA Nontagged 1483 IPX packet caused a hash array miss. The destination IPX address in the pkt is printed.

MPOA.242

Level: U-INFO

Short Syntax: MPOA.242 1483 pkt: matching entry inactive! (entry *ipxAddrPtr*, Prefix = *ipxPrefix*, state *entryState*)

Long Syntax: MPOA.242 1483 pkt: matching entry inactive! (entry *ipxAddrPtr*, Prefix = *ipxPrefix*, state *entryState*)

Description: The tag based lookup for a received MPOA non-tagged 1483 packet matched to an e-cache entry which was not in an active state. The IPX protocol address and prefix of the matching entry are printed alongwith the state of the entry.

MPOA.243

Level: U-INFO

Short Syntax: MPOA.243 Unable to send data plane purge rqst (addr *ipxAddrPtr*, rc *errorCode*)

Long Syntax: MPOA.243 Unable to send data plane purge rqst (addr *ipxAddrPtr*, rc *errorCode*)

Description: The empc was unable to send out a data plane purge request message. The IPX protocol address for which the DPP was being sent and an internal error code are printed.

MPOA.244

Level: U-INFO

Short Syntax: MPOA.244 No match in Nontag 1483 host routes for IPX pkt (*ipxAddrPtr*)

Long Syntax: MPOA.244 No match in Nontag 1483 host routes for IPX pkt (*ipxAddrPtr*)

Description: No matching e-cache entry was found for the destination IPX protocol address in a received packet in the Nontag 1483 host routes database.

MPOA.245

Level: U-INFO

Short Syntax: MPOA.245 No match in Nontag 1483 net routes for IPX pkt (*ipxAddrPtr*)

Long Syntax: MPOA.245 No match in Nontag 1483 net routes for IPX pkt (*ipxAddrPtr*)

Description: No matching e-cache entry was found for the destination IPX protocol address in a received packet in the Nontag 1483 Network routes database.

MPOA.246

Level: C-INFO

Short Syntax: MPOA.246 Imposn rqst valid (*ipxAddrPtr*, Prefix: *ipxPrefix*, rqst ID: x *requestID*)

Long Syntax: MPOA.246 Imposition request valid (*ipxAddrPtr*, Prefix: *ipxPrefix*, rqst ID: x *requestID*)

Description: A valid cache imposition request has been received for the specified destination IPX address and prefix. The request ID in the message is also printed.

MPOA.247

Level: C-INFO

Short Syntax: MPOA.247 Processing imposn rqst for new ntry (*ipxAddrPtr*, Prefix: *ipxPrefix*, x *cacheID*, *entryTypeString*)

Long Syntax: MPOA.247 Processing imposition request for a new entry (*ipxAddrPtr*, Prefix: *ipxPrefix*, x *cacheID*, *entryTypeString*)

Description: An MPOA Cache imposition request for a new egress cache entry is being processed. The destination IPX protocol address, prefix length, cache ID and type of entry are displayed.

MPOA.248

Level: C-INFO

Short Syntax: MPOA.248 New egress cache IPX entry created (*ipxAddrPtr*, Prefix: *ipxPrefix*, x *cacheID*, *entryTypeString*)

Long Syntax: MPOA.248 New egress cache IPX entry created (*ipxAddrPtr*, Prefix: *ipxPrefix*, x *cacheID*, *entryTypeString*)

Description: A new egress IPX cache entry has been created for the specified IPX protocol address and prefix. The cacheID and the type of the entry are also printed.

MPOA.249

Level: C-INFO

Short Syntax: MPOA.249 Imposn rfrsh for exstng IPX ntry recvd (*ipxAddrPtr*, Prefix: *ipxPrefix*, x *cacheID*, *entryTypeString*)

Long Syntax: MPOA.249 Imposition refresh for existing IPX entry received (*ipxAddrPtr*, Prefix: *ipxPrefix*, x *cacheID*, *entryTypeString*)

Description: An MPOC Cache imposition request was received to refresh an existing IPX entry in the egress cache. The destination IPX protocol address, prefix, cache ID and type of entry are displayed.

MPOA.250

Level: UE-ERROR

Short Syntax: MPOA.250 Recvd imposn rfrsh for purging ntry (*ipxAddrPtr*, Prefix: *ipxPrefix*)

Long Syntax: MPOA.250 Recvd imposn rfrsh for purging ntry (*ipxAddrPtr*, Prefix: *ipxPrefix*)

Description: An imposition request refresh was received for an IPX entry which was in the process of being purged. Indicates a likely error in the E-MPS logic.

MPOA.251

Level: UE-ERROR

Short Syntax: MPOA.251 Recvd imposn rqst, IPX network mismatch (old *oldNetwork*/ new *newNetwork*)

Long Syntax: MPOA.251 Recvd imposn rqst, IPX network mismatch (old *oldNetwork*/ new *newNetwork*)

Description: An imposition request received for an existing e-cache entry had a different destination IPX network than the existing one. The existing entry will internally be deleted and a new one created. Indicates a likely error in the E-MPS logic.

MPOA.252

Level: UE-ERROR

Short Syntax: MPOA.252 Recvd imposn rqst, IPX address mismatch (old *ipxAddrPtr1*/ new *ipxAddrPtr2*)

Long Syntax: MPOA.252 Recvd imposn rqst, IPX address mismatch (old *ipxAddrPtr1*/ new *ipxAddrPtr2*)

Description: An imposition request received for an existing e-cache entry had a different destination IPX address than the existing one. The existing entry will internally be deleted and a new one created. Indicates a likely error in the E-MPS logic.

MPOA.253

Level: UE-ERROR

Short Syntax: MPOA.253 Recvd IPX imposn rqst had src atm addr mismatch (*ipxAddrPtr*, Prefix: *ipxPrefix*)

Long Syntax: MPOA.253 Recvd IPX imposn rqst had src atm addr mismatch (*ipxAddrPtr*, Prefix: *ipxPrefix*)

Description: An IPX imposition request received for an existing e-cache entry had a different source ATM address than the existing one. The existing entry will internally be deleted and a new one created. Indicates a likely error in the E-MPS logic.

MPOA.254

Level: U-INFO

Short Syntax: MPOA.254 Recvd IPX imposn rqst had dll mismatch!(*ipxAddrPtr*, Prefix: *ipxPrefix*)

Long Syntax: MPOA.254 Recvd IPX imposn rqst had dll mismatch!(*ipxAddrPtr*, Prefix: *ipxPrefix*)

Description: An IPX imposition request received for an existing e-cache entry had a different DLL than the one provided on the previous request. The MPC will internally delete the old entry and create a new one with the new information.

MPOA.255

Level: C-INFO

Short Syntax: MPOA.255 Egress IPX cache entry refreshed (*ipxAddrPtr*, Prefix: *ipxPrefix*, x *cacheID*, *entryTypeString*)

Long Syntax: MPOA.255 Egress IPX cache entry refreshed (*ipxAddrPtr*, Prefix: *ipxPrefix*, x *cacheID*, *entryTypeString*)

Description: An existing egress IPX cache entry has been refreshed for the specified protocol address and prefix pair. The cacheID and the type of the entry are also printed.

MPOA.256

Level: C-INFO

Short Syntax: MPOA.256 IPX prtcl addr based e-mps purge recvd (*ipxAddrPtr*, Prefix: *ipxPrefix*)

Long Syntax: MPOA.256 IPX prtcl addr based e-mps purge recvd (*ipxAddrPtr*, Prefix: *ipxPrefix*)

Description: An e-mps purge was received for a range of IPX protocol addresses. This range is specified by the protocol address and prefix (the range could also indicate only one address).

MPOA.257

Level: U-INFO

Short Syntax: MPOA.257 MPC (*atmIntfNum*) IP STARTED

Long Syntax: MPOA.257 MPC (*atmIntfNum*) IP STARTED

Description: IP Operation of MPC instance is being started.

MPOA.258

Level: U-INFO

Short Syntax: MPOA.258 MPC (*atmIntfNum*) IPX STARTED

Long Syntax: MPOA.258 MPC (*atmIntfNum*) IPX STARTED

Description: IPX Operation of MPC instance is being started.

MPOA.259

Level: U-INFO

Short Syntax: MPOA.259 Sending MPS MAC invalid IPX dpp rqst! (*ipxAddrPtr*, Prefix: *ipxPrefix*, id *cacheId*)

Long Syntax: MPOA.259 Sending MPS MAC invalid IPX dpp rqst! (*ipxAddrPtr*, Prefix: *ipxPrefix*, id *cacheId*)

Description: A data plane purge request is being sent for an entry because the corresponding MPS MAC address is no longer valid. The IPX protocol address and prefix of the entry and the cache ID are displayed.

MPOA.260

Level: U-INFO

Short Syntax: MPOA.260 MPC (*atmIntfNum*): recv ' *enable_disable_string* IPX' config TLV from LECS: LEC (*lecIntfNum*)

Long Syntax: MPOA.260 MPC (*atmIntfNum*): recv ' *enable_disable_string* IPX' config TLV from LECS: LEC (*lecIntfNum*)

Description: The MPC has received a configuration TLV from the LECS to control whether IPX flow detection is enabled or disabled.

MPOA.261

Level: UI-ERROR

Short Syntax: MPOA.261 MPC => DOWN: FDDL user reg failed: x *errString*

Long Syntax: MPOA.261 MPC => DOWN: FDDL user reg failed: x *errString*

Description: Registration as user of FDDL interface failed. MPC going down as result.

MPOA.262

Level: UI-ERROR

Short Syntax: MPOA.262 MPC => DOWN: FDDL user reg failed: x *errString*

Long Syntax: MPOA.262 MPC => DOWN: FDDL user reg failed: x *errString*

Description: Configuration set request of FDDL interface failed. MPC going down as result.

MPOA.263

Level: U-INFO

Short Syntax: MPOA.263 MPC (*atmIntfNum*): FDDL add failed MPS MAC addr (x *macAddr*): LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Long Syntax: MPOA.263 MPC (*atmIntfNum*): refreshed MPS MAC Addr (x *macAddr*): LEC (*lecIntfNum*): MPS ATM addr = x *mpsAtmAddr*

Description: The association of a MAC address with a MPS has not been refeshed.

MPOA.264

Level: UI-ERROR

Short Syntax: MPOA.264 MPC: anchor tag cb init failed

Long Syntax: MPOA.264 MPC: anchor tag control block initialization failed.

Description: MPC: anchor tag control block initialization failed.

MPOA.265

Level: U-INFO

Short Syntax: MPOA.265 MPC: FDDL MPS MAC ADDR Update cycle successful

Long Syntax: MPOA.265 MPC: FDDL MPS MAC ADDR Update cycle successful

Description: MPC: FDDL MPS MAC ADDR Update cycle successful

MPOA.266

Level: UI-ERROR

Short Syntax: MPOA.266 MPC (*atmIntfNum*) Egress Cache Tag (*tagValue*) Deletion from FDDL Failed

Long Syntax: MPOA.266 MPC (*atmIntfNum*) Egress Cache Tag (*tagValue*) Deletion from FDDL Failed

Description: MPC instance could not delete the egress cache tag from FDDL.

MPOA.267

Level: UI-ERROR

Short Syntax: MPOA.267 MPC (*atmIntfNum*) Egress Cache Tag (*tagValue*) Addition to FDDL Failed

Long Syntax: MPOA.267 MPC (*atmIntfNum*) Egress Cache Tag (*tagValue*) Addition to FDDL Failed

Description: MPC instance could not add the egress cache tag to FDDL.

MPOA.268

Level: UE-ERROR

Short Syntax: MPOA.268 Recvd imposn rqst had VEGA Blade/Box mismatch (*ipAddress*/ *ipAddressMask*)

Long Syntax: MPOA.268 Recvd imposn rqst had VEGA Blade/Boxmismatch (*ipAddress*/ *ipAddressMask*)

Description: An imposition request received for an existing e-cache entry had a different Blade/Box attribute than the existing one. The existing entry will internally be deleted and a new one created.

MPOA.269

Level: U-INFO

Short Syntax: MPOA.269 Recvd IPX imposn rqst had box/blade mismatch!(*ipxAddrPtr*, Prefix: *ipxPrefix*)

Long Syntax: MPOA.269 Recvd IPX imposn rqst had box/blade mismatch!(*ipxAddrPtr*, Prefix: *ipxPrefix*)

Description: An IPX imposition request received for an existing e-cache entry had a different box/blade attribute than the one provided on the previous request. The MPC will internally delete the old entry and create a new one with the new information.

MPOA.270

Level: C-INFO

Short Syntax: MPOA.270 MPC: FDDL MPS MAC ADDR Update cycle called

Long Syntax: MPOA.270 MPC: FDDL MPS MAC ADDR Update cycle called

Description: MPC: FDDL MPS MAC ADDR Update cycle called

MPOA.271

Level: C-INFO

Short Syntax: MPOA.271 MPC (*atmIntfNum*) Egress Cache Tag (*tagValue*) Deletion from FDDL Called

Long Syntax: MPOA.271 MPC (*atmIntfNum*) Egress Cache Tag (*tagValue*) Deletion from FDDL Called

Description: MPC:Egress Cache Tag Deletion from FDDL Called

MPOA.272

Level: C-INFO

Short Syntax: MPOA.272 MPC (*atmIntfNum*) Egress Cache Tag (*tagValue*) Addition to FDDL Called

Long Syntax: MPOA.272 MPC (*atmIntfNum*) Egress Cache Tag (*tagValue*) Addition to FDDL Called

Description: MPC: Egress cache tag addition to FDDL Called

MPOA.273

Level: C-INFO

Short Syntax: MPOA.273 MPC (*atmIntfNum*): MPS ID (*mpsId*) MPS MAC addr (x *macAddr*) Add. to FDDL Called

Long Syntax: MPOA.273 MPC (*atmIntfNum*): MPS ID (*mpsId*) MPS MAC addr (x *macAddr*) Addition to FDDL Called

Description: MPC: MPS MAC address addition to FDDL Called

MPOA.274

Level: U-INFO

Short Syntax: MPOA.274 MPC (*atmIntfNum*): MPS ID (*mpsId*) MPS MAC addr (x *macAddr*) Add. to FDDL Failed

Long Syntax: MPOA.274 MPC (*atmIntfNum*): MPS ID (*mpsId*) MPS MAC addr (x *macAddr*) Addition to FDDL Failed

Description: MPC: MPS MAC address addition to FDDL Failed

MPOA.275

Level: C-INFO

Short Syntax: MPOA.275 MPC (*atmIntfNum*): MPS ID (*mpsId*) MPS MAC addr (x *macAddr*) Del. from FDDL Called

Long Syntax: MPOA.275 MPC (*atmIntfNum*): MPS ID (*mpsId*) MPS MAC addr (x *macAddr*) Deletion from FDDL Called

Description: MPC: MPS MAC address deletion from FDDL Called

MPOA.276

Level: U-INFO

Short Syntax: MPOA.276 MPC (*atmIntfNum*): MPS ID (*mpsId*) MPS MAC addr (x *macAddr*) Del. from FDDL Failed.

Long Syntax: MPOA.276 MPC (*atmIntfNum*): MPS ID (*mpsId*) MPS MAC addr (x *macAddr*) Deletion from FDDL Failed.

Description: MPC: MPS MAC address addition to FDDL Failed

MPOA.277

Level: U-INFO

Short Syntax: MPOA.277 MPC (*atmIntfNum*) Init Failed. Max. possible MPC instances exceeded

Long Syntax: MPOA.277 MPC (*atmIntfNum*) Init Failed. Max. possible MPC instances exceeded

Description: MPC instance could not ne initialized because maximum possible MPC instances are already running.

MPOA.278

Level: C-INFO

Short Syntax: MPOA.278 MPC (*atmIntfNum*): SRAM update performed (Addition of MPC_PROTOCOL SRAM)

Long Syntax: MPOA.278 MPC (*atmIntfNum*): SRAM update performed (Addition of MPC_PROTOCOL SRAM)

Description: Only Common SRAM record was present. So added the protocol SRAM record to the config.

MPOA.279

Level: UI-ERROR

Short Syntax: MPOA.279 MPOA: LEC assoc. with MPC instance failed, no MPC on device: (*atmIntfNum*)

Long Syntax: MPOA.279 MPOA: LEC assoc. with MPC instance failed, no MPC on device: (*atmIntfNum*)

Description: When the LEC came up, the lecUp message says it is running on some ATM interface over which there is no MPC running.

MPOA.280

Level: UI-ERROR

Short Syntax: MPOA.280 MPC (*atmIntfNum*): Loc. shortcut to (*ipAddress*) is untagged. FDDL Ing cache Add not done

Long Syntax: MPOA.280 MPC (*atmIntfNum*): Loc. shortcut to (*ipAddress*) is untagged. FDDL Ing cache Add not done

Description: Local shortcut should only be of the type ″tagged″. But in this case it is not. So fddl call to add ingress cache entry is not made.

MPOA.281

Level: UI-ERROR

Short Syntax: MPOA.281 MPC (*atmIntfNum*): Loc. shortcut to (*ipAddress*) has bad tag (*tag*). FDDL Ing cache Add not done

Long Syntax: MPOA.281 MPC (*atmIntfNum*): Loc. shortcut to (*ipAddress*) is bad tag (*tag*). FDDL Ing cache Add not done

Description: Local shortcut has invalid tag. So fddl call to add ingress cache entry is not made.

MPOA.282

Level: UI-ERROR

Short Syntax: MPOA.282 MPC (*atmIntfNum*): FDDL LAN dest (*macAddress*) to interface failed.

Long Syntax: MPOA.282 MPC (*atmIntfNum*): FDDL LAN dest (*macAddress*) to interface failed.

Description: The fddl call that looks for the legacy interface at which the given mac address is present did not succeed.

MPOA.283

Level: C-INFO

Short Syntax: MPOA.283 MPC (*atmIntfNum*): Flow Indication for MPS (*mpsId*) received.

Long Syntax: MPOA.283 MPC (*atmIntfNum*): Flow Indication for the MPS (*mpsId*) received at CPU.

Description: A data frame was bridged to the MPS and an indication is sent to the CPU by the EPIF.

MPOA.284

Level: C-INFO

Short Syntax: MPOA.284 MPOA: MPC(*atmIntfNum*): FDDL Delete Ingress Cache Entry. IP addr = *ipAddr*, prefix len = *prefixLength* int = *intf*

Long Syntax: MPOA.284 MPOA: MPC(*atmIntfNum*): FDDL Delete Ingress Cache Entry. IP addr = *ipAddr*, prefix len = *prefixLength* from interface *intf*

Description: An ingress cache entry is being deleted from the EPIF memory at the given interface.

MPOA.285

Level: C-INFO

Short Syntax: MPOA.285 MPOA: MPC(*atmIntfNum*): FDDL Add Ingress Cache Entry. IP addr = *ipAddr*, prefix len = *prefixLength* int = *intf*

Long Syntax: MPOA.285 MPOA: MPC(*atmIntfNum*): FDDL Add Ingress Cache Entry. IP addr = *ipAddr*, prefix len = *prefixLength* from interface *intf*

Description: An ingress cache entry is being added to the EPIF memory at the given interface.

MPOA.286

Level: C-INFO

Short Syntax: MPOA.286 MPC (*atmIntfNum*): MPS ID (*mpsId*) Addition to FDDL Called

Long Syntax: MPOA.286 MPC (*atmIntfNum*): MPS ID (*mpsId*) Addition to FDDL Called

Description: MPC: MPS ID addition to FDDL Called

MPOA.287

Level: U-INFO

Short Syntax: MPOA.287 MPC (*atmIntfNum*): MPS ID (*mpsId*) Addition to FDDL Failed!

Long Syntax: MPOA.287 MPC (*atmIntfNum*): MPS ID (*mpsId*) Addition to FDDL Failed!

Description: MPC: MPS ID addition to FDDL Failed

MPOA.288

Level: C-INFO

Short Syntax: MPOA.288 MPC (*atmIntfNum*): MPS ID (*mpsId*) Deletion from FDDL Called

Long Syntax: MPOA.288 MPC (*atmIntfNum*): MPS ID (*mpsId*) Deletion from FDDL Called

Description: MPC: MPS ID deletion from FDDL Called

MPOA.289

Level: U-INFO

Short Syntax: MPOA.289 MPC (*atmIntfNum*): MPS ID (*mpsId*) MPS MAC addr (x) Del. from FDDL Failed.

Long Syntax: MPOA.289 MPC (*atmIntfNum*): MPS ID (*mpsId*) MPS MAC addr (x) Deletion from FDDL Failed.

Description: MPC: MPS ID deletion from FDDL Failed

MPOA.290

Level: C-INFO

Short Syntax: MPOA.290 MPC (*atmIntfNum*): Flow Indication for IP destination (*ipAddr*) is processed.

Long Syntax: MPOA.290 MPC (*atmIntfNum*): Flow Detection Indication for IP destination (*ipAddr*) is processed.

Description: MPC: A frame to the given IP destination is bridged to an MPS MAC address.

MPOA.291

Level: C-INFO

Short Syntax: MPOA.291 MPC (*atmIntfNum*): Flow Indication for IPX destination (x *ipxAddr*) is processed.

Long Syntax: MPOA.291 MPC (*atmIntfNum*): Flow Detection Indication for IPX destination (x *ipxAddr*) is processed.

Description: MPC: A frame to the given IPX destination is bridged to an MPS MAC address.

MPOA.292

Level: U-INFO

Short Syntax: MPOA.292 MPC (*atmIntfNum*) Init Aborted.EPIF runtime image is not MPOA capable

Long Syntax: MPOA.292 MPC (*atmIntfNum*) Init Aborted.EPIF runtime image is not MPOA capable

Description: MPC instance could not ne initialized because EPIF runtime image is not MPOA capable

---

MPOA.293

Level: U-INFO

Short Syntax: MPOA.293 MPC (*atmIntfNum*) Init Aborted because of lack of SRAM record

Long Syntax: MPOA.293 MPC (*atmIntfNum*) Init Aborted because of lack of SRAM record

Description: MPC instance could not ne initialized because there was no SRAM for it. Currently existence of SRAM record is a requirement for initialization only for 8371.

MPOA.294

Level: U-INFO

Short Syntax: MPOA.294 MPC (*atmIntfNum*) Init Aborted because of SRAM record being in diabled state

Long Syntax: MPOA.294 MPC (*atmIntfNum*) Init Aborted because of SRAM record being in diabled state

Description: MPC instance could not ne initialized

because the corresponding SRAM record is in disabled state.

MPOA.295

Level: U-INFO

Short Syntax: MPOA.295 MPC (*atmIntfNum*) Init Aborted because of bridge SRAM record is disabled/absent

Long Syntax: MPOA.295 MPC (*atmIntfNum*) Init Aborted because of bridge SRAM record is disabled/absent

Description: MPC instance could not ne initialized because the bridge SRAM record is in disabled/absent

---

MPOA.296

Level: U-INFO

Short Syntax: MPOA.296 MPC (*atmIntfNum*) Init Aborted because of SELF SRAM record is enabled

Long Syntax: MPOA.296 MPC (*atmIntfNum*) Init Aborted because of SELF SRAM record is enabled

Description: MPC instance could not ne initialized because the SELF SRAM record is enabled.

MPOA.297

Level: C-INFO

Short Syntax: MPOA.297 MPC (*atmIntfNum*): Imposition Requestion for IP Host (*ipAddr*) is processed.

Long Syntax: MPOA.297 MPC (*atmIntfNum*): Imposition Requestion for IP Host (*ipAddr*) is processed.

Description: MPC: The Impostion request is targeted for an internal IP address.

Chapter 81. NetBIOS Support Subsystem (NBS)

This chapter describes NetBIOS Support Subsystem (NBS) messages. For information on message content and how to use the message, refer to the Introduction.

NBS.001

Level: C-INFO

Short Syntax: NBS.001 *instance_str*NetBIOS Add_Name_Query received from bridge for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.001 *instance_str*NetBIOS Add_Name_Query received from bridge for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Add_Name_Query frame from the bridged network.

NBS.002

Level: C-INFO

Short Syntax: NBS.002 *instance_str*NetBIOS Add_Group_Name_Query received from bridge for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.002 *instance_str*NetBIOS Add_Group_Name_Query received from bridge for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Add_Group_Name_Query frame from the bridged network.

NBS.003

Level: C-INFO

Short Syntax: NBS.003 *instance_str*NetBIOS Add_Name_Response received from bridge for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.003 *instance_str*NetBIOS Add_Name_Response received from bridge for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Add_Name_Response frame from the bridged network.

NBS.004

Level: C-INFO

Short Syntax: NBS.004 *instance_str*NetBIOS Name_Query received from bridge for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.004 *instance_str*NetBIOS Name_Query received from bridge for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Name_Query frame from the bridged network.

NBS.005

Level: C-INFO

Short Syntax: NBS.005 *instance_str*NetBIOS Name_Recognized received from bridge for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.005 *instance_str*NetBIOS Name_Recognized received from bridge for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Name_Recognized frame from the bridged network.

NBS.006

Level: C-INFO

Short Syntax: NBS.006 *instance_str*NetBIOS Name_In_Conflict received from bridge for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.006 *instance_str*NetBIOS Name_In_Conflict received from bridge for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Name_In_Conflict frame from the bridged network.

NBS.007

Level: C-INFO

Short Syntax: NBS.007 *instance_str*NetBIOS Status_Query received from bridge for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.007 *instance_str*NetBIOS Status_Query received from bridge for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Status_Query frame from the bridged network.

NBS.008

Level: C-INFO

Short Syntax: NBS.008 *instance_str*NetBIOS Status_Response received from bridge for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.008 *instance_str*NetBIOS Status_Response received from bridge for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Status_Response frame from the bridged network.

NBS.009

Level: C-INFO

Short Syntax: NBS.009 *instance_str*NetBIOS Datagram received from bridge for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.009 *instance_str*NetBIOS Datagram received from bridge for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Datagram frame from the bridged network.

NBS.010

Level: C-INFO

Short Syntax: NBS.010 *instance_str*NetBIOS Datagram_Broadcast received from bridge for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.010 *instance_str*NetBIOS Datagram_Broadcast received from bridge for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Datagram_Broadcast frame from the bridged network.

NBS.011

Level: C-INFO

Short Syntax: NBS.011 *instance_str*NetBIOS Terminate_Trace_07 received from bridge for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.011 *instance_str*NetBIOS Terminate_Trace_07 received from bridge for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Terminate_Trace_07 frame from the bridged network.

NBS.012

Level: C-INFO

Short Syntax: NBS.012 *instance_str*NetBIOS Terminate_Trace_13 received from bridge for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.012 *instance_str*NetBIOS Terminate_Trace_13 received from bridge for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Terminate_Trace_13 frame from the bridged network.

NBS.013

Level: C-INFO

Short Syntax: NBS.013 *instance_str*Unrecognized NetBIOS frame received from bridge for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.013 *instance_str*Unrecognized NetBIOS frame received from bridge for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received an unrecognized NetBIOS frame from the bridged network.

NBS.014

Level: C-INFO

Short Syntax: NBS.014 *instance_str*NetBIOS Add_Name_Query received from dlsw for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.014 *instance_str*NetBIOS

Add_Name_Query received from dlsw for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Add_Name_Query frame from the DLSw network.

NBS.015

Level: C-INFO

Short Syntax: NBS.015 *instance_str*NetBIOS Add_Group_Name_Query received from dlsw for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.015 *instance_str*NetBIOS Add_Group_Name_Query received from dlsw for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Add_Group_Name_Query frame from the DLSw network.

NBS.016

Level: C-INFO

Short Syntax: NBS.016 *instance_str*NetBIOS Add_Name_Response received from dlsw for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.016 *instance_str*NetBIOS Add_Name_Response received from dlsw for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Add_Name_Response frame from the DLSw network.

NBS.017

Level: C-INFO

Short Syntax: NBS.017 *instance_str*NetBIOS Name_Query received from dlsw for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.017 *instance_str*NetBIOS Name_Query received from dlsw for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Name_Query frame from the DLSw network.

NBS.018

Level: C-INFO

Short Syntax: NBS.018 *instance_str*NetBIOS Name_Recognized received from dlsw for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.018 *instance_str*NetBIOS Name_Recognized received from dlsw for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Name_Recognized frame from the DLSw network.

NBS.019

Level: C-INFO

Short Syntax: NBS.019 *instance_str*NetBIOS Name_In_Conflict received from dlsw for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.019 *instance_str*NetBIOS Name_In_Conflict received from dlsw for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Name_In_Conflict frame from the DLSw network.

NBS.020

Level: C-INFO

Short Syntax: NBS.020 *instance_str*NetBIOS Status_Query received from dlsw for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.020 *instance_str*NetBIOS Status_Query received from dlsw for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Status_Query frame from the DLSw network.

NBS.021

Level: C-INFO

Short Syntax: NBS.021 *instance_str*NetBIOS Status_Response received from dlsw for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.021 *instance_str*NetBIOS Status_Response received from dlsw for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a

NetBIOS Status_Response frame from the DLSw network.

NBS.022

Level: C-INFO

Short Syntax: NBS.022 *instance_str*NetBIOS Datagram received from dlsw for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.022 *instance_str*NetBIOS Datagram received from dlsw for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Datagram frame from the DLSw network.

NBS.023

Level: C-INFO

Short Syntax: NBS.023 *instance_str*NetBIOS Datagram_Broadcast received from dlsw for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.023 *instance_str*NetBIOS Datagram_Broadcast received from dlsw for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Datagram_Broadcast frame from the DLSw network.

NBS.024

Level: C-INFO

Short Syntax: NBS.024 *instance_str*NetBIOS Terminate_Trace_07 received from dlsw for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.024 *instance_str*NetBIOS Terminate_Trace_07 received from dlsw for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Terminate_Trace_07 frame from the DLSw network.

NBS.025

Level: C-INFO

Short Syntax: NBS.025 *instance_str*NetBIOS Terminate_Trace_13 received from dlsw for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.025 *instance_str*NetBIOS Terminate_Trace_13 received from dlsw for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received a NetBIOS Terminate_Trace_13 frame from the DLSw network.

NBS.026

Level: C-INFO

Short Syntax: NBS.026 *instance_str*Unrecognized NetBIOS frame received from dlsw for *source_nbname*(*source_macaddr*)-> *dest_nbname*(*dest_macaddr*)

Long Syntax: NBS.026 *instance_str*Unrecognized NetBIOS frame received from dlsw for source name(MAC) *source_nbname*(*source_macaddr*) -> target name(MAC) *dest_nbname*(*dest_macaddr*)

Description: The NetBIOS software received an unrecognized NetBIOS frame from the DLSw network.

NBS.027

Level: C-INFO

Short Syntax: NBS.027 *instance_str*NetBIOS frame for *source_nbname*-> *dest_nbname* not forwarded to bridge - frame type filter

Long Syntax: NBS.027 *instance_str*NetBIOS frame for source name *source_nbname* -> dest name *dest_nbname* not forwarded to bridge due to frame type filter

Description: The NetBIOS software bridge frame type filter did not forward the given NetBIOS frame to the bridged network. It was filtered by the NetBIOS support bridge frame type filter.

NBS.028

Level: C-INFO

Short Syntax: NBS.028 *instance_str*NetBIOS frame for *source_nbname*-> *dest_nbname* not forwarded to dlsw - frame type filter

Long Syntax: NBS.028 *instance_str*NetBIOS frame for source name *source_nbname* -> dest name *dest_nbname* not forwarded to DLSw due to frame type filter

Description: The router did not forward the given NetBIOS frame to the DLSw network because it was filtered by the NetBIOS support DLSw frame type filter.

NBS.029

Level: C-INFO

Short Syntax: NBS.029 *instance_str*NetBIOS frame for *source_nbname*-> *dest_nbname* not forwarded to bridge - no name cache entry

Long Syntax: NBS.029 *instance_str*NetBIOS frame for source name *source_nbname* -> dest name *dest_nbname*

not forwarded to bridge due to no matching name cache entry created

Description: The router did not forward the given NetBIOS frame to the bridged network because it could not find or create a corresponding name cache entry.

NBS.030

Level: C-INFO

Short Syntax: NBS.030 *instance_str*NetBIOS frame for *source_nbname*-> *dest_nbname* not forwarded to dlsw - no name cache entry

Long Syntax: NBS.030 *instance_str*NetBIOS frame for source name *source_nbname* -> dest name *dest_nbname* not forwarded to dlsw due to no matching name cache entry created

Description: The router did not forward the given NetBIOS frame to the DLSw network because it could not find or create a corresponding name cache entry.

NBS.031

Level: C-INFO

Short Syntax: NBS.031 *instance_str*NetBIOS frame for *source_nbname*-> *dest_nbname* not forwarded to bridge - command processing

Long Syntax: NBS.031 *instance_str*NetBIOS frame for source name *source_nbname* -> dest name *dest_nbname* not forwarded to bridge due to duplicate command frame processing

Description: The router did not forward the given NetBIOS frame to the bridged network because the router filtered it as a duplicate NetBIOS command frame.

NBS.032

Level: C-INFO

Short Syntax: NBS.032 *instance_str*NetBIOS frame for *source_nbname*-> *dest_nbname* not forwarded to dlsw - command processing

Long Syntax: NBS.032 *instance_str*NetBIOS frame for source name *source_nbname* -> dest name *dest_nbname* not forwarded to dlsw due to duplicate command frame processing

Description: The router did not forward the given NetBIOS frame to the DLSw network because the router filtered it as a duplicate NetBIOS command frame.

NBS.033

Level: C-INFO

Short Syntax: NBS.033 *instance_str*NetBIOS frame for *source_nbname*-> *dest_nbname* not forwarded to bridge - response processing

Long Syntax: NBS.033 *instance_str*NetBIOS frame for source name *source_nbname* -> dest name *dest_nbname* not forwarded to bridge due to no command matching this response

Description: The router did not forward the given NetBIOS frame to the bridged network because the router could not find a command frame matching this response frame.

NBS.034

Level: C-INFO

Short Syntax: NBS.034 *instance_str*NetBIOS frame for *source_nbname*-> *dest_nbname* not forwarded to dlsw - response processing

Long Syntax: NBS.034 *instance_str*NetBIOS frame for source name *source_nbname* -> dest name *dest_nbname* not forwarded to dlsw due to no command matching this response

Description: The router did not forward the given NetBIOS frame to the DLSw network because the router could not find a command frame matching this response frame.

NBS.035

Level: C-INFO

Short Syntax: NBS.035 *instance_str*NetBIOS frame for *source_nbname*-> *dest_nbname* not forwarded to bridge - checking cache

Long Syntax: NBS.035 *instance_str*NetBIOS frame for source name *source_nbname* -> dest name *dest_nbname* not forwarded to bridge due to name cache processing checks

Description: The router did not forward the given NetBIOS frame to the bridged network because the name cache processing indicated the router should not forward it.

NBS.036

Level: C-INFO

Short Syntax: NBS.036 *instance_str*NetBIOS frame for *source_nbname*-> *dest_nbname* not forwarded to dlsw - checking cache

Long Syntax: NBS.036 *instance_str*NetBIOS frame for source name *source_nbname* -> dest name *dest_nbname* not forwarded to dlsw due to name cache processing checks

Description: The router did not forward the given NetBIOS frame to the DLSw network because the name cache processing indicated the router should not forward it.

NBS.037

Level: C-INFO

Short Syntax: NBS.037 *instance_str*NetBIOS frame for *source_nbname*-> *dest_nbname* not forwarded to bridge - checking other

Long Syntax: NBS.037 *instance_str*NetBIOS frame for source name *source_nbname* -> dest name *dest_nbname* not forwarded to bridge due to other processing checks

Description: The router did not forward the given NetBIOS frame to the bridged network because the processing indicated the router should not forward it.

NBS.038

Level: C-INFO

Short Syntax: NBS.038 *instance_str*NetBIOS frame for *source_nbname*-> *dest_nbname* not forwarded to dlsw - checking other

Long Syntax: NBS.038 *instance_str*NetBIOS frame for source name *source_nbname* -> dest name *dest_nbname* not forwarded to dlsw due to other processing checks

Description: The router did not forward the given NetBIOS frame to the DLSw network because processing indicated the router should not forward it.

NBS.039

Level: C-INFO

Short Syntax: NBS.039 *instance_str*Learning new NetBIOS name / MAC and RIF assoc for *source_nbname* to *source_macaddr*/ *rif*

Long Syntax: NBS.039 *instance_str*Learning new NetBIOS name to MAC address and RIF association for NetBIOSname *source_nbname* to MAC *source_macaddr* / RIF *rif*

Description: The NetBIOS software is associating a MAC address and RIF with a NetBIOS name. Find this association on NetBIOS Name_Querys, Name_Recognizeds, and Datagrams.

NBS.040

Level: C-INFO

Short Syntax: NBS.040 *instance_str*NetBIOS frame for *dest_nbname* modified with new MAC (*dest_macaddr*) and RIF (*rif*)

Long Syntax: NBS.040 *instance_str*NetBIOS frame for destination name *dest_nbname* was modified with the new MAC (*dest_macaddr*) and RIF (*rif*)

Description: The router modified a NetBIOS frame to be forwarded to the bridged network by using the cached MAC address and routing information (if available). This modification takes place on NetBIOS Name_Querys, Status_Querys, and Datagrams.

NBS.041

Level: C-INFO

Short Syntax: NBS.041 *instance_str*NetBIOS name cache entry created for *nbname*

Long Syntax: NBS.041 *instance_str*NetBIOS name cache entry created for NetBIOS name *nbname*

Description: The routed created a new NetBIOS name cache entry. This typically occurs on NetBIOS Name_Querys, Status_Querys, Add_Name_Querys, Add_Group_Name_Querys, and Datagrams.

NBS.042

Level: C-INFO

Short Syntax: NBS.042 *instance_str*NetBIOS command/response entry created for *nbname*

Long Syntax: NBS.042 *instance_str*NetBIOS command/response entry created for NetBIOS name *nbname*

Description: The router created a new NetBIOS command/response entry. This typically occurs on NetBIOS Name_Querys, Status_Querys, and Datagrams.

NBS.043

Level: UE-ERROR

Short Syntax: NBS.043 *instance_str*NetBIOS name cache entry invalid (reason *reason*) for *nbname*

Long Syntax: NBS.043 *instance_str*NetBIOS name cache entry validation error occurred (reason *reason*) for NetBIOS name *nbname*

Description: A validation of the name cache entry indicated that the entry is invalid. That is, certain fields contain invalid values or invalid combinations of values. The possible reason codes are as follows: 01 - bad nlist_search / name_type combination; 02 - bad entry_type / name_type combination; 03 - bad name_type value.

NBS.044

Level: C-INFO

Short Syntax: NBS.044 *instance_str*NetBIOS name cache entry deleted for *nbname*

Long Syntax: NBS.044 *instance_str*NetBIOS name cache entry deleted for NetBIOS name *nbname*

Description: The router deleted a NetBIOS name cache entry. This typically occurs as a result of it aging out.

NBS.045

Level: C-INFO

Short Syntax: NBS.045 *instance_str*NetBIOS Support component is active

Long Syntax: NBS.045 *instance_str*NetBIOS Support component is active

Description: The NetBIOS software has now been activated and initialized.

NBS.046

Level: UI-ERROR

Short Syntax: NBS.046 no mem to alloc NB flt

Long Syntax: NBS.046 No memory to allocate a NETBIOS Filter

Description: The router will not enable at least one configured NetBIOS filter, because there is not enough memory.

Cause: Insufficient free memory.

Action: Increase memory size.

NBS.047

Level: U-INFO

Short Syntax: NBS.047 *input_output* NB flt lst, port *port_number*, dlted

Long Syntax: NBS.047 *input_output* NETBIOS filter list, for port *port_number*, deleted by user. Filter will not be enabled

Description: You deleted a filter list, that was part of an already configured filter. You cannot enable the filter.

Cause: User configuration error.

Action: Reconfigure the filter list that was deleted.

NBS.048

Level: U-INFO

Short Syntax: NBS.048 *input_output* NB flt configd for port *port_number*, port doesnt exist

Long Syntax: NBS.048 *input_output* NETBIOS filter for port *port_number* is configured, but that port number is not configured

Description: You configured a NetBIOS filter for a particular port, but that port number is not configured.

Cause: User configuration error.

Action: Either reconfigure the NetBIOS filter for the correct port number, or add to the SRT configuration the port number that you configured in the NETBIOS filter.

NBS.049

Level: C-TRACE

Short Syntax: NBS.049 NB outp pkt fltd *source_mac*-> *dest_mac*, prt *port*, nt *network*

Long Syntax: NBS.049 NETBIOS Output Packet Filtered - *source_mac*-> *dest_mac* , port *port*, network *network*

Description: A NetBIOS packet has matched the criteria the router specified in a NetBIOS filter configuration record. The packet is dropped.

NBS.050

Level: UI-ERROR

Short Syntax: NBS.050 no mem to alloc NB cnsl info

Long Syntax: NBS.050 No memory to allocate information for NETBIOS Filter console display

Description: The part of the router that handles NetBIOS console display cannot allocate enough memory to do the complete display from the T 5 process.

Cause: Insufficient free memory.

Action: Increase memory size.

Chapter 82. Network Address Translation (NAT)

This chapter describes Network Address Translation (NAT) messages. For information on message content and how to use the message, refer to the Introduction.

NAT.001

Level: P-TRACE

Short Syntax: NAT.001 *source_ip_address* -> *destination_ip_address* - Prot= *protocol* Flg=x *ip_flags* Dir= *direction*

Long Syntax: NAT.001 Translating IP packet from *source_ip_address* for *destination_ip_address*; protocol: *protocol* flags: x *ip_flags* flow: *direction*

Description: Trace point for IP packets before being translated by NAT.

NAT.002

Level: P-TRACE

Short Syntax: NAT.002 *source_ip_address* -> *destination_ip_address* - Status= *status*

Long Syntax: NAT.002 Translated IP packet from *source_ip_address* for *destination_ip_address*; packet status: *status*

Description: Trace point for IP packets after being translated by NAT.

NAT.003

Level: P-TRACE

Short Syntax: NAT.003 *source_ip_address* -> *destination_ip_address* - ICMP Type= *icmp_type*,Code= *icmp_code*

Long Syntax: NAT.003 Translating ICMP packet from *source_ip_address* for *destination_ip_address*; ICMP Type *icmp_type* - Code *icmp_code*

Description: Trace point for ICMP packets before being translated by NAT.

NAT.004

Level: P-TRACE

Short Syntax: NAT.004 *source_ip_address*/ *source_udp_port* -> *destination_ip_address*/ *destination_udp_port* - UDP

Long Syntax: NAT.004 Translating UDP packet from *source_ip_address*/port *source_udp_port* for *destination_ip_address*/port *destination_udp_port*

Description: Trace point for UDP packets before being translated by NAT.

NAT.005

Level: P-TRACE

Short Syntax: NAT.005 *source_ip_address*/ *source_tcp_port* -> *destination_ip_address*/ *destination_tcp_port* - TCP

Long Syntax: NAT.005 Translating TCP packet from *source_ip_address*/port *source_tcp_port* for *destination_ip_address*/port *destination_tcp_port*

Description: Trace point for TCP packets before being translated by NAT.

NAT.006

Level: P-TRACE

Short Syntax: NAT.006 *source_ip_address*/ *source_port* -> *destination_ip_address*/ *destination_port* - *protocol* (x *specific_data*x)

Long Syntax: NAT.006 Translating packet from *source_ip_address*/port *source_port* for *destination_ip_address*/port *destination_port*, protocol *protocol*, data x *specific_data*x

Description: Trace point for higher layer protocol packets before being translated by NAT. For FTP, the specific data is the current data delta from any previous translation. For DNS, the specific data is the number of RRs (in the upper word), and the flags fields from the DNS Header (in the lower word).

NAT.007

Level: UI-ERROR

Short Syntax: NAT.007 *source_ip_address*/ *source_tcp_port* -> *destination_ip_address*/ *destination_tcp_port* - No mem for TCP entry

Long Syntax: NAT.007 No memory available to create new TCP session entry from *source_ip_address*/port *source_tcp_port* for *destination_ip_address*/port *destination_tcp_port*

Description: No memory is available to create a new TCP session entry for NAT.

Cause: Not enough memory to support this configuration.

Action: Upgrade for more memory, or reduce configuration.

NAT.008

Level: UI-ERROR

Short Syntax: NAT.008 Private= *private_ip_address* Public= *public_ip_address* - TCP entry not deleted

Long Syntax: NAT.008 Given TCP entry (PrivateIP= *private_ip_address*, PublicIP= *public_ip_address*) not found and deleted from TCP session list

Description: A given TCP entry was not found and could not be deleted from a list of active TCP sessions being monitored by NAT.

Cause: Internal NAT error.

Action: Check NAT configuration. If valid, contact customer service.

NAT.009

Level: UI-ERROR

Short Syntax: NAT.009 *source_ip_address*/*source_ftp_port* -> *destination_ip_address*/*destination_ftp_port* - FTP/TCP not tracked

Long Syntax: NAT.009 Active FTP session from *source_ip_address*/port *source_ftp_port* for *destination_ip_address*/port *destination_ftp_port* not being monitored by NAT

Description: An active FTP session is not being monitored by NAT. NAT should monitor all active TCP sessions when NAPT is in use.

Cause: Internal NAT error.

Action: Check NAT configuration. If valid, contact customer service.

NAT.010

Level: UE-ERROR

Short Syntax: NAT.010 *source_ip_address*/*source_ftp_port* -> *destination_ip_address*/*destination_ftp_port* - FTP <host-port> trans fail - state= *tcp_state*

Long Syntax: NAT.010 Translation of FTP <host-port> string failed for session from *source_ip_address*/port *source_ftp_port* for *destination_ip_address*/port *destination_ftp_port* - TCP State *tcp_state*

Description: An invalid <host-port> string was encountered by NAT in an FTP PORT or PASV command.

Cause: A misbehaving FTP application or corrupted FTP data received in an IP packet.

Action: Check network and FTP application integrity.

NAT.011

Level: UE-ERROR

Short Syntax: NAT.011 Bad FTP <host-port> string: *ftp_host_port_string*

Long Syntax: NAT.011 NAT tried to translate an invalid FTP <host-port> string: *ftp_host_port_string*

Description: An invalid <host-port> string was encountered by NAT in an FTP PORT or PASV command.

Cause: A misbehaving FTP application or corrupted FTP data received in an IP packet.

Action: Check network and FTP application integrity.

NAT.012

Level: UI-ERROR

Short Syntax: NAT.012 *base_ip_address- range_mask* - No mem for NAT range entry

Long Syntax: NAT.012 No memory available to create new Translate Range entry for Base Address *base_ip_address* Mask *range_mask*

Description: No memory is available to create a new Translate Range entry for NAT.

Cause: Not enough memory to support this configuration.

Action: Upgrade for more memory, or reduce configuration.

NAT.013

Level: UE-ERROR

Short Syntax: NAT.013 Bad Reserve Pool: *pool_name*, *starting_ip_address*, *pool_mask*, *pool_size*: *error_msg*

Long Syntax: NAT.013 NAT Reserve Pool *pool_name* misconfigured: StartAddr= *starting_ip_address* Mask= *pool_mask* Size= *pool_size*: *error_msg*

Description: An invalid value for a reserve pool was encountered during NAT initialization.

Cause: Either no pool size, no pool mask, or a duplicate reserved address was configured for a NAT Reserve Pool.

Action: Correct the NAT configuration.

NAT.014

Level: UE-ERROR

Short Syntax: NAT.014 Multiple NaptAddr ReservePool: *pool_name*, *napt_ip_address*, *napt_ip_address*

Long Syntax: NAT.014 NAT Reserve Pool *pool_name* configured with multiple NAPT addresses: NaptAddr= *napt_ip_address* NewNaptAddr= *napt_ip_address*

Description: Multiple NAPT addresses for a single reserve pool were encountered during NAT initialization.

Cause: Multiple NAPT addresses were configured for NAT. Only one is allowed.

Action: Correct the NAT configuration.

NAT.015

Level: UI-ERROR

Short Syntax: NAT.015 *pool_name* - No mem for Reserve Pool entry

Long Syntax: NAT.015 No memory available to create new Reserve Pool entry for Pool *pool_name*

Description: No memory is available to create a new Reserve Pool entry for NAT.

Cause: Not enough memory to support this configuration.

Action: Upgrade for more memory, or reduce configuration.

NAT.016

Level: UI-ERROR

Short Syntax: NAT.016 *pool_name* - Reserve Pool entry not deleted

Long Syntax: NAT.016 Given Reserve Pool *pool_name* not found and deleted from Reserve Pool list

Description: A given reserve pool was not found and could not be deleted from a list of reserve pools being kept by NAT.

Cause: Internal NAT error.

Action: Check NAT configuration. If valid, contact customer service.

NAT.017

Level: UE-ERROR

Short Syntax: NAT.017 *pool_name* <- *base_ip_address* - Assoc Reserve Pool not found for NAT range

Long Syntax: NAT.017 Associated Reserve Pool *pool_name* not found for configured Translate Range *base_ip_address*

Description: A translate range was encountered during NAT initialization that does not have an existing associated reserve pool.

Cause: A range of IP addresses eligible for NAT was configured with an associated reserve pool that does not exist.

Action: Correct the NAT configuration.

NAT.018

Level: CI-ERROR

Short Syntax: NAT.018 *ip_address* - not removed from Reserve Pool

Long Syntax: NAT.018 Given IP Address *ip_address* not found and removed from Reserve Pool list

Description: A given public IP address was not found and could not be deleted from a list of available reserved IP addresses being kept by NAT.

Cause: Internal NAT error.

Action: Check NAT configuration. If valid, contact customer service.

NAT.019

Level: CI-ERROR

Short Syntax: NAT.019 *source_ip_address* -> *destination_ip_address* - NAT not enabled

Long Syntax: NAT.019 Request to translate IP packet from *source_ip_address* for *destination_ip_address*, but NAT not enabled

Description: A possible NAT configuration error has caused NAT to not become enabled. Thus, although access controls has indicated to translate packet, no translation performed.

Cause: Possible NAT configuration error.

Action: Recheck the NAT config and correct any errors/inconsistencies.

NAT.020

Level: U-INFO

Short Syntax: NAT.020 Private= *private_ip_address* - no BasicNAT addr avail for non-TCP/UDP pkt

Long Syntax: NAT.020 Private IP Address *private_ip_address* trying to use Basic NAT for a non-TCP/UDP packet, but none available

Description: A non-TCP/UDP session initiated from the private network tried to use NAT but no Basic NAT public IP addresses were available for use.

Cause: No public IP addresses were available for use by NAT.

Action: Configure more public IP addresses for NAT to use.

NAT.021

Level: UI-ERROR

Short Syntax: NAT.021 Private= *private_ip_address* - no Assoc Reserve Pool for NAPT

Long Syntax: NAT.021 Private IP Address

private_ip_address trying to use NAPT but does not have an associated Reserve Pool

Description: A session initiated from the private network was mapped by NAT to use a NAPT address but no associated reserve pool was given for use.

Cause: Internal NAT error.

Action: Check NAT configuration. If valid, contact customer service.

NAT.022

Level: U-INFO

Short Syntax: NAT.022 Private= *private_ip_address* - no BasicNAT/NAPT addr avail for use

Long Syntax: NAT.022 Private IP Address *private_ip_address* trying to use Basic NAT or NAPT, but none available

Description: An IP packet received from the private network tried to use NAT but no Basic NAT or NAPT public IP addresses were available for use.

Cause: No configured public IP addresses were available for use by NAT.

Action: Configure more public IP addresses for NAT to use.

NAT.023

Level: UI-ERROR

Short Syntax: NAT.023 *direction*/Hash= *hash_value* - no 1st entry to remove

Long Syntax: NAT.023 No NAT entry to remove from the *direction* NAT Table at position *hash_value*

Description: A given NAT entry was not found and could not be deleted from a list of entries being kept by NAT.

Cause: Internal NAT error.

Action: Check NAT configuration. If valid, contact customer service.

NAT.024

Level: UE-ERROR

Short Syntax: NAT.024 Private: *private_ip_address*/ *private_port*, *private_pool_name*<->Public: *public_ip_address*/ *public_port*, *public_pool_name* - static map failed: *reason*

Long Syntax: NAT.024 Static mapping (Private: *private_ip_address*/ *private_port*,pool= *private_pool_name*<->Public: *public_ip_address*/ *public_port*,pool= *public_pool_name*) failed: *reason*

Description: A static mapping was encountered during NAT initialization that failed. The mapping may conflict with the configured reserve pools associated with the private IP address being mapped. Or one of the given addresses may already be bound.

Cause: Invalid static mapping was configured for NAT.

Action: Correct the NAT configuration.

NAT.025

Level: C-INFO

Short Syntax: NAT.025 proxy ARP rsp for NAT: *ip_address*

Long Syntax: NAT.025 Proxy ARP for NAT is responding to an ARP request for IP address *ip_address*

Description: The Proxy ARP for NAT is responding to an ARP request for an IP address that NAT has reserved.

NAT.026

Level: UI-ERROR

Short Syntax: NAT.026 *ip_address*/ *port_number* - duplicate key added *direction*

Long Syntax: NAT.026 A duplicate key (IpAddr= *ip_address*/Port= *port_number*) has been added to the *direction* NAT Table

Description: A duplicate NAT entry was added to a list of entries being kept by NAT.

Cause: Internal NAT error.

Action: Check NAT configuration. If valid, contact customer service.

NAT.027

Level: UI-ERROR

Short Syntax: NAT.027 *ip_address*/ *port_number* - entry not removed *direction*

Long Syntax: NAT.027 Given NAT entry (IpAddr= *ip_address*/Port= *port_number*) not removed from the *direction* NAT Table

Description: A given NAT entry was not found and could not be deleted from a list of entries being kept by NAT.

Cause: Internal NAT error.

Action: Check NAT configuration. If valid, contact customer service.

NAT.028

Level: UI-ERROR

Short Syntax: NAT.028 Private: *private_ip_address*/

private_port Public: *public_ip_address*/ *public_port* - no mem for entry

Long Syntax: NAT.028 No memory available to create new NAT entry for Private *private_ip_address*/port *private_port* - Public *public_ip_address*/port *public_port*

Description: No memory is available to create a new translation entry for NAT.

Cause: Not enough memory to support this configuration/traffic load.

Action: Upgrade for more memory, or reduce configuration/traffic.

NAT.029

Level: UI-ERROR

Short Syntax: NAT.029 No NAT entry given to unbind

Long Syntax: NAT.029 No NAT entry given to removed from the NAT Table

Description: No NAT entry was given to delete from the NAT Table.

Cause: Internal NAT error.

Action: Check NAT configuration. If valid, contact customer service.

NAT.030

Level: UI-ERROR

Short Syntax: NAT.030 *ip_address*/ *port_number* - STATIC entry being unbound

Long Syntax: NAT.030 Statically defined entry (*ip_address*/Port= *port_number*) is trying to get unbound

Description: A static mapping is trying to be unbound by NAT.

Cause: Internal NAT error.

Action: Check NAT configuration. If valid, contact customer service.

NAT.031

Level: C-INFO

Short Syntax: NAT.031 *ip_address*/ *port_number* - unbound *direction*

Long Syntax: NAT.031 Given NAT entry (IpAddr= *ip_address*/Port= *port_number*) removed from the *direction* NAT Table

Description: A given NAT entry was found and deleted from a list of entries being kept by NAT.

NAT.032

Level: UI-ERROR

Short Syntax: NAT.032 NAT init failed

Long Syntax: NAT.032 Initialization of NAT failed

Description: A failure occurred while NAT was initializing its internal data structures.

Cause: Configuration error or internal NAT processing error.

Action: Check the ELS messages that appear prior to this one to get a better indication of why initialization failed.

NAT.033

Level: UI-ERROR

Short Syntax: NAT.033 Return of NAT mem failed

Long Syntax: NAT.033 Returning of all NAT memory failed

Description: A failure occurred while NAT was returning its memory back to the system.

Cause: Internal NAT processing error.

Action: Check the ELS messages that appear prior to this one to get a better indication of why the return of memory failed.

NAT.034

Level: UI-ERROR

Short Syntax: NAT.034 No mem for frag chain *fragment_id*: *source_ip_address* -> *destination_ip_address*

Long Syntax: NAT.034 No memory available to track new IP fragment chain *fragment_id* from *source_ip_address* for *destination_ip_address*

Description: No memory is available to create a new IP fragment entry for NAT.

Cause: Not enough memory to support this configuration.

Action: Upgrade for more memory, or reduce configuration.

NAT.035

Level: U-INFO

Short Syntax: NAT.035 1st frag lost in chain *fragment_id*: *source_ip_address* -> *destination_ip_address*

Long Syntax: NAT.035 The first IP fragment packet was assumed to be lost for chain *fragment_id* from *source_ip_address* for *destination_ip_address*

Description: The first IP fragment packet in a chain of fragments was lost.

Cause: Packets are being lost in the network.

Action: Check network performance and look for areas of congestion.

NAT.036

Level: UI-ERROR

Short Syntax: NAT.036 *source_ip_address* -> *destination_ip_address* - Cannot trans pkt using frag *fragment_chain_key*

Long Syntax: NAT.036 Cannot translate packet from *source_ip_address* to *destination_ip_address* using tracked IP fragment for *fragment_chain_key*

Description: NAT is unable to translate an IP packet using info saved from a tracked fragment chain.

Cause: Internal NAT error.

Action: Check NAT configuration. If valid, contact customer service.

NAT.037

Level: C-INFO

Short Syntax: NAT.037 track frag chain *fragment_id*/ *source_ip_address* -> *destination_ip_address*

Long Syntax: NAT.037 NAT is tracking fragment chain *fragment_id* from *source_ip_address* to *destination_ip_address*

Description: NAT is starting to track a fragment chain.

NAT.038

Level: C-INFO

Short Syntax: NAT.038 saving frag *fragment_id*/ *source_ip_address* -> *destination_ip_address*

Long Syntax: NAT.038 NAT is saving a fragment *fragment_id* from *source_ip_address* to *destination_ip_address*

Description: NAT is unable to process the current fragment due to missing information from the first fragment. NAT is saving the current fragment waiting for the first fragment to arrive.

NAT.039

Level: C-INFO

Short Syntax: NAT.039 freeing saved frag *fragment_id*/ *ip_address* saved port *saved_port*

Long Syntax: NAT.039 NAT is freeing saved fragment *fragment_id* from *ip_address* saved port *saved_port*

Description: NAT is freeing a saved IP fragment.

NAT.040

Level: C-INFO

Short Syntax: NAT.040 process saved frag *fragment_id*/ *ip_address* saved port *saved_port*

Long Syntax: NAT.040 NAT is processing saved fragment *fragment_id* from *ip_address* saved port *saved_port*

Description: NAT is processing a saved IP fragment.

NAT.041

Level: C-INFO

Short Syntax: NAT.041 stopped tracking frag chain *fragment_id*/ *ip_address* saved port *saved_port*

Long Syntax: NAT.041 NAT is stopping tracking of fragment chain *fragment_id*: *ip_address* saved port *saved_port*

Description: NAT is no longer tracking IP fragment chain.

NAT.042

Level: C-INFO

Short Syntax: NAT.042 *source_ip_address* -> *destination_ip_address*: modified IP option x *ip_option*

Long Syntax: NAT.042 NAT modified packet going from *source_ip_address* to *destination_ip_address* with IP option x *ip_option*

Description: NAT has modified the IP addresses contained within an IP header option field.

NAT.043

Level: U-INFO

Short Syntax: NAT.043 *source_ip_address* -> *destination_ip_address*: dropped because of IP option x *ip_option*

Long Syntax: NAT.043 Packet from *source_ip_address* to *destination_ip_address* was dropped by NAT because of IP option x *ip_option*

Description: An IP packet with IP options tried to use NAT but no Basic NAT public IP addresses were available for use.

Cause: No public IP addresses were available for use by NAT.

Action: Configure more public IP addresses for NAT to use.

NAT.044

Level: C-INFO

Short Syntax: NAT.044 *source_ip_address*/ *source_port* -> *destination_ip_address*/ *destination_port*: delta: *data_delta*, modified FTP data: *ftp_data*

Long Syntax: NAT.044 NAT modified FTP data going from *source_ip_address*/ *source_port* to *destination_ip_address*/ *destination_port* with delta *data_delta*, FTP data *ftp_data*

Description: NAT has modified FTP data within a TCP packet.

NAT.045

Level: UI-ERROR

Short Syntax: NAT.045 dup 1st pkts in frag chain *fragment_id*(*saved_port*): *source_ip_address* -> *destination_ip_address*

Long Syntax: NAT.045 NAT has received duplicate 1st packets in fragment chain *fragment_id* (saved port *saved_port*) from *source_ip_address* to *destination_ip_address*

Description: NAT has received duplicate 1st packets in a fragment chain.

Cause: Packets getting corrupted in the network.

Action: Check network and network devices.

NAT.046

Level: C-INFO

Short Syntax: NAT.046 *description*: *private_ip_address* to *public_ip_address*

Long Syntax: NAT.046 NAT modified data: *description*: from *private_ip_address* to *public_ip_address*

Description: NAT has translated IP addresses within the IP data.

NAT.047

Level: C-INFO

Short Syntax: NAT.047 Private= *private_ip_address* Public= *public_ip_address* - TCP entry deleted

Long Syntax: NAT.047 Given TCP entry (PrivateIP= *private_ip_address*, PublicIP= *public_ip_address*) deleted from TCP session list

Description: A given TCP entry was found and deleted from a list of active TCP sessions being monitored by NAT.

NAT.048

Level: C-INFO

Short Syntax: NAT.048 NAT dropping packet: *reason*

Long Syntax: NAT.048 NAT dropping packet: *reason*

Description: NAT is dropping IP packet for stated reason.

Chapter 83. Network Dispatcher Router (NDR)

This chapter describes Network Dispatcher Router (NDR) messages. For information on message content and how to use the message, refer to the Introduction.

NDR.001

Level: P-TRACE

Short Syntax: NDR.001 rcv *source_ip_address* -> *destination_ip_address*

Long Syntax: NDR.001 Receiving packet from *source_ip_address* for *destination_ip_address*

Description: This message is generated for each packet which has passed first-level reasonableness checks.

NDR.002

Level: C-TRACE

Short Syntax: NDR.002 frg pkt *source_ip_address* -> *destination_ip_address*

Long Syntax: NDR.002 Packet from *source_ip_address* for *destination_ip_address* requires fragmentation

Description: This message is generated when an IP packet needs to be fragmented for transmission.

NDR.003

Level: UE-ERROR

Short Syntax: NDR.003 LL broadcast *source_ip_address* -> *destination_ip_address*, discarded

Long Syntax: NDR.003 Received link level broadcast from *source_ip_address* for *destination_ip_address*, discarded

Description: This message is generated when an attempt is made to forward an IP packet that was received as a link level broadcast/multicast. Such packets are not forwarded, and are discarded without even sending back an ICMP message to the source.

NDR.004

Level: CE-ERROR

Short Syntax: NDR.004 TTL zero *source_ip_address* -> *destination_ip_address*

Long Syntax: NDR.004 Time-to-live expired on packet from *source_ip_address* for *destination_ip_address*

Description: This message is generated when a packet is discarded because the time-to-live expired.

Cause: The packet has been through more routers than the initial value placed in the time-to-live field of the IP header by the originator. Many older systems use values of 15 or 30, which are not standard-conformant, and are often too small for current networks.

Action: Increase initial time-to-live value.

Cause: The packet was in a routing loop, going through a sequence of routers over and over until the time-to-live expired.

Action: Check the routing from the source of the packet to the destination, and see that there are no loops. However, temporary loops are an inevitable result of the timing out of routes in some routing protocols.

NDR.005

Level: CI-ERROR

Short Syntax: NDR.005 pkt *source_ip_address* -> *destination_ip_address* dsc rsn *reason_code*, nt *Network ID*

Long Syntax: NDR.005 Packet from *source_ip_address* for *destination_ip_address* discarded for reason *reason_code*,network *Network ID*

Description: An attempt was made to send the packet on the specified network, but it was not accepted for transmission on that network. The reason_code indicates why the packet was not accepted. If the reason was flow-control, an ICMP source quench will be sent to the sender, otherwise an ICMP destination unreachable will be sent.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

NDR.006

Level: C-INFO

Short Syntax: NDR.006 New cnnct rqst r_a *remote_addr* r_p *remote_port* l_p *local_port* nd *local_addr*

Long Syntax: NDR.006 New connection request r_a *remote_addr* r_p *remote_port* l_p *local_port* node *local_addr*

Description: A new connection request has ebstablished.

NDR.007

Level: UE-ERROR

Short Syntax: NDR.007 no cnnct r_a *remote_addr* r_p *remote_port* l_a *local_addr* l_p *local_port* flg *flags*

Long Syntax: NDR.007 no connection r_a *remote_addr* r_p *remote_port* l_a *local_addr* l_p *local_port* flg *flags*

Description: This packet is arriving for a connection that NDR do not have a connection record for

Action: Increment error counters and process locally.

NDR.008

Level: C-TRACE

Short Syntax: NDR.008 fwd fr *client_addr* cl *cluster_addr* pt *port_id* srv *server_addr*

Long Syntax: NDR.008 Forward from *client_addr* for cluster *cluster_addr* port *port_id* to server *server_addr*

Description: A TCP packet is forwarded to a server

NDR.009

Level: C-TRACE

Short Syntax: NDR.009 found FTP CTRL connection

Long Syntax: NDR.009 found FTP CTRL connection

Description: found FTP CTRL connection

NDR.010

Level: UE-ERROR

Short Syntax: NDR.010 no srv fnd src *remote_addr* clst *local_addr* pt *local_port*

Long Syntax: NDR.010 no server found for source *remote_addr* cluster *local_addr*I port *local_port*

Description: This packet is arriving for a connection that NDR do not have a connection record for

Action: Increment error counters and process locally.

NDR.011

Level: UE-ERROR

Short Syntax: NDR.011 unexp SYN src *remote_addr* clst *local_addr* pt *local_port* srv *server_addr*

Long Syntax: NDR.011 unexpected SYN source *remote_addr* cluster *local_addr* port *local_port* server *server_addr*

Description: Unexpected SYN bit is set in the packet.

Action: Let the server resolve it.

NDR.012

Level: C-TRACE

Short Syntax: NDR.012 clean up everything older than the limit.

Long Syntax: NDR.012 clean up everything older than the limit

Description: clean up everything older than the limit

NDR.013

Level: UE-ERROR

Short Syntax: NDR.013 bad INCONN rd rtrn *remote_addr* rmt pt *remote_port*

Long Syntax: NDR.013 bad INCONN record returned source *remote_addr* remote post *remote_port*

Description: Bad INCONN record returned.

Action: Continue executing.

NDR.014

Level: P-TRACE

Short Syntax: NDR.014 rcv frg pkt src *source_ip_address* dst *destination_ip_address*

Long Syntax: NDR.014 Receiving fragment packet from source *source_ip_address* for destination *destination_ip_address*

Description: A fragment packet is received.

NDR.015

Level: P-TRACE

Short Syntax: NDR.015 last frg pkt src *source_ip_address* dst *destination_ip_address*

Long Syntax: NDR.015 Last fragment packet from source *source_ip_address* for destination *destination_ip_address*

Description: A fragment packet is received.

NDR.016

Level: P-TRACE

Short Syntax: NDR.016 first frg pkt src *source_ip_address* dst *destination_ip_address*

Long Syntax: NDR.016 First fragment packet from source *source_ip_address* for destination *destination_ip_address*

Description: A fragment packet is received.

NDR.017

Level: P-TRACE

Short Syntax: NDR.017 fwd frg pkt pkt src *source_ip_address* dst *destination_ip_address* srv *server_addr*

Long Syntax: NDR.017 Forward a fragment packet from source *source_ip_address* for destination *destination_ip_address* server *server_addr*

Description: Forward a fragmentation packet to server.

NDR.018

Level: UI-ERROR

Short Syntax: NDR.018 discd pkt src *source_ip_address* dst *destination_ip_address*

Long Syntax: NDR.018 discard packet source *source_ip_address* destination *destination_ip_address*

Description: Discard the packet.

NDR.019

Level: P-TRACE

Short Syntax: NDR.019 Rply ARP clst *cluster_addr*

Long Syntax: NDR.019 Reply ARP request for cluster *cluster_addr*

Description: A cluster address is found for an ARP request

NDR.020

Level: UI-ERROR

Short Syntax: NDR.020 ARP rqst for clst *cluster_addr* not fnd

Long Syntax: NDR.020 ARP request for cluster *cluster_addr* is not found.

Description: A cluster address is not found for an ARP request

NDR.021

Level: P-TRACE

Short Syntax: NDR.021 adv *adv_name* on prt *adv_port* created.

Long Syntax: NDR.021 advisor *adv_name* on port *adv_port* created

Description: An advisor is created.

NDR.022

Level: P-TRACE

Short Syntax: NDR.022 adv *adv_name* on prt *adv_port* destroyed.

Long Syntax: NDR.022 advisor *adv_name* on port *adv_port* destroyed

Description: An advisor is destroyed.

NDR.023

Level: UI-ERROR

Short Syntax: NDR.023 err adding adv *adv_name* on prt *adv_port*

Long Syntax: NDR.023 error in adding advisor *adv_name* on port *adv_port*

Description: Cannot add the advisor due to either the port already is in used, or the advisor table is full, the advisor could not be established on specified port.

NDR.024

Level: UI-ERROR

Short Syntax: NDR.024 err in adv *adv_name* on prt *adv_port*

Long Syntax: NDR.024 error in advisor *adv_name* on port *adv_port*

Description: There is an error occurs on the advisor. The error is either the advisor: Failed to create socket for hostlist connection, Failed to connect to manager hostlist port Failed to send authorization successfuly for hostlist connection Failed to write hostlist command Authorization failed on hostlist connection Failed to read count of hosts Failed to read addresses of hosts

NDR.025

Level: P-TRACE

Short Syntax: NDR.025 adv mk cnntn on lcl addr *ip_addr* prt *port_number*

Long Syntax: NDR.025 advisor makes connection on addr *ip_addr* and port *port_number*

Description: The advisor is making a connection.

NDR.026

Level: UI-ERROR

Short Syntax: NDR.026 adv: sckt err code *error_code*

Long Syntax: NDR.026 advisor: socket error code *error_code*

Description: There is a socket error with the advisor

NDR.027

Level: UI-ERROR

Short Syntax: NDR.027 adv: cnntn fld on prt *port_number*

Long Syntax: NDR.027 advisor: connection failed on port *port_number*

Description: The advisor failed to make a connection.

NDR.028

Level: UI-ERROR

Short Syntax: NDR.028 mgr: host not in table

Long Syntax: NDR.028 manager: Tried to get the info on a host that is not in the table

Description: Tried to get the info on a host that is not in the table

NDR.029

Level: UI-ERROR

Short Syntax: NDR.029 mgr: Error reading metric report

Long Syntax: NDR.029 manager: Error reading metric report.

Description: There is an error reading metric report data, or number of metrics sent, or port number.

NDR.030

Level: C-INFO

Short Syntax: NDR.030 mgr: Metric table has been updated.

Long Syntax: NDR.030 manager: Metric table has been updated.

Description: Metric table has been updated.

NDR.031

Level: P-TRACE

Short Syntax: NDR.031 mgr: compute *comp_type* prop: *host_name host_weight host_totalweight host_weight_prop*

Long Syntax: NDR.031 manager: compute *comp_type* proportions : *host_name host_weight host_totalweight host_weight_prop*

Description: Computing proportion

NDR.032

Level: P-TRACE

Short Syntax: NDR.032 mgr: Port *port_number* has been updated

Long Syntax: NDR.032 manager: Port *port_number* has been updated

Description: A port has been updated

NDR.033

Level: P-TRACE

Short Syntax: NDR.033 ha: prev: *prev_state* evt: *event* cur: *cur_state*

Long Syntax: NDR.033 High Availability: previsous: *prev_state* event: *event* current: *cur_state*

Description: State change

NDR.034

Level: UI-ERROR

Short Syntax: NDR.034 ha: err in State *backup_state* Event *event*

Long Syntax: NDR.034 High Availability: error in State *backup_state* Event *event*

Description: error in HA

NDR.035

Level: UI-ERROR

Short Syntax: NDR.035 ha: err in State *backup_state* Event *event*

Long Syntax: NDR.035 High Availability: error in State *backup_state* Event *event*

Description: error in HA

NDR.036

Level: C-INFO

Short Syntax: NDR.036 ha: Send Gratuitous ARP for : *vec_address*

Long Syntax: NDR.036 High Availability: Send Gratuitous ARP for : *vec_address*

Description: Response tp ARP request

NDR.037

Level: C-INFO

Short Syntax: NDR.037 ha: Send pkt *cmd*

Long Syntax: NDR.037 High Availability: Send packet *cmd*

Description: Send packet

NDR.038

Level: C-INFO

Short Syntax: NDR.038 ha: rcv pkt *cmd*

Long Syntax: NDR.038 High Availability: receive packet *cmd*

Description: receive packet

NDR.039

Level: UI-ERROR

Short Syntax: NDR.039 internal IP addr is not set

Long Syntax: NDR.039 internal IP address is not set.

Description: internal IP address needs to be set in order advisors can open communication with manager.

NDR.040

Level: UE-ERROR

Short Syntax: NDR.040 err msg: *ec msg*

Long Syntax: NDR.040 error message: *ec msg*

Description: An error mesaage is set by setuerror function

NDR.041

Level: UE-ERROR

Short Syntax: NDR.041 No mem avail for init.

Long Syntax: NDR.041 No memory available for initialization.

Description: There is not enough memory from heap to allocate.

NDR.042

Level: C-INFO

Short Syntax: NDR.042 mgr: Invld mtrc rpt rcvd.

Long Syntax: NDR.042 manager: Invalid metric report received. Report ignored.

Description: The metric report received from the advisor does not appear to be valid. The metric report was ignored.

NDR.043

Level: UE-ERROR

Short Syntax: NDR.043 Total no. of unique srvr IP addr. exceeds *max_unique_servers*. Disabling the NDR

Long Syntax: NDR.043 Total no. of unique server IP addresses exceeds *max_unique_servers*. Disabling the NDR

Description: Too many physical servers(each with a unique server IP address) have been configured. Note that each of these physical servers can be configured under many clusters and ports. Also notice that Network Dispatcher will be disabled whenever this situation is detected.

Action: Modify configuration and make sure that total no. physical servers (each with a unique server IP address) is not more than maximum allowed for the given platform.

NDR.044

Level: UE-ERROR

Short Syntax: NDR.044 Total no. of srvr for port no. *port_no* exceeds *max_servers_per_port*. Disabling the NDR

Long Syntax: NDR.044 Total no. of server for port number *port_no* exceeds *max_servers_per_port*. Disabling the NDR

Description: Too many servers on a specific port number have been configured. Note that a unique physical server(with a unique server IP address) can be configured under specific port on multiple clusters and each instance counts towards the limit of maximum allowed for the given platform. Also notice that Network Dispatcher will be disabled whenever this situation is detected.

Action: Modify the configuration.

NDR.045

Level: UE-ERROR

Short Syntax: NDR.045 Error:Srvr addr *server_ip_address* is same as cluster address. Disabling the NDR

Long Syntax: NDR.045 Error:Server address *server_ip_address* is same as cluster address. Disabling the NDR

Description: Server address and Cluster address aren't unique.

Action: Modify the configuration.

Chapter 84. Neighbor Discovery Protocol for IPv6 (NDP6)

This chapter describes Neighbor Discovery Protocol for IPv6 (NDP6) messages. For information on message content and how to use the message, refer to the Introduction.

NDP6.001

Level: P-TRACE

Short Syntax: NDP6.001 unknown dest protocol address net *network ID*

Long Syntax: NDP6.001 Unknown destination protocol address net *network ID*

Description: This message is generated when a Neighbor Discovery request specifies an unknown protocol address (i.e. request not for this router).

Cause: Neighbor Discovery request for a host on this network that is not for this router.

Action: None needed. The request is dropped.

NDP6.002

Level: UI-ERROR

Short Syntax: NDP6.002 Send request failed reason *reason_code* net *network ID*

Long Syntax: NDP6.002 Transmission of request failed for reason *reason_code* net *network ID*

Description: An outgoing Neighbor Discovery request packet was dropped as the result of some problem in the router. The reason code gives the cause.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

NDP6.003

Level: U-INFO

Short Syntax: NDP6.003 Unknown hdwe space *hardware_address_space* net *network ID*

Long Syntax: NDP6.003 Unknown hardware space *hardware_address_space* net *network ID*

Description: An incoming Neighbor Discovery packet was received on a network which is not using Neighbor Discovery for address translation in any protocol.

Cause: The gateway is misconfigured.

Action: Correct the configuration.

Cause: A protocol is in use on that network which requires the use of the Neighbor Discovery protocol, but the router does not support that protocol.

Action: None.

NDP6.004

Level: UI-ERROR

Short Syntax: NDP6.004 recv: No mem for cache entry, prot *protocol_type* net *network ID*

Long Syntax: NDP6.004 receive: No memory for cache entry, protocol *protocol_type* net *network ID*

Description: During the input processing of a Neighbor Discovery packet, the router did not have memory available to make a Neighbor cache entry for the given protocol.

Cause: The router is extremely low on heap memory.

Action: Find some way to reduce memory usage.

NDP6.005

Level: P-TRACE

Short Syntax: NDP6.005 Unknown protocol type *protocol_type* net *network ID*

Long Syntax: NDP6.005 Unknown protocol type *protocol_type* net *network ID*

Description: An incoming Neighbor Discovery packet was received for a protocol that is not using the Neighbor Discovery protocol for address translation.

Cause: The gateway is misconfigured.

Action: Correct the configuration.

Cause: A protocol is in use on that network which requires the use of Neighbor Discovery, but the router does not support that protocol.

Action: None.

NDP6.006

Level: C-INFO

Short Syntax: NDP6.006 NDP6 init

Long Syntax: NDP6.006 Neighbor Discovery initialization

Description: This message is printed when Neighbor Discovery is initializing.

NDP6.007

Level: C-TRACE

Short Syntax: NDP6.007 New cache entry *hardware_address_space protocol_type* net *network ID*

Long Syntax: NDP6.007 New neighbor cache entry *hardware_address_space protocol_type* net *network ID*

Description: An incoming Neighbor Discovery packet addressed to this host contained a mapping which was not in the neighbor cache. A new cache entry was filled in with the information in the packet.

NDP6.008

Level: UI-ERROR

Short Syntax: NDP6.008 NDP unable to get memory

Long Syntax: NDP6.008 Neighbor Discovery Protocol unable to get memory

Description: Neighbor Discovery was unable to allocate the necessary memory. Neighbor Discovery is unable to run because of this.

Cause: There is a shortage in heap memory, possibly because too many memory intensive forwarders/protocols are running.

Action: Disable unnecesary forwarders/protocols or get more memory.

NDP6.009

Level: C-TRACE

Short Syntax: NDP6.009 Sending NA *source_IP_address destination_IP_address* net *network ID*

Long Syntax: NDP6.009 Neighbor Advertisement packet sent from *source_IP_address* to *destination_IP_address* net *network ID*

Description: An ICMP6 Neighbor Advertisement is being sent as the result of a request for a translation from another host.

NDP6.010

Level: U-INFO

Short Syntax: NDP6.010 Dropping RA *source_IP_address -> destination_IP_address* net *network ID*

Long Syntax: NDP6.010 Dropping router advertisement request packet received from *source_IP_address* to *destination_IP_address* net *network ID*

Description: A Router Advertisement Request was received from the source router by this router. The request is dropped because router advertisements from other routers are ignored.

NDP6.011

Level: U-INFO

Short Syntax: NDP6.011 Dropping Redirect *source_IP_address -> destination_IP_address* net *network ID*

Long Syntax: NDP6.011 Dropping redirect packet received from *source_IP_address* to *destination_IP_address* net *network ID*

Description: A Redirect message was received from the source router by this router. The redirect is dropped because the router should depend on routing protocol, then redirect message.

NDP6.012

Level: UI-ERROR

Short Syntax: NDP6.012 No iorb for send net *network ID*

Long Syntax: NDP6.012 No buffer for send request packet net *network ID*

Description: An outgoing reply packet was dropped as the result of a lack of buffers in the router.

Cause: There are many potential causes of this problem; temporary overloads, etc.

Action: Consult logging output from the rest of the router for more information. If the problem persists, contact Customer Service.

NDP6.013

Level: C-TRACE

Short Syntax: NDP6.013 Sending RA *source_IP_address destination_IP_address* net *network ID* life

Long Syntax: NDP6.013 Router Advertisement packet sent from *source_IP_address* to *destination_IP_address* net *network ID* life

Description: An ICMP6 Router Advertisement is being sent for one of the following reasons: 1. Periodic router advertisment. 2. Response to a router solicitation. 3. Notification that the router or interface is going down by advertising a router lifetime of zero.

NDP6.014

Level: C-INFO

Short Syntax: NDP6.014 Sending redirect *source_IP_address* -> *destination_IP_address* to *new_next_hop_IP_address*

Long Syntax: NDP6.014 Sending redirect for packet from *source_IP_address* to *destination_IP_address* to use router *new_next_hop_IP_address*

Description: The router is sending an ICMP6 Redirect, advising a source host on a directly connected network that there is a better first hop router for this traffic.

NDP6.015

Level: U-INFO

Short Syntax: NDP6.015 Dropping NS *source_IP_address* -> *destination_IP_address*

Long Syntax: NDP6.015 Dropping neighbor solicitation packet received from *source_IP_address* to *destination_IP_address*

Description: An ICMP6 Neighbor Solicitation was received from the source host by the router. The request is dropped because ICMP6 packet validity failed.

NDP6.016

Level: C-INFO

Short Syntax: NDP6.016 NS received *source_IP_address* -> *destination_IP_address*

Long Syntax: NDP6.016 Neighbor Solicitation request packet received from *source_IP_address* to *destination_IP_address*

Description: An ICMP6 Neighbor Solicitation Request was received from the source host by the router.

NDP6.017

Level: U-INFO

Short Syntax: NDP6.017 Dropping NA *source_IP_address* -> *destination_IP_address*

Long Syntax: NDP6.017 Dropping neighbor advertisement packet received from *source_IP_address* to *destination_IP_address*

Description: An ICMP6 Neighbor Advertisement was received from the source host by the router. The request is dropped because ICMP6 packet validity failed.

NDP6.018

Level: C-INFO

Short Syntax: NDP6.018 NA received *source_IP_address* -> *destination_IP_address*

Long Syntax: NDP6.018 Neighbor Advertise request packet received from *source_IP_address* to *destination_IP_address*

Description: An ICMP6 Neighbor Advertisement was received from the source host by the router.

NDP6.019

Level: C-INFO

Short Syntax: NDP6.019 Received RS *source_IP_address* -> *destination_IP_address*

Long Syntax: NDP6.019 Router Solicitation packet received from *source_IP_address* to *destination_IP_address*

Description: An ICMP6 Router Solicitation was received from the source host by the router.

NDP6.020

Level: U-INFO

Short Syntax: NDP6.020 Dropping RS *source_IP_address* -> *destination_IP_address*

Long Syntax: NDP6.020 Dropping Router Solicitation packet received from *source_IP_address* to *destination_IP_address*

Description: An ICMP6 Router Solicitation was received from the source host by the router. The request is dropped because ICMP6 packet validity failed.

NDP6.021

Level: UE-ERROR

Short Syntax: NDP6.021 Dropping invalid *ND_packet_type source_IP_address* -> *destination_IP_address*

Long Syntax: NDP6.021 Invalid neighbor discovery packet (*ND_packet_type*) received from *source_IP_address* to *destination_IP_address* is dropped

Description: An invalid Neighbor Discovery packet was received from the source host by the router. The packet is dropped.

Cause: The software found the net pointer to be invalid which may be caused by an invalid network configuration.

Action: Check network configuration.

NDP6.022

Level: C-TRACE

Short Syntax: NDP6.022 Sending NS *source_IP_address* -> *destination_IP_address* target *addr_to_resolve* net *network ID*

Long Syntax: NDP6.022 Neighbor Solicitation packet sent *source_IP_address* to *destination_IP_address* target *addr_to_resolve* net *network ID*

Description: An ICMP6 Neighbor Solicitation is being sent as the result of a request for a address translation.

Chapter 85. Next Hop Routing Protocol (NHRP)

This chapter describes Next Hop Routing Protocol (NHRP) messages. For information on message content and how to use the message, refer to the Introduction.

NHRP.001

Level: UE_ERROR

Short Syntax: NHRP.001 *caller_string* detected ext offst incorrect

Long Syntax: NHRP.001 *caller_string* detected ext offset too small or too big

Description: Extension offset is either too small or too large. If too large then the buffer is not big enough to include the specified extension.

Cause: Either Extension offset is wrong or the MTU is truely not big enough or there's an internal error.

Action: Verify the extension offset is valid. Resize the MTU for the network.

NHRP.002

Level: UE_ERROR

Short Syntax: NHRP.002 addr family mssmtch in *caller_string*: rcvd *fh_AddressFamily* vs cached *family*

Long Syntax: NHRP.002 addr family missmatch in *caller_string*: we received *fh_AddressFamily* and our cache is *family*

Description: While processing the specified process, we detected that the Address Family we received is not what's in our cache.

NHRP.003

Level: UE_ERROR

Short Syntax: NHRP.003 addr family *fh_AddressFamily* !supprtd in *caller_string*

Long Syntax: NHRP.003 addr family *fh_AddressFamily* not supported in *caller_string*

Description: While process the specified process, we detected that the Address Family we received is not one that we support.

NHRP.004

Level: UE_ERROR

Short Syntax: NHRP.004 in *caller_string*, detctd a loop *proto_addr*

Long Syntax: NHRP.004 while in *caller_string*, we detected that an NHRP frame is in a loop *proto_addr*.

Description: While processing an NHRP Packet, we found that we have processed this packet once before.

Cause: There is a real loop detected or someone else is using the same IP address as we are.

Action: Double check that there is not a duplicate IP address being used in the network.

NHRP.005

Level: UE_ERROR

Short Syntax: NHRP.005 sbntwrk id rcvd *subnet_id* not cfigd on nt *network ID*

Long Syntax: NHRP.005 subnetwork ID rcvd *subnet_id* not configured on the net *network ID*

Description: Not on the same switched connected network. The switched connected network have been subdivided. The sender is sending to a switched connected network that is not configured to be the same subnetwork.

NHRP.006

Level: UE_ERROR

Short Syntax: NHRP.006 nll ext not last found by *caller_string*

Long Syntax: NHRP.006 null extension found in the middle of the extensions by *caller_string*

Description: The null extension was found in the middle of the extension list.

NHRP.007

Level: C_INFO

Short Syntax: NHRP.007 proc sbntwrk id ext in rply

Long Syntax: NHRP.007 processing subnetwork id ext in a reply

Description: Processing a subnetwork ID extension in a reply.

NHRP.008

Level: C_INFO

Short Syntax: NHRP.008 proc sbntwrk id ext in req

Long Syntax: NHRP.008 processing subnetwork id ext in a request

Description: Processing a subnetwork ID extension in a request.

NHRP.009

Level: UE_ERROR

Short Syntax: NHRP.009 rspndr addr ext len=0 in rply

Long Syntax: NHRP.009 responder address ext length is zero is a reply

Description: While processing the transit extensions in a reply, found the responder address extension length equal zero. This means that the responder erroneously didn't fill in the extension.

NHRP.010

Level: UI_ERROR

Short Syntax: NHRP.010 unexpctd err hndlng in res_ext_hndlr

Long Syntax: NHRP.010 unexpected error handling in the res_ext_handler

Description: This is the default case of the common error handling for the IBM vendor private extension.

NHRP.011

Level: CI_ERROR

Short Syntax: NHRP.011 unspprtd cmp ext *ext_type* rcvd in *caller_string*

Long Syntax: NHRP.011 unsupported compulsory extension *ext_type* received in *caller_string*

Description: We do not have support for the specified compulsory extensions.

NHRP.012

Level: C_INFO

Short Syntax: NHRP.012 rspndr addr ext rcvd *respndr_addr*

Long Syntax: NHRP.012 responder address extension reply received *respndr_addr*

Description: The specified address is the responder that returned the reply.

NHRP.013

Level: C_INFO

Short Syntax: NHRP.013 *transit_ext_type* transit ext rsp rcvd

Long Syntax: NHRP.013 *transit_ext_type* transit extension response received

Description: This identifies the type of transit extension (forward or reverse) The next event lists the NHS's.

NHRP.014

Level: C_INFO

Short Syntax: NHRP.014 nhs: *nhs_paddr*

Long Syntax: NHRP.014 nhs: *nhs_paddr*

Description: This address is one of the nhs in the above extension. The order it is displayed is the order in the extension.

NHRP.015

Level: UE_ERROR

Short Syntax: NHRP.015 unrcgnzd ext type *ext_type* in *caller_string*

Long Syntax: NHRP.015 unrecognized extension type *ext_type* in *caller_string* reply

Description: Client didn't send the extension but the extension is in the reply.

Cause: This could be an internal bug, we sent the extension in the request but forgot to add processing to process the extension in the reply.

Action: fix the code.

Cause: Someone is adding extensions to our request packets.

NHRP.016

Level: C_INFO

Short Syntax: NHRP.016 lsi paddr= *proto_addr*, mac= *mac_addr*, atm= *atm_addr*

Long Syntax: NHRP.016 Lane Shortcuts to paddr= *proto_addr*, mac= *mac_addr*, atm= *atm_addr*

Description: A call to Lane Shortcut Interface to set up the NHRP Data Direct VCC to the specified addresses.

NHRP.017

Level: UE_ERROR

Short Syntax: NHRP.017 invld ATM addr rcvd *atm_addr atm_saddr* in *caller_string*

Long Syntax: NHRP.017 invalid ATM addr received *atm_addr atm_saddr* in *caller_string*

Description: The ATM address rcvd is not valid.

NHRP.018

Level: UE_ERROR

Short Syntax: NHRP.018 *caller* can't parse the frame at address *frame_address*.

Long Syntax: NHRP.018 *caller* cannot parse the frame at address *frame_address*.

Description: Caller cannot parse the frame at the given address.

NHRP.019

Level: P_TRACE

Short Syntax: NHRP.019 Trace NHRP/MPOA Ctrl pkt.

Long Syntax: NHRP.019 Trace NHRP/MPOA Ctrl pkt.

Description: NHRP/MPOA control frame packet tracing.

NHRP.020

Level: C_INFO

Short Syntax: NHRP.020 *caller_string*

Long Syntax: NHRP.020 *caller_string*

Description: Common information.

NHRP.021

Level: C_INFO

Short Syntax: NHRP.021 *integer*

Long Syntax: NHRP.021 *integer*

Description: This integer comes from an old TYPEN() call, from the days when we did not use ELS. See definitions in mscs.h.

NHRP.022

Level: UE_ERROR

Short Syntax: NHRP.022 Could not xmit pkt to *protocol_address*, out net intf *net_number*

Long Syntax: NHRP.022 Could not transmit NHRP packet to *protocol_address*, out network interface *net_number*

Description: Could not transmit NHRP packet

Cause: Routed path does not exist to this protocol address

Action: Repair routed path for sending protocol data to this protocol address (eg, IP).

NHRP.023

Level: C_INFO

Short Syntax: NHRP.023 fwding res reqst for *destination_addr* to *nhrp_server_addr*

Long Syntax: NHRP.023 forwarding resolution request for dest= *destination_addr* to nhs= *nhrp_server_addr*

Description: NHRP Resolution Request is being forwarded

NHRP.025

Level: UI_ERROR

Short Syntax: NHRP.025 *caller* can't xmit purge pkt to client= *cli_net_addr/ cli_node_addr* for dest_addr= *dest_net_addr/ dest_node_addr*

Long Syntax: NHRP.025 *caller* can't send purge pkt to client= *cli_net_addr/ cli_node_addr* for destination address= *dest_net_addr/ dest_node_addr*

Description: The specified caller can't Purge information.

NHRP.026

Level: UI_ERROR

Short Syntax: NHRP.026 cant get memory for *struct_type* in *caller_string*

Long Syntax: NHRP.026 can not get memory for *struct_type* in *caller_string*

Description: Cannot get memory for the structure specified in the specified routine.

NHRP.027

Level: CE_ERROR

Short Syntax: NHRP.027 nak *nhrp_client_addr* for *route_type_text* route to dest= *destination_addr* because *reason_text*

Long Syntax: NHRP.027 nak to client addr *nhrp_client_addr* for *route_type_text* shortcut route to destination *destination_addr* because *reason_text*

Description: NHRP Server cannot satisfy request received by client

Cause: The detailed reason is explained in reason_text

Action: No action is required, but reason_text can help determine how to possibly stop NAKs for this client/destination.

NHRP.028

Level: C_INFO

Short Syntax: NHRP.028 rcvd res reqst from *nhrp_client_addr* for *destination_addr*

Long Syntax: NHRP.028 received resolution request from *nhrp_client_addr* for *destination_addr*

Description: NHRP Server received a Resolution Request

NHRP.029

Level: C_INFO

Short Syntax: NHRP.029 xmit purge pkt to client= *nhrp_client_addr* for dest_addr= *destination_addr* w/ prefix= *prefix*

Long Syntax: NHRP.029 Sending purge pkt to client= *nhrp_client_addr* for destination address= *destination_addr* with prefix= *prefix*

Description: Purge Packet transmit information.

NHRP.030

Level: C_INFO

Short Syntax: NHRP.030 *function_name*: *general_message*

Long Syntax: NHRP.030 *function_name*: *general_message*

Description: The message is the description.

NHRP.031

Level: C_INFO

Short Syntax: NHRP.031 *function_name*: *general_message general_code*

Long Syntax: NHRP.031 *function_name*: *general_message general_code*

Description: The message is the description.

NHRP.032

Level: C_INFO

Short Syntax: NHRP.032 *function_name*: *general_message general_code*

Long Syntax: NHRP.032 *function_name*: *general_message general_code*

Description: The message is the description.

NHRP.033

Level: C_INFO

Short Syntax: NHRP.033 *function_name*: *general_message proto_addr*

Long Syntax: NHRP.033 *function_name*: *general_message proto_addr*

Description: The message is the description.

NHRP.034

Level: C_INFO

Short Syntax: NHRP.034 *function_name*: *general_message proto_addr1/ proto_addr2*

Long Syntax: NHRP.034 *function_name*: *general_message proto_addr1/ proto_addr2*

Description: The message is the description.

NHRP.035

Level: C_INFO

Short Syntax: NHRP.035 *function_name*: *general_message proto_addr1/ proto_addr2/ proto_addr3*

Long Syntax: NHRP.035 *function_name*: *general_message proto_addr1/ proto_addr2/ proto_addr3*

Description: The message is the description.

NHRP.036

Level: C_INFO

Short Syntax: NHRP.036 Exclude lst match for: *ip_addr*

Long Syntax: NHRP.036 Exclude list match for: *ip_addr*

Description: NHRP cannot process all or part of the NHRP packet because there is an IP address in the packet that matches one that is configured in the NHRP exclude list.

NHRP.037

Level: C_INFO

Short Syntax: NHRP.037 *caller_string* dtctd potential mac chgs so attempt to send purge

Long Syntax: NHRP.037 *caller_string* detected potential mac changes, so attempt to sen purge

Description: Potential Level 2 changes detected on the Lane Shortcut Interface server side, therefore, attempt to send an NHRP Purge if necessary.

NHRP.038

Level: C_INFO

Short Syntax: NHRP.038 *caller_string* cant get L2 parms, retries exceeded, attempt to send Purge

Long Syntax: NHRP.038 *caller_string* cannot get L2 parms, retries exceeded, attempt to send Purge

Description: Lane Shortcut Interface server waited for Level 2 parameters to be retrieved but ran out of

retries. So attempt to send an NHRP Purge.

NHRP.039

Level: U_INFO

Short Syntax: NHRP.039 *caller_string* dtctd *cache_type* has reached its limit

Long Syntax: NHRP.039 *caller_string* detected that the *cache_type* has reached its limit

Description: The NHS/MPS has reached the cache limit for the specified cache. Reconfigure the cache limit if necessary.

NHRP.040

Level: UI_ERROR

Short Syntax: NHRP.040 *caller_string* dtctd err with proto addr= *proto_addr*, type= *type*, table= *table*

Long Syntax: NHRP.040 *caller_string* detected error with protocol addr= *proto_addr*, type= *type*, table= *table*

Description: We can't get a mib entry based on the protocol address.

NHRP.041

Level: UI_ERROR

Short Syntax: NHRP.041 *function_name*: *general_message general_code*

Long Syntax: NHRP.041 *function_name*: *general_message general_code*

Description: The message is the description.

NHRP.042

Level: UI_ERROR

Short Syntax: NHRP.042 *function_name*: No dest for frame, rc = *general_code*

Long Syntax: NHRP.042 *function_name*: No destination for this frame, rc = *general_code*

Description: There is no destination address for this frame.

NHRP.043

Level: UI_ERROR

Short Syntax: NHRP.043 *function_name*: No src for frame, rc = *general_code*

Long Syntax: NHRP.043 *function_name*: No source for this frame, rc = *general_code*

Description: There is no source address from which to send this frame.

NHRP.044

Level: UI_ERROR

Short Syntax: NHRP.044 *function_name*: Can't crush frame, rc = *general_code*

Long Syntax: NHRP.044 *function_name*: Unable to crush NHRP frame, rc = *general_code*

Description: Unable to crush the local (internally used) NHRP frame before transmitting on the network.

NHRP.045

Level: UI_ERROR

Short Syntax: NHRP.045 *function_name*: Can't expand frame, rc = *general_code*

Long Syntax: NHRP.045 *function_name*: Unable to expand NHRP frame, rc = *general_code*

Description: Unable to expand the NHRP frame for the local (internally used) copy.

NHRP.046

Level: UI_ERROR

Short Syntax: NHRP.046 *function_name*: Proto *protocol_type* not handled

Long Syntax: NHRP.046 *function_name*: Protocol *protocol_type* not handled by NHRP

Description: Addresses of this protocol type are unsupported in the current release of NHRP.

Cause: Function passing in a protocol address of a type that is not supported by NHRP.

Action: None

NHRP.047

Level: UI_ERROR

Short Syntax: NHRP.047 *function_name*: No *data_type* available

Long Syntax: NHRP.047 *function_name*: No *data_type* could be allocated

Description: No structures of type data_type could be allocated.

Cause: No memory is available to allocated to a new structure.

Action: None

NHRP.048

Level: UI_ERROR

Short Syntax: NHRP.048 *function_name*: No ATM info for *proto_addr*, rc = *general_code*

Long Syntax: NHRP.048 *function_name*: Could not get ATM info for *proto_addr*, rc = *general_code*

Description: Could not retrieve the ATM address for this protocol address.

NHRP.049

Level: U_INFO

Short Syntax: NHRP.049 New RIF= *new_rif*, Current RIF= *current_rif*

Long Syntax: NHRP.049 New RIF= *new_rif*, Current RIF= *current_rif*

Description: The RIF associated with the parameters for a shortcut is different from the one that is currently being used for this shortcut. Both RIFs are non-NULL. The new RIF will now be associated with this shortcut.

NHRP.050

Level: U_INFO

Short Syntax: NHRP.050 *function_name*: *general_message*

Long Syntax: NHRP.050 *function_name*: *general_message*

Description: The message is the description.

NHRP.051

Level: U_INFO

Short Syntax: NHRP.051 New RIF= *new_rif*, Current RIF=NULL

Long Syntax: NHRP.051 New RIF= *new_rif*, Current RIF=NULL

Description: The RIF associated with the parameters for a shortcut is non-NULL. This is different from the NULL RIF currently in use. The new non-NULL RIF will now be associated with this shortcut.

NHRP.052

Level: U_INFO

Short Syntax: NHRP.052 New RIF=NULL, Current RIF= *current_rif*

Long Syntax: NHRP.052 New RIF=NULL, Current RIF= *current_rif*

Description: The RIF associated with the parameters for a shortcut is now NULL. This is different from the non-NULL RIF currently in use. The RIF associated with this shortcut will be changed to NULL.

NHRP.053

Level: C_INFO

Short Syntax: NHRP.053 fwding frame for *dest_net_addr*/ *dest_node_addr* to *nhs_net_addr*/ *nhs_node_addr*

Long Syntax: NHRP.053 forwarding req/rep for dest/src= *dest_net_addr*/ *dest_node_addr* to nhs= *nhs_net_addr*/ *nhs_node_addr*

Description: NHRP Resolution Request is being forwarded

NHRP.054

Level: C_INFO

Short Syntax: NHRP.054 *function_name*: Timer type *timer_type timer_state* for CCE: *proto_addr*

Long Syntax: NHRP.054 *function_name*: Timer type *timer_type timer_state* for ClientCacheElement: *proto_addr*

Description: none

NHRP.055

Level: UI_ERROR

Short Syntax: NHRP.055 *caller_string*

Long Syntax: NHRP.055 *caller_string*

Description: none

NHRP.056

Level: U_INFO

Short Syntax: NHRP.056 *function_name*: VCs marked down for *destination_protocol_addr*

Long Syntax: NHRP.056 *function_name*: VCs marked down for *destination_protocol_addr*

Description: The VC for this destination protocol has been closed or become invalid, so it has been marked down in the NHRP cache.

NHRP.057

Level: U_INFO

Short Syntax: NHRP.057 *function_name*: Function currently unimplemented

Long Syntax: NHRP.057 *function_name*: Function currently unimplemented

Description: This function is not currently implemented, so it does nothing.

NHRP.058

Level: C_INFO

Short Syntax: NHRP.058 *function_name*: NextHop (1483 or LSI) will not bypass *proto_addr1*

Long Syntax: NHRP.058 *function_name*: NHRP NextHop (1483 or LSI) will not bypass *proto_addr1*

Description: The 1483 or LSI NextHop received in Resolution Reply is the same as the routed-path NextHop.

NHRP.060

Level: UE_ERROR

Short Syntax: NHRP.060 *function_name*: *general_message*

Long Syntax: NHRP.060 *function_name*: *general_message*

Description: The message is the description.

NHRP.061

Level: UE_ERROR

Short Syntax: NHRP.061 *function_name*: *general_message general_code*

Long Syntax: NHRP.061 *function_name*: *general_message general_code*

Description: The message is the description.

NHRP.062

Level: UE_ERROR

Short Syntax: NHRP.062 *function_name*: *general_message general_code*

Long Syntax: NHRP.062 *function_name*: *general_message general_code*

Description: The message is the description.

NHRP.063

Level: UE_ERROR

Short Syntax: NHRP.063 *function_name*: *general_message proto_addr*

Long Syntax: NHRP.063 *function_name*: *general_message proto_addr*

Description: The message is the description.

NHRP.064

Level: UE_ERROR

Short Syntax: NHRP.064 *function_name*: NHRP vers mismatch, vers = *general_code*

Long Syntax: NHRP.064 *function_name*: NHRP version mismatch, version = *general_code*

Description: An NHRP frame was received with wrong version number.

NHRP.065

Level: UE_ERROR

Short Syntax: NHRP.065 *function_name*: Checksum not 0: *general_code*

Long Syntax: NHRP.065 *function_name*: Checksum not 0: *general_code*

Description: A frame was received whose checksum did not compute to 0.

NHRP.066

Level: UE_ERROR

Short Syntax: NHRP.066 *function_name*: *addr_name proto_addr* not cached

Long Syntax: NHRP.066 *function_name*: *addr_name proto_addr* not cached

Description: This address was not found in the client cache.

NHRP.067

Level: UE_ERROR

Short Syntax: NHRP.067 *function_name*: Hold time 0 rcvd from *proto_addr1* for *proto_addr2*

Long Syntax: NHRP.067 *function_name*: Holding time of 0 received from *proto_addr1* for *proto_addr2*

Description: A Holding time of 0 was received in response to a Resolution Request.

NHRP.068

Level: UE_ERROR

Short Syntax: NHRP.068 *function_name*: Can't match MTU for netp *general_pointer*

Long Syntax: NHRP.068 *function_name*: Can't provide correct MTU size for netp *general_pointer*

Description: There is no netp available that can be used for the MTU returned in the reply.

NHRP.069

Level: UE_ERROR

Short Syntax: NHRP.069 *function_name*: Rcvd fragment, length = *general_length*

Long Syntax: NHRP.069 *function_name*: Received a fragment, length = *general_length*

Description: The frame received was just a fragment.

NHRP.070

Level: CE_ERROR

Short Syntax: NHRP.070 *function_name*: *general_message*

Long Syntax: NHRP.070 *function_name*: *general_message*

Description: The message is the description.

NHRP.071

Level: CE_ERROR

Short Syntax: NHRP.071 *function_name*: NAK rqst sent from local client *proto_addr1* for dest *proto_addr2*, Code = *reply_code*

Long Syntax: NHRP.071 *function_name*: NAK for request made by local client *proto_addr1* for destination *proto_addr2*, Code = *reply_code*

Description: A NAK was received for local client proto_addr1 for request that it made for proto_addr2 if reply_code is non-zero; otherwise, client forced NAK by processing reply as if it were a NAK.

NHRP.072

Level: UI_ERROR

Short Syntax: NHRP.072 *caller_string* rc = *integer*

Long Syntax: NHRP.072 SNMP interface function *caller_string* returned error (rc = *integer*)

Description: SNMP interface function returned an error

NHRP.073

Level: C_INFO

Short Syntax: NHRP.073 NHRP LSI AddrStateChg (Active): nt *network ID*

Long Syntax: NHRP.073 NHRP LSI AddrStateChg (Active): nt *network ID*

Description: This NHRP LSI network has received an address state change from the switch. This means that the address ESI and SEL have been registered with the switch. NHRP LANE shortcuts can now be set up over this interface.

NHRP.074

Level: UI_ERROR

Short Syntax: NHRP.074 NHRP LSI GetAddrByHandle rc= *return_code*: nt *network ID*

Long Syntax: NHRP.074 NHRP LSI GetAddrByHandle rc= *return_code*: nt *network ID*

Description: While attempting to get the address from the switch, an error was detected.

NHRP.075

Level: UI_ERROR

Short Syntax: NHRP.075 NHRP LSI OpenCallSap rc= *return_code*: nt *network ID*

Long Syntax: NHRP.075 NHRP LSI OpenCallSap rc= *return_code*: nt *network ID*

Description: While attempting to open a call sap, an error was detected. A call sap is required in order to place or receive ATM calls to a remote destination.

NHRP.076

Level: UE_ERROR

Short Syntax: NHRP.076 NHRP LSI Addr Deactivated!: nt *network ID*

Long Syntax: NHRP.076 NHRP LSI Addr Deactivated!: nt *network ID*

Description: The ATM address for this NHRP LSI was deactivated. All calls are deleted. This NHRP LSI will be waiting for the address to be reactivated.

NHRP.077

Level: UE_ERROR

Short Syntax: NHRP.077 NHRP LSI Addr Refused!: nt *network ID*

Long Syntax: NHRP.077 NHRP LSI Addr Refused!: nt *network ID*

Description: The requested address has been refused by the switch.

Cause: The likely cause is that a duplicate MAC address is already registered with the switch.

NHRP.078

Level: UI_ERROR

Short Syntax: NHRP.078 NHRP LSI AddrStChg unknown: nt *network ID*

Long Syntax: NHRP.078 NHRP LSI AddrStChg unknown: nt *network ID*

Description: The Address State Change function was invoked, but the requested state is unknown.

NHRP.079

Level: UE_ERROR

Short Syntax: NHRP.079 NHRP LSI OpenDataPath failr(*return_code*): nt *network ID*

Long Syntax: NHRP.079 NHRP LSI OpenDataPath failr(*return_code*): nt *network ID*

Description: When attempting to open up a data path

with the specified parameters, a failure occured. The call will be hung up with the appropriate cause code.

NHRP.080

Level: C_INFO

Short Syntax: NHRP.080 NHRP LSI PlaceCallAck: nt *network ID*

Long Syntax: NHRP.080 NHRP LSI PlaceCallAck: nt *network ID*

Description: A call that we have placed has been received and acknowledged by the remote destination. We will open up a data path to the remote side, and will begin transmitting and receiving on the VCC.

NHRP.081

Level: U_INFO

Short Syntax: NHRP.081 NHRP LSI DisconnectCall: NULL CORRELATOR received

Long Syntax: NHRP.081 NHRP LSI DisconnectCall: NULL CORRELATOR received

Description: A call was released immediately before we received it.

NHRP.082

Level: U_INFO

Short Syntax: NHRP.082 NHRP LSI DisconnectCall: nt *network ID*

Long Syntax: NHRP.082 NHRP LSI DisconnectCall: nt *network ID*

Description: Either a call already active, or a call that we are placing has been released. The reason for the release is shown in additional ELS messages. This is a normal occurance. If the channel is required, we will reinitiate it.

Cause: Either the network or the remote user has released the call.

NHRP.083

Level: U_INFO

Short Syntax: NHRP.083 NHRP LSI DisconnectCall: rsn= *reason_code*, cause= *cause_code*, diagLen= *diag_len*, diagData[0]= *diag_data*

Long Syntax: NHRP.083 NHRP LSI DisconnectCall: rsn= *reason_code*, cause= *cause_code*, diagLen= *diag_len*, diagData[0]= *diag_data*

Description: The information in this message is the reason for which the call has been released.

NHRP.084

Level: U_INFO

Short Syntax: NHRP.084 NHRP LSI DisconnectCall: vpi= *vcc_vpi*, vci= *vcc_vci*, AtmAddr= *vcc_remote_atm_address*

Long Syntax: NHRP.084 NHRP LSI DisconnectCall: vpi= *vcc_vpi*, vci= *vcc_vci*, AtmAddr= *vcc_remote_atm_address*

Description: The information in this message is the channel vpi/vci, and remote atm address of the channel that is being disconnected.

NHRP.085

Level: U_INFO

Short Syntax: NHRP.085 NHRP LSI DisconnectCall WalkDwn PCR= *walk_down_PCR*, SCR= *walk_down_SCR*:nt *network ID*

Long Syntax: NHRP.085 NHRP LSI DisconnectCall WalkDwn PCR= *walk_down_PCR*, SCR= *walk_down_SCR*:nt *network ID*

Description: The call that was released, was released due to cell rate. The NHC code will attempt to walk down to commonly used data rates in order to establish a connection with the target listed in NHRP_XX(used to be ARP_48).

Cause: Either the network or the remote user has released the call due to cell rate mismatches.

NHRP.086

Level: UI_ERROR

Short Syntax: NHRP.086 NHRP LSI Register failure (rc= *return_code*): nt *network ID*

Long Syntax: NHRP.086 NHRP LSI Register failure (rc= *return_code*): nt *network ID*

Description: This NHRP LSI has failed to register as a user to the underlying device driver and net handler. This NHRP LSI will be inoperable.

Action: Reboot the router and contact the appropriate service personnel.

NHRP.087

Level: C_INFO

Short Syntax: NHRP.087 NHRP LSI Register successfull: nt *network ID*

Long Syntax: NHRP.087 NHRP LSI Register successfull: nt *network ID*

Description: This NHRP LSI has successfully registered with the underlying device driver and net handler. This is normal initialization.

NHRP.088

Level: UI_ERROR

Short Syntax: NHRP.088 NHRP LSI OpnBffFrmSap Failed (rc= *return_code*): nt *network ID*

Long Syntax: NHRP.088 NHRP LSI OpnBffFrmSap Failed (rc= *return_code*): nt *network ID*

Description: This NHRP LSI has failed while opening a buffered frame sap. This is caused by an internal error. This NHRP LSI will be inoperable.

Action: Reboot the router and contact the appropriate service personnel.

NHRP.089

Level: C_INFO

Short Syntax: NHRP.089 NHRP LSI Address Activation pending: nt *network ID*

Long Syntax: NHRP.089 NHRP LSI Address Activation pending: net *network ID*

Description: This NHRP LSI has initiated the sequence that registers it's ATM address with the switch. When the registration completes, another message of Address State change will be logged describing the status of the NHRP LSI's ATM address.

Action: No action required. This is normal processing.

NHRP.090

Level: C_INFO

Short Syntax: NHRP.090 NHRP LSI Address Activation success: nt *network ID*

Long Syntax: NHRP.090 NHRP LSI Address Activation success: nt *network ID*

Description: This NHRP LSI has been successful at activating an address.

NHRP.091

Level: CE_ERROR

Short Syntax: NHRP.091 NHRP LSI AAL IE:Not prsnt, or Invld AAL type (x *AAL_type*)

Long Syntax: NHRP.091 NHRP LSI AAL IE:Not present, or Invalid AAL type (x *AAL_type*)

Description: Invalid AAL type, AAL type should be AAL5

NHRP.092

Level: CE_ERROR

Short Syntax: NHRP.092 NHRP LSI AAL IE:Invld fwd max SDU sz (*fwd_max_SDU_size*)

Long Syntax: NHRP.092 NHRP LSI AAL IE:Invalid forward maximum SDU size (*fwd_max_SDU_size*)

Description: Forward maximum SDU size is not valid

NHRP.093

Level: CE_ERROR

Short Syntax: NHRP.093 NHRP LSI AAL IE:Invld bak max SDU sz for P2P call (*bak_max_SDU_size*)

Long Syntax: NHRP.093 NHRP LSI AAL IE:Invalid backward maximum SDU size for Point-to-Point Call (*bak_max_SDU_size*)

Description: For a point-to-point call, the backward maximum SDU size is invalid

---

NHRP.094

Level: UI_ERROR

Short Syntax: NHRP.094 NHRP LSI No Next Hop @ match: nt *network ID*

Long Syntax: NHRP.094 NHRP LSI No Next Hop Address match: net *network ID*

Description: While attempting to set up a shortcut, the corresponding data structures for the NHRP LSI were not found. It appears that the initialization of the NHRP LSI did not complete successfully.

NHRP.095

Level: UI_ERROR

Short Syntax: NHRP.095 NHRP LSI Invld user or frm sap hndl: nt *network ID*

Long Syntax: NHRP.095 NHRP LSI Invalid user or frame sap handle: nt *network ID*

Description: While attempting to set up an SVC, the NHRP LSI user handle or frame sap handle was NULL.

NHRP.096

Level: UI_ERROR

Short Syntax: NHRP.096 NHRP LSI Call sap invld: nt *network ID*

Long Syntax: NHRP.096 NHRP LSI Call sap invalid: network *network ID*

Description: While attempting to set up an SVC, the NHRP LSI user does not have a valid call sap.

NHRP.097

Level: UI_ERROR

Short Syntax: NHRP.097 NHRP LSI atmPlaceCall Failure (rc= *return_code*): nt *network ID*

Long Syntax: NHRP.097 NHRP LSI atmPlaceCall Failure (rc= *return_code*): net *network ID*

Description: While attempting to set up an SVC, the services of the device driver returned a value other than SUCCESS.

NHRP.098

Level: UI_ERROR

Short Syntax: NHRP.098 NHRP LSI atmPlaceCall Failure destination: Atm@= *vcc_remote_atm_address*

Long Syntax: NHRP.098 NHRP LSI atmPlaceCall Failure destination: AtmAddr= *vcc_remote_atm_address*

Description: While attempting to set up an SVC, the services of the device driver returned a value other than SUCCESS. This is the addresses of the remote station that we are attempting to establish a VCC with.

NHRP.099

Level: C_INFO

Short Syntax: NHRP.099 NHRP LSI atmPlaceCall Success: Atm@= *atm_address* nt *network ID*

Long Syntax: NHRP.099 NHRP LSI atmPlaceCall Success: AtmAddr= *atm_address* net *network ID*

Description: A call was successfully placed. This channel should show up on the new channel list. It has not yet been answered. When it is answered, a PlaceCallAck message will appear in the log.

NHRP.100

Level: C_INFO

Short Syntax: NHRP.100 Function *function_name* called, nt *network ID*

Long Syntax: NHRP.100 Function *function_name* called, on network *network ID*

Description: NHRP LSI function called

NHRP.102

Level: UE_ERROR

Short Syntax: NHRP.102 NHRP LSI: Inbnd data rcvd frm ATM@= *atm_addr* nt *network ID*

Long Syntax: NHRP.102 NHRP LSI: Inbound data received from ATM Address= *atm_addr* nt *network ID*

Description: The NHRP LSI has received data over a VCC. This should not occur since all NHRP LSI VCCs should be transmit only VCCs. The NHRP LSI will mark this ATM address as unusable and no other shortcuts will be set up to it.

Action: correct the situaion so that the LEC at the other end of the NHRP LSI VCC does not send data. reboot the router.

NHRP.103

Level: UE_ERROR

Short Syntax: NHRP.103 NHRP LSI: Invld Shrtct Atm@= *atm_addr* nt *network ID*

Long Syntax: NHRP.103 NHRP LSI: Invalid Shortcut Atm Addr= *atm_addr* nt *network ID*

Description: An NHRP LSI has been requested to set up a shortcut to an ATM address which has previously been determined to be unusable. This is a result of the NHRP LSI having previously received data over a VCC from this same ATM address. All NHRP LSI VCCs are transmit only.

NHRP.104

Level: C_INFO

Short Syntax: NHRP.104 NHRP LSI: New Shrtct Rqst NxtHp@= *next_hop_prot_addr* Atm@= *atm_addr* nt *network ID*

Long Syntax: NHRP.104 NHRP LSI: New Shortcut Request Next Hop Addr= *next_hop_prot_addr* Atm Addr= *atm_addr* nt *network ID*

Description: An NHRP LSI has been asked to set up a shortcut to a Next Hop for which no current shortcut is active. This is normal and a shortcut will now be set up to this new Next Hop.

NHRP.105

Level: C_INFO

Short Syntax: NHRP.105 NHRP LSI: Mdfy Shrtct Rqst NxtHp@= *next_hop_prot_addr* Atm@= *atm_addr* nt *network ID*

Long Syntax: NHRP.105 NHRP LSI: Modify Shortcut Request Next Hop Addr= *next_hop_prot_addr* Atm Addr= *atm_addr* nt *network ID*

Description: An NHRP LSI has been requested to set up a shortcut to a Next Hop for which there is already a shortcut. The parameters passed with this request will be checked against the parameters of the currently active shortcut and any changes in the new parameters will be reflected in the current shortcut.

NHRP.106

Level: U_INFO

Short Syntax: NHRP.106 NHRP LSI: MAC @ Chngd NxtHp@= *next_hop_prot_addr* New MAC@= *new_mac_addr* Crrnt MAC @= *current_mac_addr* nt *network ID*

Long Syntax: NHRP.106 NHRP LSI: MAC Address

Changed Next Hop Addr= *next_hop_prot_addr* New MAC Addr= *new_mac_addr* Current MAC Addr= *current_mac_addr* nt *network ID*

Description: The mac address of the destination associated with an existing shortcut has been found to have changed.

NHRP.107

Level: C_INFO

Short Syntax: NHRP.107 NHRP LSI: Delete Shrtct Rqst NxtHp@= *next_hop_prot_addr* nt *network ID*

Long Syntax: NHRP.107 NHRP LSI: Delete Shortcut Request Next Hop Addr= *next_hop_prot_addr* nt *network ID*

Description: A request to delete a shortcut has been received and will be executed.

NHRP.108

Level: UI_ERROR

Short Syntax: NHRP.108 NHRP LSI: VCC Setup Err NxtHp@= *next_hop_prot_addr* Atm@= *atm_addr* nt *network ID*

Long Syntax: NHRP.108 NHRP LSI: VCC Setup Error Next Hop Addr= *next_hop_prot_addr* Atm Addr= *atm_addr* nt *network ID*

Description: The return code to a request to set up a VCC indicates that the VCC was not set up. Prior ELS messages should indicate the reason for this situation.

NHRP.109

Level: U_INFO

Short Syntax: NHRP.109 NHRP LSI: RIF Chngd NxtHp@= *next_hop_prot_addr* Atm@= *atm_addr* nt *network ID*

Long Syntax: NHRP.109 NHRP LSI: RIF Changed Next Hop Addr= *next_hop_prot_addr* Atm Addr= *atm_addr* nt *network ID*

Description: The routing information field (rif) associated with an existing shortcut has been found to have changed.

NHRP.110

Level: C_INFO

Short Syntax: NHRP.110 NHRP LSI: Hldng Time Reset NxtHp@= *next_hop_prot_addr* nt *network ID*

Long Syntax: NHRP.110 NHRP LSI: Holding Time Reset Next Hop Addr= *next_hop_prot_addr* nt *network ID*

Description: The holding time parameter passed to the NHRP LSI is greater than the current time to live associated with an existing shortcut. The existing shortcut will be modified to reflect the new holding time.

NHRP.111

Level: C_INFO

Short Syntax: NHRP.111 NHRP LSI: Cntrl Frame *direction* Atm@= *atm_addr* nt *network ID*

Long Syntax: NHRP.111 NHRP LSI: Control Frame *direction* Atm Addr= *atm_addr* nt *network ID*

Description: A control frame has been received by an NHRP LSI.

NHRP.112

Level: P_TRACE

Short Syntax: NHRP.112 Trace NHRP LSI data packet

Long Syntax: NHRP.112 Trace NHRP LSI data packet

Description: Trace NHRP LSI data packet

NHRP.113

Level: P_TRACE

Short Syntax: NHRP.113 Trace NHRP LSI control packet

Long Syntax: NHRP.113 Trace NHRP LSI control packet

Description: Trace NHRP LSI control packet

NHRP.114

Level: UI_ERROR

Short Syntax: NHRP.114 *caller_string* is passed a ibm lec net

Long Syntax: NHRP.114 *caller_string* is being passed a non-forum compliant lec net

Description: We do not support lane shortcuts to IBM LECs.

NHRP.115

Level: UE_ERROR

Short Syntax: NHRP.115 *caller_string* dtctd invld ATM addr tl= *addr_tl* or sub addr tl= *sub_addr_tl*

Long Syntax: NHRP.115 *caller_string* detected invalid ATM address type/len= *addr_tl* or sub address type/len= *sub_addr_tl*

Description: Either the type or the length field of the addr or sub address is not valid.

NHRP.116

Level: UI_ERROR

Short Syntax: NHRP.116 *caller_string* cant get the ccb

Long Syntax: NHRP.116 *caller_string* can not get the ccb for 1483 transmit

Description: We can't get the ccb to do 1483 transmits.

NHRP.117

Level: UI_ERROR

Short Syntax: NHRP.117 *caller_string* cant find the atmarp side car

Long Syntax: NHRP.117 *caller_string* can not find the atmarp side car

Description: The atmarp side car is not there.

NHRP.118

Level: C_INFO

Short Syntax: NHRP.118 *caller_string* connction exsts w/ autorfrsh set, arp ent cant be owned by NHRP

Long Syntax: NHRP.118 *caller_string* detects existing connection with autorefresh configured

Description: Atm connection already exists. Auto refresh is configured so NHRP cannot own this arp entry.

NHRP.119

Level: C_INFO

Short Syntax: NHRP.119 *caller_string* new arp ent being added

Long Syntax: NHRP.119 *caller_string* new arp entry being added.

Description: NHRP is adding an arp entry to the atmarp.

NHRP.120

Level: C_INFO

Short Syntax: NHRP.120 *caller_string* holding time updated

Long Syntax: NHRP.120 *caller_string* holding time updated

Description: Lowered the holding time to what's in the arp entry.

NHRP.121

Level: UI_ERROR

Short Syntax: NHRP.121 *caller_string* detctd unexpctd ATM addr changed

Long Syntax: NHRP.121 *caller_string* detected unexpected ATM address changed

Description: This is an unexpected ATM address change.

NHRP.122

Level: UI_ERROR

Short Syntax: NHRP.122 *caller_string* rdatm is null for corrspndng macrd= *macrd_elem*

Long Syntax: NHRP.122 *caller_string* rdatm element is null for the associated macrd element= *macrd_elem*

Description: if macrd element exist then the corresponding rdatm must exist. There must be an internal bug that caused this.

NHRP.123

Level: UI_ERROR

Short Syntax: NHRP.123 *caller_string* rdatm= *rdatm_elem* is not pting back to the corrspndng macrd= *macrd_elem*

Long Syntax: NHRP.123 *caller_string* rdatm= *rdatm_elem* is not pointing back to the associated macrd element= *macrd_elem*

Description: if macrd element exist then the corresponding rdatm must exist. The rdatm element is not pointing back to the macrd element. There must be an internal bug that caused this.

NHRP.124

Level: C_INFO

Short Syntax: NHRP.124 *caller_string* detctd a protocol, mac or ri change

Long Syntax: NHRP.124 *caller_string* detected a protocol, mac or ri change

Description: A Protocol, MAC or RI change was detected in the function call.

NHRP.125

Level: C_INFO

Short Syntax: NHRP.125 *caller_string* free learp mac-atm elem for mac addr= *mac_addr*

Long Syntax: NHRP.125 *caller_string* free learp mac-atm element for mac addr= *mac_addr*

Description: Free LEARP_MAC_ATM_ENTRY for the specified mac address.

NHRP.126

Level: C_INFO

Short Syntax: NHRP.126 *caller_string* free learp rd-atm elem for next_rd= *next_rd*

Long Syntax: NHRP.126 *caller_string* free learp rd-atm element for the next route descriptor= *next_rd*

Description: Free LEARP_RD_ATM_ENTRY for the specified next route descriptor.

NHRP.127

Level: C_INFO

Short Syntax: NHRP.127 *caller_string* free learp mac-rd elem for nxt hp addr= *prot_addr*

Long Syntax: NHRP.127 *caller_string* free learp mac-rd element for next hop addr= *prot_addr*

Description: Free LEARP_MAC_RD_ENTRY for the specified ip address.

NHRP.128

Level: C_INFO

Short Syntax: NHRP.128 *caller_string* dtctd hldng time exprd for mac addr= *mac_addr*

Long Syntax: NHRP.128 *caller_string* detected holding time expired for mac addr= *mac_addr*

Description: Holding time has expired for the LEARP MAC-ATM Element Entry.

NHRP.129

Level: C_INFO

Short Syntax: NHRP.129 *caller_string* dtctd hldng time exprd for next_rd= *next_rd*

Long Syntax: NHRP.129 *caller_string* detected holding time expired for next route descriptor= *next_rd*

Description: Holding time has expired for the LEARP RD-ATM Element Entry.

NHRP.130

Level: C_INFO

Short Syntax: NHRP.130 *caller_string* rfrsh ent for mac addr= *mac_addr*

Long Syntax: NHRP.130 *caller_string* refresh entry for mac addr= *mac_addr*

Description: Refresh the LEARP MAC-ATM Element Entry for the specified mac address.

NHRP.131

Level: C_INFO

Short Syntax: NHRP.131 *caller_string* rfrsh ent for next_rd= *next_rd*

Long Syntax: NHRP.131 *caller_string* refresh entry for next route descriptor= *next_rd*

Description: Refresh the LEARP RD-ATM Element Entry for the specified next route descriptor.

NHRP.132

Level: U_INFO

Short Syntax: NHRP.132 LEC arp timer timed out for mac addr= *mac_addr*

Long Syntax: NHRP.132 LEC's arp timer timed out for mac addr= *mac_addr*

Description: NHRP triggered an LEARP for the specified MAC address but didn't get a reply.

NHRP.133

Level: U_INFO

Short Syntax: NHRP.133 LEC arp timr timed out for next_rd= *next_rd*

Long Syntax: NHRP.133 LEC's arp timer timed out for next route descriptor= *next_rd*

Description: NHRP triggered an LEARP for the specified route descriptor but didn't get a reply.

NHRP.134

Level: C_INFO

Short Syntax: NHRP.134 LEC rcvd LE_ARP rply for mac addr= *mac_addr*

Long Syntax: NHRP.134 LEC received LE_ARP reply for mac addr= *mac_addr*

Description: NHRP triggered an LEARP for the specified MAC address and got a reply.

NHRP.135

Level: C_INFO

Short Syntax: NHRP.135 LEC rcvd LE_ARP rply for next_rd= *next_rd*

Long Syntax: NHRP.135 LEC received LE_ARP reply for next route descriptor= *next_rd*

Description: NHRP triggered an LEARP for the specified route descriptor and got a reply.

NHRP.136

Level: U_INFO

Short Syntax: NHRP.136 LEC rcvd LE_ARP rply but atm addr changed, new addr= *atm_addr*

Long Syntax: NHRP.136 LEC received LE_ARP reply but atm address changed, new addr= *atm_addr*

Description: NHRP triggered an LEARP for a specified MAC address or Route Descriptor, and detected a change in the ATM address.

NHRP.137

Level: C_INFO

Short Syntax: NHRP.137 entry exists in LEC's arp table for mac_addr= *mac_addr*

Long Syntax: NHRP.137 entry exists in LEC's arp table for mac_addr= *mac_addr*

Description: NHRP triggered an LEARP for the specified MAC address but didn't get a reply.

NHRP.138

Level: C_INFO

Short Syntax: NHRP.138 entry exists in LEC's arp table for next_rd= *next_rd*

Long Syntax: NHRP.138 entry exists in LEC's arp table for next route descriptor= *next_rd*

Description: NHRP triggered an LEARP for the specified route descriptor but didn't get a reply.

NHRP.139

Level: UI_ERROR

Short Syntax: NHRP.139 LEC arp tbl is full, nhrp cant get entry

Long Syntax: NHRP.139 LEC's arp table is full, nhrp cannot get an entry

Description: LEC's arp table is full and nhrp cannot get an entry. get a reply.

Action: Configure the LEC's arp table size to a larger value.

NHRP.140

Level: U_INFO

Short Syntax: NHRP.140 LEC notifd NHRP of a tplgy change in net= *net_no*

Long Syntax: NHRP.140 LEC has notified NHRP of a topology change in net= *net_no*

Description: LEC notified NHRP that there's been a topology change.

NHRP.141

Level: C_INFO

Short Syntax: NHRP.141 *call_string* retrvd MAC addr succssflly from ARP for *proto_addr*

Long Syntax: NHRP.141 *call_string* retrieved MAC address successfully from ARP for *proto_addr*

Description: NHRP calls the ARP code to get the MAC address. If it can't get the MAC, NHRP waits a second before retrying.

NHRP.142

Level: C_INFO

Short Syntax: NHRP.142 *call_string* found *element_type* elemnt in del pendng

Long Syntax: NHRP.142 *call_string* found *element_type* element in delete pending state

Description: The caller found an learp element in delete pending state.

NHRP.143

Level: C_INFO

Short Syntax: NHRP.143 *call_string* cant find *element_type* elemnt

Long Syntax: NHRP.143 *call_string* cannot find *element_type* element

Description: The caller cannot find the learp element.

NHRP.144

Level: C_INFO

Short Syntax: NHRP.144 *caller_string* detctd LE_Regstrtn faild for *element_type*= *mac_addr* on net= *net_no*

Long Syntax: NHRP.144 *caller_string* detected LE_Registration failed for *element_type*= *mac_addr* on net= *net_no*

Description: NHRP's registration of a MAC/RD and ATM address that does not belong in this ELAN failed.

NHRP.145

Level: C_INFO

Short Syntax: NHRP.145 *caller_string* detctd LE_Regstrtn workd for *element_type*= *mac_addr* on net= *net_no*

Long Syntax: NHRP.145 *caller_string* detected LE_Registration worked for *element_type*= *mac_addr* on net= *net_no*

Description: NHRP's registration of a MAC/RD and

ATM address that does not belong in this ELAN was successful.

NHRP.146

Level: C_INFO

Short Syntax: NHRP.146 LE_Regstrtn pendng for *element_type*= *mac_addr* on net= *net_no*

Long Syntax: NHRP.146 LE_Registration pending for *element_type*= *mac_addr* on net= *net_no*

Description: NHRP's registration of a MAC/RD and ATM address that does not belong in this ELAN is pending.

NHRP.147

Level: UI_ERROR

Short Syntax: NHRP.147 Cant send LE_Regstrtn for *element_type*= *mac_addr* on net= *net_no*

Long Syntax: NHRP.147 Cannot send the LE_Registration for *element_type*= *mac_addr* on net= *net_no*

Description: LEC having problems sending the LE_Registration of a MAC/RD and ATM address that does not belong in this ELAN

NHRP.148

Level: C_INFO

Short Syntax: NHRP.148 entry expird for *element_type*= *mac_addr* but LE_Regstrtn pendng

Long Syntax: NHRP.148 entry expired for *element_type*= *mac_addr* but LE_Registration pending

Description: The holding time for NHRP's registration of a MAC/RD and ATM address that does not belong in this ELAN has expired. However, there's a LE_Registration outstanding. This entry will not be deleted now.

NHRP.149

Level: C_INFO

Short Syntax: NHRP.149 LE_Regstrtn entry expird for *element_type*= *mac_addr* and markd to be deltd

Long Syntax: NHRP.149 LE_Registration entry expired for *element_type*= *mac_addr* and marked to be deletd

Description: The holding time for NHRP's registration of a MAC/RD and ATM address that does not belong in this ELAN has expired. The entry is marked to be deleted.

NHRP.150

Level: C_INFO

Short Syntax: NHRP.150 LE_Regstrtn entry expird for *element_type*= *mac_addr* unreg the entry

Long Syntax: NHRP.150 LE_Registration entry expired for *element_type*= *mac_addr*, unregister the entry

Description: The holding time for NHRP's registration of a MAC/RD and ATM address that does not belong in this ELAN has expired. The entry is marked to be deleted. This entry was registered successfully so NHRP will now unregister the entry.

NHRP.151

Level: UE_ERROR

Short Syntax: NHRP.151 *caller_string* detctd invalid lsi lan_type= *lan_type*

Long Syntax: NHRP.151 *caller_string* detected invalid lsi lan_type= *lan_type*

Description: NHRP does not recognize the Lane Shortcut Interface lan_types.

NHRP.152

Level: C_INFO

Short Syntax: NHRP.152 LANE shrtct to one of our int = *ip_addr*

Long Syntax: NHRP.152 LANE shortcut to one of our interface= *ip_addr*

Description: NHRP is allowing a shortcut to one of the NHS's LEC IP address.

NHRP.153

Level: C_INFO

Short Syntax: NHRP.153 NHS rcvd Res Req

Long Syntax: NHRP.153 NHS received Resolution Request

Description: NHS has received a resolution request.

NHRP.154

Level: UE_ERROR

Short Syntax: NHRP.154 *caller_string* dtctd src or dst proto len err

Long Syntax: NHRP.154 *caller_string* detected source or destination protocol length error

Description: Caller detected protocol length error.

NHRP.155

Level: CE_ERROR

Short Syntax: NHRP.155 nak *src_net_addr/ src_node_addr* for *route_type_text* route to dest= *dest_net_addr/ dest_node_addr* because *reason_text*

Long Syntax: NHRP.155 nak to addr *src_net_addr/ src_node_addr* for *route_type_text* shortcut route to destination *dest_net_addr/ dest_node_addr* because *reason_text*

Description: NHRP Server cannot satisfy request received by client

Cause: The detailed reason is explained in reason_text

Action: No action is required, but reason_text can help determine how to possibly stop NAKs for this client/destination.

NHRP.156

Level: UE_ERROR

Short Syntax: NHRP.156 Could not delete Imp Cache entry for dest= *proto_addr*, pfx= *prefix*, cid= *cacheid*

Long Syntax: NHRP.156 Could not delete Imposition Cache entry for dest= *proto_addr*, prefix= *prefix*, cacheid= *cacheid*

Description: Deleting an Imposition Cache entry for e-mpc initiated purge failed

---

NHRP.161

Level: UE_ERROR

Short Syntax: NHRP.161 *caller_string* dtctd dst unreachable to *proto_addr*

Long Syntax: NHRP.161 *caller_string* detected destination unreachable to *proto_addr*

Description: The caller has no route to the specified destination.

NHRP.162

Level: UE_ERROR

Short Syntax: NHRP.162 *caller_string* dtctd hop count exceeded in the NHRP fwd pkt

Long Syntax: NHRP.162 *caller_string* detected hop count exceeded in the NHRP forward packet.

Description: Caller is forwarding NHRP packet but exceeded the hop count.

NHRP.163

Level: C_INFO

Short Syntax: NHRP.163 *caller_string* cant get 1483 ATM addr

Long Syntax: NHRP.163 *caller_string* cannot get 1483 ATM address

Description: Caller is cannot get 1483 ATM address must queue the request and try again later.

NHRP.164

Level: UI_ERROR

Short Syntax: NHRP.164 *caller_string* dtctd err in the q mngmnt for *queue_type*

Long Syntax: NHRP.164 *caller_string* detected error in the queue management for *queue_type*

Description: Caller detected error while processing an element on a queue. There is a missmatch in the size of the queue and what's queued.

NHRP.165

Level: C_INFO

Short Syntax: NHRP.165 *caller_string* cant get MAC or corresponding ATM addr

Long Syntax: NHRP.165 *caller_string* cannot get MAC or corresponding ATM address

Description: Caller is cannot get either the MAC or the corresponding ATM address must queue the request and try again later.

NHRP.166

Level: UI_ERROR

Short Syntax: NHRP.166 *caller_string* invalid rc from *called_function*

Long Syntax: NHRP.166 *caller_string* invalid return code from *called_function*

Description: Caller encountered invalid return code.

NHRP.167

Level: C_INFO

Short Syntax: NHRP.167 NHS sending a *reply_type* to *src_proto_addr*

Long Syntax: NHRP.167 NHRP Server sending a *reply_type* to *src_proto_addr*

Description: NHRP Server is sending a the specified reply to the specified client.

NHRP.168

Level: UI_ERROR

Short Syntax: NHRP.168 NHS cant send a ResReply to *src_proto_addr*

Long Syntax: NHRP.168 NHRP Server cannot send a Resolution Reply to *src_proto_addr*

Description: NHRP Server cannot send a Resolution Reply to the specified client.

NHRP.169

Level: C_INFO

Short Syntax: NHRP.169 *caller_string* ARP/LEARP was successful

Long Syntax: NHRP.169 *caller_string* ARP and/or LEARP was successful

Description: Caller got the MAC and or ATM address needed to send a Resolution Reply.

NHRP.170

Level: UE_ERROR

Short Syntax: NHRP.170 Zero Hop Cli snding a RegReq

Long Syntax: NHRP.170 Zero Hop Client is sending a Registration Request.

Description: Zero Hop or Route Switching Client is sending a Registration Request.

NHRP.171

Level: C_INFO

Short Syntax: NHRP.171 *caller_string* forwarding *packet_type* pkt

Long Syntax: NHRP.171 *caller_string* forwarding *packet_type* packet

Description: Caller is forwarding the specified packet type.

NHRP.172

Level: UE_ERROR

Short Syntax: NHRP.172 *caller_string* cant forward *packet_type* pkt

Long Syntax: NHRP.172 *caller_string* cannot forward *packet_type* packet

Description: Caller cannot forward the specified packet.

Cause: NHS is not permitted to the next hop or the nexthop's net is not NHRP enabled or the nexthop's net is not switch connected to input net.

NHRP.173

Level: C_INFO

Short Syntax: NHRP.173 *caller_string* rcvd a *packet_type* pkt destined to me

Long Syntax: NHRP.173 *caller_string* received a *packet_type* packet destined to me

Description: Caller received the specified packet type destined to the NHS/MPS.

NHRP.174

Level: UE_ERROR

Short Syntax: NHRP.174 NHS dtctd a Proto-ATM mapping chg on a RegRequest spcfd as unique..src= *proto_addr*

Long Syntax: NHRP.174 NHS detected Protocol-ATM mapping change on a RegRequest specified as unique..source= *proto_addr*

Description: NHRP Server detected a Protocol and ATM mapping change on a refresh of a registration request that was specified as unique.

NHRP.175

Level: C_INFO

Short Syntax: NHRP.175 NHS ran out of mem for client registrations

Long Syntax: NHRP.175 NHS ran out of memory for client registrations

Description: NHRP Server ran out of memory to serve client registrations.

Cause: Either NHS cannot get memory or we have reached the configured number of clients to be registered.

NHRP.176

Level: UI_ERROR

Short Syntax: NHRP.176 *caller_string* cant recgnz the NHRP pkt type = *packet_type_value*

Long Syntax: NHRP.176 *caller_string* cannot recognize the NHRP packet type = *packet_type_value*

Description: The caller does not recognize the NHRP Packet Type.

NHRP.177

Level: UE_ERROR

Short Syntax: NHRP.177 *caller_string* dtctd NHRP pktsz= *pktsize* greater than the input net's MTU= *mtu*

Long Syntax: NHRP.177 *caller_string* detected NHRP

packet size= *pktsize* greater than the input net's MTU= *mtu*

Description: NHRP packet size is greater than the MTU of the input net. The MTU is maximum data size minus the LLC.

NHRP.178

Level: UI_ERROR

Short Syntax: NHRP.178 *caller_string* dtctd NHRP pktsz= *pktsize* greater than bytes rcvd= *bytes_rcvd*

Long Syntax: NHRP.178 *caller_string* detected NHRP packet size= *pktsize* greater than bytes received= *bytes_rcvd*

Description: NHRP packet size is greater than bytes received.

NHRP.179

Level: UI_ERROR

Short Syntax: NHRP.179 *caller_string* dtctd bytes rcvd= *bytes_rcvd* greater than max pkt size= *max_pkt_sz*

Long Syntax: NHRP.179 *caller_string* detected NHRP bytes received= *bytes_rcvd* greater than max packet size= *max_pkt_sz*

Description: NHRP bytes received is greater than the maximum data size for this net.

NHRP.180

Level: U_INFO

Short Syntax: NHRP.180 *caller_string* dtctd iniatlst == NULL implies IP not in the box

Long Syntax: NHRP.180 *caller_string* detected iniatlst == NULL implies IP not in the box

Description: IP is not configured in this NHS/MPS.

NHRP.181

Level: C_INFO

Short Syntax: NHRP.181 *caller_string* dtctd no IP defined on the physical net= *net_num*

Long Syntax: NHRP.181 *caller_string* detected that no IP address is defined on the physical net= *net_num*

Description: No IP address is configured on the physical net. This may limit NHRP shortcuts.

NHRP.182

Level: C_INFO

Short Syntax: NHRP.182 *caller_string* using anthr net= *net_num* to allw shrtcts

Long Syntax: NHRP.182 *caller_string* using another net= *net_num* to allow shortcuts

Description: NHRP tries to find a different comparable net for shortcuts.

NHRP.183

Level: C_INFO

Short Syntax: NHRP.183 *caller_string* no alternate net found for shrtcts

Long Syntax: NHRP.183 *caller_string* no alternate net found for shortcuts

Description: Can't find an alternate interface to allow shortcuts on.

NHRP.184

Level: U_INFO

Short Syntax: NHRP.184 *caller_string* rcvd NHRP pkts on intrfce= *net_num*

Long Syntax: NHRP.184 *caller_string* received NHRP pkts on interface= *net_num*

Description: Received NHRP packets on an interface that does not have NHRP enabled.

NHRP.185

Level: UI_ERROR

Short Syntax: NHRP.185 *caller_string* cant recgnz the dest_type = *dest_type_value*

Long Syntax: NHRP.185 *caller_string* cannot recognize the dest_type = *dest_type_value*

Description: The caller does not recognize the dest_type.

NHRP.186

Level: UI_ERROR

Short Syntax: NHRP.186 *caller_string* called n_send() but it cant send the NHRP pkt

Long Syntax: NHRP.186 *caller_string* called n_send() but it cannot send the NHRP packet

Description: n_send returned a bad return code.

NHRP.187

Level: UI_ERROR

Short Syntax: NHRP.187 *caller_string* called w/ bad input parm

Long Syntax: NHRP.187 *caller_string* called with bad input parameter

Description: One of the input parameter is incorrect.

NHRP.188

Level: UI_ERROR

Short Syntax: NHRP.188 *caller_string* cant find the nxthp to send NHRP/MPOA Packet

Long Syntax: NHRP.188 *caller_string* cannot find the nexthop to send NHRP/MPOA Packet

Description: Can't send the NHRP Packet out any interface.

NHRP.189

Level: UI_ERROR

Short Syntax: NHRP.189 *caller_string* cant recgnz proto_state = *proto_state*

Long Syntax: NHRP.189 *caller_string* cannot recognize the proto_state = *proto_state*

Description: The caller does not recognize the enable protocol bit pattern.

NHRP.190

Level: C_INFO

Short Syntax: NHRP.190 *caller_string* NHRP initialized on net = *net_no*

Long Syntax: NHRP.190 *caller_string* NHRP initialized on network number = *net_no*

Description: Notification that an NHRP enabled net has been initialized.

NHRP.191

Level: C_INFO

Short Syntax: NHRP.191 *caller_string* NHRP enbld net = *net_no* is down.

Long Syntax: NHRP.191 *caller_string* NHRP enabled net = *net_no* is down.

Description: Notification that an NHRP enabled net is down.

NHRP.192

Level: U_INFO

Short Syntax: NHRP.192 Rcvd Err Ind w/ err code= *error_code*, err offset= *error_offset*, from= *from_proto_addr*

Long Syntax: NHRP.192 Received Error Indication w/ error code= *error_code*, error offset= *error_offset*, from= *from_proto_addr*

Description: Notification that an Error Indication Packet is received and targeted to us.

NHRP.193

Level: U_INFO

Short Syntax: NHRP.193 *caller_string* rcvd err rc from IP Route Table for *ip_addr*: rte= *rte*, inrretyp= *inrretyp*

Long Syntax: NHRP.193 *caller_string* received error return code from IP Routing Table for *ip_addr*: rte= *rte*, inrretyp= *inrretyp*

Description: Notification that there is a ip routing error.

NHRP.194

Level: C_INFO

Short Syntax: NHRP.194 *caller_string* rcvd a req for *ip_addr* which is one of our proto addr

Long Syntax: NHRP.194 *caller_string* received a request for *ip_addr*, which is one of our protocol address

Description: We received a request to shortcut to one of our protocol addresses.

NHRP.195

Level: U_INFO

Short Syntax: NHRP.195 *caller_string* rcvd out_net=sink net from IP Route Table for *ip_addr*

Long Syntax: NHRP.195 *caller_string* received out_net=sink net from IP Routing Table for *ip_addr*

Description: The output net from the IP Routing Table call is a sink net.

NHRP.196

Level: U_INFO

Short Syntax: NHRP.196 *caller_string* cant route *dest*, no explicity defined NHRP IP Servers

Long Syntax: NHRP.196 *caller_string* cannot route *dest*, there is no explicity defined NHRP IP Servers

Description: We cannot use the routed path to route NHRP Packet and there are no NHRP IP Servers defined.

NHRP.197

Level: C_INFO

Short Syntax: NHRP.197 *caller_string* scan routing tbl bcause change occurred since last scan

Long Syntax: NHRP.197 *caller_string* scan routing table because change has occurred since last scan

Description: Notification that NHRP is scanning the routing table.

NHRP.198

Level: UI_ERROR

Short Syntax: NHRP.198 Invalid prefix= *prefix_flag* in *caller_string*

Long Syntax: NHRP.198 Invalid prefix flag= *prefix_flag* detected in *caller_string*

Description: Detected an invalid prefix flag.

NHRP.199

Level: C_INFO

Short Syntax: NHRP.199 *caller_string* did not dtct mac chgs so no purge triggered.

Long Syntax: NHRP.199 *caller_string* did not detect mac changes, so no purge is triggered.

Description: No Level 2 changes detected on the Lane Shortcut Interface server side, therefore, no NHRP Purge is sent.

NHRP.200

Level: C_INFO

Short Syntax: NHRP.200 *caller_string* waiting for L2 parms to be retrieved bfore chgs can be dtctd

Long Syntax: NHRP.200 *caller_string* waiting for L2 parms to be retrieved before changes can be detected

Description: Lane Shortcut Interface server must wait for Level 2 parameters to be retrieved before it can detect any changes to send an NHRP Purge.

NHRP.201

Level: UE_ERROR

Short Syntax: NHRP.201 ATM netids mismatch on nt *in_net* (netid *in_netid*) and nt *out_net* (netid *out_netid*)

Long Syntax: NHRP.201 ATM network-ids mismatch on net *in_net* (netid *in_netid*) and net *out_net* (netid *out_netid*)

Description: Inbound net and outbound net are not on same switched connected network. The ATM network-ids of the inbound net and outbound net are configured with different values.

NHRP.202

Level: UI_ERROR

Short Syntax: NHRP.202 Cant send LE_Unregstrtn for *element_type*= *mac_addr* on net= *net_no*

Long Syntax: NHRP.202 Cannot send the LE_Unregistration for *element_type*= *mac_addr* on net= *net_no*

Description: LEC having problems sending the LE_Unregistration of a MAC/RD and ATM address that does not belong in this ELAN

NHRP.203

Level: C_INFO

Short Syntax: NHRP.203 *caller_string* Cant register existing MAC/RD with new ATM addr for *element_type*= *mac_addr* on net= *net_no*

Long Syntax: NHRP.203 *caller_string* Cannot register existing MAC/RD with new ATM address for *element_type*= *mac_addr* on net= *net_no*

Description: NHRP's registration of a MAC/RD and ATM address failed

---

NHRP.204

Level: C_INFO

Short Syntax: NHRP.204 *caller_string* Rings not unique in merged RIF (ring *ring_no*)

Long Syntax: NHRP.204 *caller_string* Ring numbers not unique in merged RIF (ring *ring_no*)

Description: Merged RIF for 0-hop routing is in error

NHRP.205

Level: C_INFO

Short Syntax: NHRP.205 *caller_string* Merged RIF too long (driflen= *rif_len* sriflen= *rif_len* mriflen= *rif_len*)

Long Syntax: NHRP.205 *caller_string* Merged RIF exceeded MAX_RIF_LEN (dest riflen= *rif_len*, src riflen= *rif_len*, merged riflen= *rif_len*)

Description: Merged RIF for 0-hop routing is too long

NHRP.206

Level: C_INFO

Short Syntax: NHRP.206 *caller_string* No free virtual RDs

Long Syntax: NHRP.206 *caller_string* All virtual route-descriptors are in use

Description: All virtual Route-Descriptors are in use

NHRP.207

Level: UI_ERROR

Short Syntax: NHRP.207 *caller_string* retd failure for mac= *caller_string*

Long Syntax: NHRP.207 *caller_string* returned failure for mac address= *caller_string*

Description: The LEC code returned failure when trying to get local_flag for 0-hop client

NHRP.208

Level: C_INFO

Short Syntax: NHRP.208 0-Hop detd bridging type mismatch between mac= *caller_string* and mac= *caller_string*

Long Syntax: NHRP.208 0-Hop detected transparent and source-route bridges between mac= *caller_string* and mac= *caller_string*

Description: The 0-Hop routing incompatibility due to mixed bridging types

---

NHRP.209

Level: C_INFO

Short Syntax: NHRP.209 0-Hop rings overlap on net= *net_no*

Long Syntax: NHRP.209 0-Hop virtual ring range overlapping with another router on net= *net_no*

Description: Warning user to configure non-overlapping 0-Hop virtual ring ranges

---

NHRP.210

Level: UE_ERROR

Short Syntax: NHRP.210 *caller_string* dtctd no 0-hop reg tbl alloc on net= *net_num*

Long Syntax: NHRP.210 *caller_string* detected that 0-hop registration table is not allocated on net= *net_num*

Description: 0-hop registration table was not allocated on this net

---

NHRP.211

Level: UI_ERROR

Short Syntax: NHRP.211 Function *caller_string*, no valid LSI net on intf *net_num*

Long Syntax: NHRP.211 Function *caller_string* called, no valid LSI net found on interface *net_num*

Description: NHRP LSI net SRAM record is not defined or is in Rel 1.1 format

---

NHRP.212

Level: UE_ERROR

Short Syntax: NHRP.212 *packet_type* not rcvd; cannot snd *packet_type* pkt; inbound/outbound rqst-id *inbound_request_id/ outbound_request_id*

Long Syntax: NHRP.212 *packet_type* not received; therefore, cannot send *packet_type* for inbound/outbound Request-ID *inbound_request_id/ outbound_request_id*

Description: Cannot send packet_type reply because corresponding reply not received

Cause: NHRP or MPOA is disabled or misconfigured in one of the routers along the routed path

Action: Find where the packet is being dropped

NHRP.213

Level: C_INFO

Short Syntax: NHRP.213 MPOA 1483 cntrl VC dwn reason= *reason_code*, cause= *cause_code*

Long Syntax: NHRP.213 MPOA 1483 control VCC down reason= *reason_code*, cause= *cause_code*

Description: External MPS or MPC brought down VCC

NHRP.214

Level: UE_ERROR

Short Syntax: NHRP.214 Could not xmit pkt to *atm_addr*, out net intf *net_number*

Long Syntax: NHRP.214 Could not transmit NHRP packet to *atm_addr*, out network interface *net_number*

Description: Could not transmit MPOA/NHRP packet to MPC/MPS

Cause: VCC to atm_addr has not become active

Action: Check status of external device identified with atm_addr

---

NHRP.215

Level: C_INFO

Short Syntax: NHRP.215 *ip_addr* is LEC on net intf *net_number*, but it's not an MPC/MPS

Long Syntax: NHRP.215 *ip_addr* is LEC on network interface *net_number*, but it's not an MPC/MPS

Description: The ip_addr on the ELAN associated with net_number is not an MPC/MPS

Cause: MPOA is not enabled/supported on one side or the other

Action: No action required if NHRP packet is being received at ip_addr; however, if the LEC associated with ip_addr is an MPC or MPS, check configuration of both sides

---

NHRP.216

Level: C_INFO

Short Syntax: NHRP.216 Add purge entry for dest_addr= *destination_addr*, prefix= *prefix*, nh= *next_hop_addr*

Long Syntax: NHRP.216 Adding new purge cache entry for destination address= *destination_addr*, prefix= *prefix* and nh= *next_hop_addr*

Description: Adding new purge cache entry

NHRP.217

Level: C_INFO

Short Syntax: NHRP.217 Purge Cache: *comment* nhrp_client= *nhrp_client_addr*, for dest_addr= *next_hop_addr*, nh=

Long Syntax: NHRP.217 Purge Cache info: *comment* nhrp client= *nhrp_client_addr*, for destination address= *next_hop_addr* and nh=

Description: Purge Cache information.

NHRP.218

Level: C_INFO

Short Syntax: NHRP.218 Purge Cache: *comment* MPC for dest_addr= *nhrp_client_addr*, nh= *next_hop_addr*

Long Syntax: NHRP.218 Purge Cache info: *comment* MPC for destination address= *nhrp_client_addr* and nh= *next_hop_addr*

Description: Purge Cache information.

NHRP.219

Level: C_INFO

Short Syntax: NHRP.219 Cache Imp *msg*: rid= *reqid*, dest= *dest_addr*, pfx= *prefix*, cid= *cacheid*, ht= *holding_time*, nt= *netno*

Long Syntax: NHRP.219 Cache Imposition *msg*: reqid= *reqid*, dest= *dest_addr*, prefix= *prefix*, cacheid= *cacheid*, htime= *holding_time* net= *netno*

Description: MPOA Cache Imposition Request/Reply

NHRP.220

Level: P_TRACE

Short Syntax: NHRP.220 Trace MPOA KeepAlive pkt.

Long Syntax: NHRP.220 Trace MPOA KeepAlive pkt.

Description: MPOA KeepAlive control frame packet tracing.

NHRP.221

Level: C_INFO

Short Syntax: NHRP.221 rcvd res reqst from *src_net_addr/ src_node_addr* for *dest_net_addr/ dest_node_addr*

Long Syntax: NHRP.221 received resolution request from *src_net_addr/ src_node_addr* for *dest_net_addr/ dest_node_addr*

Description: NHRP Server received a Resolution Request

NHRP.222

Level: C_INFO

Short Syntax: NHRP.222 xmit purge pkt to client= *cli_net_addr/ cli_node_addr* for dest_addr= *dest_net_addr/ dest_node_addr* w/ prefix= *prefix*

Long Syntax: NHRP.222 Send purge pkt to client= *cli_net_addr/ cli_node_addr* for destination= *dest_net_addr/ dest_node_addr* with prefix= *prefix*

Description: Purge Packet transmit information.

NHRP.223

Level: C_INFO

Short Syntax: NHRP.223 *function_name*: *general_message proto_net_addr/ proto_node_addr*

Long Syntax: NHRP.223 *function_name*: *general_message proto_net_addr/ proto_node_addr*

Description: The message is the description.

NHRP.224

Level: C_INFO

Short Syntax: NHRP.224 *function_name*: *general_message pr_net_ad1/ pr_node_ad1 pr_net_ad2/ pr_node_ad2 pr_net_ad3/ pr_node_ad3*

Long Syntax: NHRP.224 *function_name*: *general_message pr_net_ad1/ pr_node_ad1 pr_net_ad2/ pr_node_ad2 pr_net_ad3/ pr_node_ad3*

Description: The message is the description.

NHRP.225

Level: C_INFO

Short Syntax: NHRP.225 Exclude lst match for: *pr_net_addr/ pr_node_addr*

Long Syntax: NHRP.225 Exclude list match for: *pr_net_addr/ pr_node_addr*

Description: NHRP cannot process all or part of the NHRP packet because there is an IPX address in the packet that matches one that is configured in the NHRP exclude list.

NHRP.227

Level: UE_ERROR

Short Syntax: NHRP.227 *function_name*: *general_message proto_net_addr/ proto_node_addr*

Long Syntax: NHRP.227 *function_name*: *general_message proto_net_addr/ proto_node_addr*

Description: The message is the description.

NHRP.228

Level: C_INFO

Short Syntax: NHRP.228 *caller_string* free learp mac-rd elem for nxt hp addr= *prot_net_addr/ prot_node_addr*

Long Syntax: NHRP.228 *caller_string* free learp mac-rd element for next hop addr= *prot_net_addr/ prot_node_addr*

Description: Free LEARP_MAC_RD_ENTRY for the specified protocol address.

NHRP.229

Level: UE_ERROR

Short Syntax: NHRP.229 Could not delete Imp Cache entry for dest= *proto_net_addr/ proto_node_addr*, pfx= *prefix*, cid= *cacheid*

Long Syntax: NHRP.229 Could not delete Imposition Cache entry for dest= *proto_net_addr/ proto_node_addr*, prefix= *prefix*, cacheid= *cacheid*

Description: Deleting an Imposition Cache entry for e-mpc initiated purge failed

---

NHRP.230

Level: UE_ERROR

Short Syntax: NHRP.230 *caller_string* dtctd dst unreachable to *proto_net_addr/ proto_node_addr*

Long Syntax: NHRP.230 *caller_string* detected destination unreachable to *proto_net_addr/ proto_node_addr*

Description: The caller has no route to the specified destination.

NHRP.231

Level: C_INFO

Short Syntax: NHRP.231 NHS sending a *reply_type* to *src_proto_net_addr/ src_proto_node_addr*

Long Syntax: NHRP.231 NHRP Server sending a *reply_type* to *src_proto_net_addr/ src_proto_node_addr*

Description: NHRP Server is sending a the specified reply to the specified client.

NHRP.233

Level: U_INFO

Short Syntax: NHRP.233 *caller_string* rcvd err rc from IPX Route Table for *ipx_net_addr/ ipx_node_addr*

Long Syntax: NHRP.233 *caller_string* received error return code from IPX Routing Table for *ipx_net_addr/ ipx_node_addr*

Description: Notification that there is a ipx routing error.

NHRP.234

Level: C_INFO

Short Syntax: NHRP.234 *caller_string* rcvd a req for *ipx_net_addr/ ipx_node_addr* which is one of our proto addr

Long Syntax: NHRP.234 *caller_string* received a request *ipx_net_addr/ ipx_node_addr*, which is one of our protocol address

Description: We received a request to shortcut to one of our protocol addresses.

NHRP.235

Level: C_INFO

Short Syntax: NHRP.235 *ipx_net_addr/ ipx_node_addr* is LEC on net intf *net_number*, but it's not an MPC/MPS

Long Syntax: NHRP.235 *ipx_net_addr/ ipx_node_addr* is LEC on network interface *net_number*, but it's not an MPC/MPS

Description: The ipx_addr on the ELAN associated with net_number is not an MPC/MPS

Cause: MPOA is not enabled/supported on one side or the other

Action: No action required if NHRP packet is being received at ipx_addr; however, if the LEC associated with ipx_addr is an MPC or MPS, check configuration of both sides

---

NHRP.236

Level: C_INFO

Short Syntax: NHRP.236 Add purge entry for dest_net= *dest_net_addr*, nh= *nh_net_addr/ nh_node_addr*

Long Syntax: NHRP.236 Adding new purge cache entry for dest net= *dest_net_addr* and nh= *nh_net_addr/ nh_node_addr*

Description: Adding new purge cache entry

NHRP.237

Level: C_INFO

Short Syntax: NHRP.237 Cache Imp *msg*: rid= *reqid*, dest= *dest_net_addr/ dest_node_addr*, pfx= *prefix*, cid= *cacheid*, ht= *holding_time*, nt= *netno*

Long Syntax: NHRP.237 Cache Imposition *msg*: reqid= *reqid*, dest= *dest_net_addr/ dest_node_addr*, prefix= *prefix*,

cacheid= *cacheid*, htime= *holding_time* net= *netno*

Description: MPOA Cache Imposition Request/Reply

NHRP.238

Level: UI_ERROR

Short Syntax: NHRP.238 *caller* can't add purge entry for dest_net= *dest_net_addr*, nh= *nh_net_addr/ nh_node_addr*

Long Syntax: NHRP.238 *caller* can't Add new purge cache entry for dest net= *dest_net_addr* and nh= *nh_net_addr/ nh_node_addr*

Description: The above routine can't add new purge cache entry

---

NHRP.239

Level: C_INFO

Short Syntax: NHRP.239 *function_name*: *general_message general_code general_message general_code*

Long Syntax: NHRP.239 *function_name*: *general_message general_code general_message general_code*

Description: The message is the description.

Chapter 86. Open Shortest Path First (OSPF)

This chapter describes Open Shortest Path First (OSPF) messages. For information on message content and how to use the message, refer to the Introduction.

SPF.001

Level: UE-ERROR

Short Syntax: SPF.001 Bad length pkt, from *IP_source*, to *IP_destination*, OSPF len *OSPF_packet_length*, IP len *IP_packet_length*, type *OSPF_packet_type*

Long Syntax: SPF.001 Bad length packet, from *IP_source*, to *IP_destination*, OSPF *OSPF_packet_length*, IP *IP_packet_length*, type *OSPF_packet_type*

Description: An OSPF packet has been received. The OSPF length field indicates a longer packet than indicated by the IP header length field. The packet is discarded.

SPF.002

Level: UE-ERROR

Short Syntax: SPF.002 Bad pkt checksum, from *IP_source*, type *OSPF_packet_type*

Long Syntax: SPF.002 Bad packet checksum, from *IP_source*, type *OSPF_packet_type*

Description: An OSPF packet has been received. The packet has an invalid OSPF checksum. The packet is discarded.

SPF.003

Level: UE-ERROR

Short Syntax: SPF.003 Bad OSPF version, from *IP_source*, type *OSPF_packet_type*

Long Syntax: SPF.003 Bad OSPF version, from *IP_source*, type *OSPF_packet_type*

Description: An OSPF packet has been received. The version field in the OSPF header is not equal to 1. The packet is discarded.

SPF.005

Level: UE-ERROR

Short Syntax: SPF.005 No matching ifc for pkt from *IP_source*, type *OSPF_packet_type*

Long Syntax: SPF.005 No matching SPF-interface for packet from *IP_source*, type *OSPF_packet_type*

Description: An OSPF packet has been received. Either the IP destination specified in the packet is not acceptable, or the parameters in the OSPF header (like area ID) do not match the parameters configured for the receiving interface. The packet is discarded.

SPF.006

Level: UE-ERROR

Short Syntax: SPF.006 Authentication failure, from *IP_source*, type *OSPF_packet_type*

Long Syntax: SPF.006 Packet authentication failure, from *IP_source*, type *OSPF_packet_type*

Description: An OSPF packet has been received which fails to authenticate. The packet is discarded.

SPF.007

Level: UE-ERROR

Short Syntax: SPF.007 No matching nbr for pkt from *IP_source*, type *OSPF_packet_type*

Long Syntax: SPF.007 No matching OSPF neighbor for packet from *IP_source*, type *OSPF_packet_type*

Description: An OSPF packet has been received. The packet is not a hello packet, and does not match any existing OSPF neighbor. The packet is discarded.

SPF.008

Level: UE-ERROR

Short Syntax: SPF.008 Bad pkt type from *IP_source*, type *OSPF_packet_type*

Long Syntax: SPF.008 Bad packet type received from *IP_source*, type *OSPF_packet_type*

Description: An OSPF packet has been received. The OSPF packet type field is invalid. The packet is discarded.

SPF.009

Level: UI-ERROR

Short Syntax: SPF.009 No buffer for mcast to *IP_destination*

Long Syntax: SPF.009 No buffer for multicast packet to *IP_destination*

Description: An attempt was made to send a multicast packet on a non-broadcast network by expanding the packet on the link level. This expansion failed due to insufficient buffer resources.

Cause: Resource congestion

Action: Alleviate congestion

SPF.010

Level: P-TRACE

Short Syntax: SPF.010 Received packet type *OSPF_packet_type* from *IP_source*

Long Syntax: SPF.010 Received packet type *OSPF_packet_type* from *IP_source*

Description: An OSPF packet of the specified type was received.

SPF.011

Level: U-TRACE

Short Syntax: SPF.011 Sending unicast type *OSPF_packet_type* dst *IP_destination*

Long Syntax: SPF.011 Sending unicast type *OSPF_packet_type* dst *IP_destination*

Description: Unicast OSPF packet of specified type has been sent to the specified IP destination.

SPF.012

Level: P-TRACE

Short Syntax: SPF.012 Sending mcast type *OSPF_packet_type*, dst *IP_destination* net *network*

Long Syntax: SPF.012 Sending multicast, type *OSPF_packet_type*, destination *IP_destination* net *network*

Description: Multicast OSPF packet of specified type sent out specified interface.

SPF.013

Level: U-INFO

Short Syntax: SPF.013 Rxmitting type *OSPF_packet_type*, *IP_source -> IP_destination*

Long Syntax: SPF.013 Retransmitting packet, type *OSPF_packet_type*, *IP_source -> IP_destination*

Description: Unicast OSPF packet of specified type is being retransmitted.

SPF.014

Level: UI-ERROR

Short Syntax: SPF.014 No FSM match, ifc *interface_IP_address*, state *interface_state*, event *interface_event*

Long Syntax: SPF.014 No FSM match, interface *interface_IP_address*, state *interface_state*, event *interface_event*

Description: The specified event occurred while an interface was in the specified state. This occurrence was not covered by the interface Finite State Machine. The event is ignored.

Cause: Possible internal error

Action: Notify service

SPF.015

Level: U-INFO

Short Syntax: SPF.015 State change, ifc *interface_IP_address*, new state *new_interface_state*, event *interface_event*

Long Syntax: SPF.015 State change, interface *interface_IP_address*, new state *new_interface_state*, event *interface_event*

Description: The specified event occurred on the specified interface, causing its state to transition.

SPF.016

Level: UE-ERROR

Short Syntax: SPF.016 No match for hlo (virtual link) from *IP_source*

Long Syntax: SPF.016 No match for hello received on virtual link, from *IP_source*

Description: A hello packet was received that could only match a virtual link, yet that link is not configured. The packet is discarded.

SPF.017

Level: UE-ERROR

Short Syntax: SPF.017 Network mask mismatch with *IP_source*

Long Syntax: SPF.017 Network mask mismatch in hello from *IP_source*

Description: Hello packet received from neighbor. Neighbor disagrees with this router concerning the network mask of their common network. The packet is discarded.

SPF.018

Level: UE-ERROR

Short Syntax: SPF.018 Hello interval mismatch with *IP_source*

Long Syntax: SPF.018 Hello interval mismatch in hello from *IP_source*

Description: Hello packet received from neighbor. Neighbor disagrees with this router concerning the hello interval to be used on the common network. The packet is discarded.

SPF.019

Level: UE-ERROR

Short Syntax: SPF.019 Dead interval mismatch with *IP_source*

Long Syntax: SPF.019 Dead interval mismatch in hello from *IP_source*

Description: Hello packet received from neighbor. Neighbor disagrees with this router concerning the ''dead router interval'' to be used on the common network. The packet is discarded.

SPF.020

Level: UI-ERROR

Short Syntax: SPF.020 No FSM match, nbr *neighbor_IP_address*, state *neighbor_state*, event *neighbor_event*

Long Syntax: SPF.020 No FSM match, neighbor *neighbor_IP_address*, state *neighbor_state*, event *neighbor_event*

Description: The specified event has been generated for the specified neighbor, which is currently in the specified state. This was not anticipated by the neighbor Finite State Machine. The event is ignored.

Cause: Possible internal error

Action: Notify service

SPF.021

Level: U-INFO

Short Syntax: SPF.021 State change, nbr *neighbor_IP_address*, new state *neighbor_state*, event *neighbor_event*

Long Syntax: SPF.021 State change, neighbor *neighbor_IP_address*, new state *neighbor_state*, event *neighbor_event*

Description: The specified event has been generated, causing the specified neighbor to transfer to a new state.

SPF.022

Level: UI-ERROR

Short Syntax: SPF.022 Outstanding DD pkt not avail for nbr *neighbor_IP_address*

Long Syntax: SPF.022 Outstanding Database Description packet not avail for neighbor *neighbor_IP_address*

Description: An attempt was made to retransmit a Database Description packet to the specified neighbor, but the packet could not be found. Retransmission is aborted.

Cause: Possible internal error

Action: Notify service

SPF.023

Level: UI-ERROR

Short Syntax: SPF.023 Unable to get pkt, to *IP_destination*, ifc *interface_IP_address*

Long Syntax: SPF.023 Unable to get packet to send to *IP_destination*, out interface *interface_IP_address*

Description: An attempt was made to send an OSPF packet to the specified destination. The specified interface has been aborted due to lack of buffers.

Cause: Resource congestion

Action: Alleviate congestion

SPF.024

Level: UE-ERROR

Short Syntax: SPF.024 Bad length LS adv from *neighbor_IP_address*

Long Syntax: SPF.024 Bad length Link state advertisement received from *neighbor_IP_address*

Description: A link state advertisement has been received from the specified neighbor, and the advertisement's length field indicates that the entire advertisement is NOT fully contained in the received Link State Update Packet. The partial advertisement is discarded.

SPF.025

Level: UE-ERROR

Short Syntax: SPF.025 from *neighbor_IP_address*, adv. cksum fl: (*LS_type*, *advertisement_ID*)

Long Syntax: SPF.025 from *neighbor_IP_address*, LS advertisement checksum fails: LS type *LS_type* id *advertisement_ID*

Description: A link state advertisement has been received. The advertisement is identified by its LS type and two-part originating ID (see OSPF specification section 12.1). The checksum field contained in the advertisement is invalid. The advertisement is ignored.

SPF.026

Level: UE-ERROR

Short Syntax: SPF.026 from *neighbor_IP_address*, bad type, adv: (*LS_type*, *advertisement_ID*)

Long Syntax: SPF.026 from *neighbor_IP_address*, bad LS type, advertisement: typ *LS_type* id *advertisement_ID*

Description: A link state advertisement has been received. The advertisement's LS type field is invalid.

The advertisement is ignored.

SPF.027

Level: UE-ERROR

Short Syntax: SPF.027 from *neighbor_IP_address*, ext adv on VL: (*LS_type*, *advertisement_ID*)

Long Syntax: SPF.027 from *neighbor_IP_address*, AS external link adv. on Virtual Link: typ *LS_type* id *advertisement_ID*

Description: A link state advertisement has been received. It was received over a virtual link, yet its LS type is equal to AS external link. The advertisement is ignored.

SPF.028

Level: U-INFO

Short Syntax: SPF.028 from *neighbor_IP_address*, old adv: (*LS_type*, *advertisement_ID*)

Long Syntax: SPF.028 from *neighbor_IP_address*, old LS advertisement: typ *LS_type* id *advertisement_ID*

Description: A link state advertisement has been received. The advertisement is older than the current database copy. The received advertisement will be reflooded toward the originator.

SPF.029

Level: U-INFO

Short Syntax: SPF.029 from *neighbor_IP_address*, self update: (*LS_type*, *advertisement_ID*)

Long Syntax: SPF.029 from *neighbor_IP_address*, self update: typ *LS_type* id *advertisement_ID*

Description: A link state advertisement has been received. The advertisement was originated by the router itself, yet is newer than the database copy. This indicates that it originated before the router was last started. This causes the router to either advance the LS sequence number and originate a new instantiation of the advertisement, or flush the advertisement, if it's a summary LSA and the attached area does not wish to import summary LSAs anymore.

SPF.030

Level: U-INFO

Short Syntax: SPF.030 from *neighbor_IP_address*, new adv: (*LS_type*, *advertisement_ID*)

Long Syntax: SPF.030 from *neighbor_IP_address*, new LS advertisement: typ *LS_type* id *advertisement_ID*

Description: A link state advertisement has been received. The advertisement is newer than the current database copy. This advertisement is flooded out all other interfaces, and installed in the routing database.

SPF.031

Level: U-INFO

Short Syntax: SPF.031 from *neighbor_IP_address*, Old ack for adv: (*LS_type*, *advertisement_ID*)

Long Syntax: SPF.031 from *neighbor_IP_address*, Old acknowledgement for advertisement: typ *LS_type* id *advertisement_ID*

Description: An unexpected link state acknowledgement has been received. The acknowledgement, however, is for a previous instantiation of the link state advertisement.

SPF.032

Level: U-INFO

Short Syntax: SPF.032 Bad ack from *neighbor_IP_address* for adv: (*LS_type*, *advertisement_ID*)

Long Syntax: SPF.032 Bad acknowledgment from *neighbor_IP_address* for advertisement: typ *LS_type* id *advertisement_ID*

Description: An unexpected link state acknowledgement has been received. The acknowledgement however is for the current instantiation of the link state advertisement.

SPF.033

Level: U-INFO

Short Syntax: SPF.033 LS update retransmission to *neighbor_IP_address*

Long Syntax: SPF.033 LS update retransmission to neighbor *neighbor_IP_address*

Description: A Link State Update packet containing retransmitted link state advertisements has been unicast to the specified neighbor. This probably indicates packet loss during the flooding procedure.

SPF.034

Level: U-INFO

Short Syntax: SPF.034 LS ack sent direct to *neighbor_IP_address*

Long Syntax: SPF.034 LS acknowledement sent directly to neighbor *neighbor_IP_address*

Description: A Link State Acknowledgement packet has been sent directly to the specified neighbor. This is in response to duplicate link state advertisements received from the neighbor. This probably indicates packet loss during the flooding procedure.

SPF.035

Level: U-INFO

Short Syntax: SPF.035 Flushing advertisement: (*LS_type*, *advertisement_ID*)

Long Syntax: SPF.035 Flushing advertisement: typ *LS_type* id *advertisement_ID*

Description: A link state advertisement contained in the link state database has not been refreshed for 2 hours. The advertisement is deleted from the database. This probably indicates that the originator of the advertisement is unreachable. See section 14 of the OSPF specification.

SPF.036

Level: U-INFO

Short Syntax: SPF.036 Originating adv: (*LS_type*, *advertisement_ID*)

Long Syntax: SPF.036 Originating LS advertisement: typ *LS_type* id *advertisement_ID*

Description: A link state advertisement is being (re)originated by the router. This can be due to topological change, or the necessity to refresh.

SPF.037

Level: U-INFO

Short Syntax: SPF.037 new route to *destination*, type *route_type* cost *route_cost*

Long Syntax: SPF.037 New route to destination *destination*, type *route_type* cost *route_cost*

Description: The SPF routing table build process has detected a new best route to specified destination, having the specified cost.

SPF.038

Level: P-TRACE

Short Syntax: SPF.038 *Interface* hello sent to dest *type*

Long Syntax: SPF.038 *Interface* hello sent to IP destination *type*

Description: An OSPF hello has been sent to the specified IP destination. This has been done over an NBMA (Non-Broadcast Multi-Access) network or P2MP (Point-to-Multi-Point) Network.

SPF.039

Level: U-INFO

Short Syntax: SPF.039 The OSPF routing protocol is *en/dis*abled

Long Syntax: SPF.039 The OSPF routing protocol is *en/dis*abled

Description: Printed on router startup. Indicates operational status of the SPF protocol.

SPF.040

Level: U-INFO

Short Syntax: SPF.040 SPF Interface *interface_IP_address* is not an IP address, Interface not installed

Long Syntax: SPF.040 SPF Interface *interface_IP_address* is not an IP address, Interface not installed

Description: Printed on router startup when an OSPF interface address is configured, yet this address has not also been configured in the IP console. OSPF interface is not installed.

SPF.043

Level: U-INFO

Short Syntax: SPF.043 Duplicate LS ack received from *neighbor_IP_address*

Long Syntax: SPF.043 Duplicate LS acknowledgment received from neighbor *neighbor_IP_address*

Description: Unexpected link state acknowledgements have been received from the specified neighbor. This probably indicates packet loss during the flooding procedure.

SPF.044

Level: UE-ERROR

Short Syntax: SPF.044 from *neighbor_IP_address*, bad age field, adv (*LS_type*, *advertisement_ID*)

Long Syntax: SPF.044 from *neighbor_IP_address*, bad age field, advertisement: typ *LS_type* id *advertisement_ID*

Description: A link state advertisement has been received. The advertisement's LS age field is invalid. The advertisement is ignored.

SPF.045

Level: U-INFO

Short Syntax: SPF.045 non-existent transit area *proposed_transit_area*, VL discarded

Long Syntax: SPF.045 Transit area *proposed_transit_area* not configured, virtual link discarded

Description: A virtual link has been configured to have a certain transit area, yet that area has not been defined. The virtual link is ignored.

SPF.046

Level: U-INFO

Short Syntax: SPF.046 No backbone configured, VLs discarded

Long Syntax: SPF.046 Backbone area is not configured, all virtual links discarded

Description: Virtual links cannot be used unless a backbone area is configured.

SPF.047

Level: U-INFO

Short Syntax: SPF.047 *destination* now unreachable

Long Syntax: SPF.047 Destination *destination* now unreachable

Description: The destination has been found to be unreachable during the routing table build process.

SPF.048

Level: UE-ERROR

Short Syntax: SPF.048 AS ext adv limit exceeded; adv ignored

Long Syntax: SPF.048 Limit of AS external advertisements exceeded; advertisement discarded

Description: The estimated number of advertisements has been exceeded. New AS external advertisements are ignored in order to put a limit on router heap usage.

SPF.050

Level: U-INFO

Short Syntax: SPF.050 from *neighbor_IP_address*, MaxAge: (*LS_type*, *advertisement_ID*)

Long Syntax: SPF.050 from *neighbor_IP_address*, received unexpected MaxAge: typ *LS_type* id *advertisement_ID*

Description: A link state advertisement has been received. Its age is MaxAge, and there is no current instantiation of the advertisement in the router's database. The advertisement is acknowledged and then discarded without flooding.

SPF.051

Level: UE-ERROR

Short Syntax: SPF.051 bad adv/ovflo: (*LS_type*, *advertisement_ID*)

Long Syntax: SPF.051 error in advertisement or routing overflow: typ *LS_type* id *advertisement_ID*

Description: A link state advertisement has been received. The advertisement contains an error, or cannot be added to the database due to routing table overflow. In any case, the advertisement is discarded.

SPF.052

Level: UE-ERROR

Short Syntax: SPF.052 Stub area mismatch with *IP_source*

Long Syntax: SPF.052 Stub area mismatch in hello from *IP_source*

Description: Hello packet received from neighbor. Neighbor disagrees with this router concerning the attached area's ability to process AS external link advertisements. Hello packet is ignored.

SPF.053

Level: UE-ERROR

Short Syntax: SPF.053 from *neighbor_IP_address*, recvd in stub area, adv (*LS_type*, *advertisement_ID*)

Long Syntax: SPF.053 from *neighbor_IP_address*, type 5 LSA in stub area, adv: typ *LS_type* id *advertisement_ID*

Description: A type 5 link state advertisement has been received. The advertisement is being flooded through a stub area, and is therefore ignored.

SPF.054

Level: C-INFO

Short Syntax: SPF.054 Dijkstra calculation performed: *Number_areas* area(s)

Long Syntax: SPF.054 Dijkstra calculation performed, on *Number_areas* area(s)

Description: As a result of a topology change, the routing table has been recalculated, starting with the Dijkstra calculation.

SPF.055

Level: U-INFO

Short Syntax: SPF.055 Network LSA w/ old Adv Rtr: (*LS_type*, *advertisement_ID*)

Long Syntax: SPF.055 Network LSA with old Advertising Router: (*LS_type*, *advertisement_ID*)

Description: A network links advertisement having one of our addresses as Link State ID, but whose Advertising Router is not our Router ID, has been received. These advertisements are flushed, as they are assumed to be out-of-date.

SPF.056

Level: U-INFO

Short Syntax: SPF.056 Reparsing Network LSA: *Link_State_ID*

Long Syntax: SPF.056 Reparsing Network LSA: *Link_State_ID*

Description: A network link is being reparsed, owing to the fact that there are multiple network-LSAs in the network with the same Link State ID. This indicates that a router has changed OSPF Router IDs, and has originated the same router-LSA before and after the change. This is a normal, but rare, event.

SPF.057

Level: UI-ERROR

Short Syntax: SPF.057 Send unicast type *OSPF_packet_type* dst *IP_destination* fld, rsn *reason_code*, net *network*

Long Syntax: SPF.057 Sending unicast type *OSPF_packet_type* dst *IP_destination* failed, reason *reason_code*, network *network*

Description: Sending of a unicast OSPF packet of specified type failed to the specified IP destination. The reason_code is the internal error code for the failure.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

SPF.058

Level: UI-ERROR

Short Syntax: SPF.058 Send multicast type *OSPF_packet_type* dst *IP_destination* fld, rsn *reason_code*, net *network*

Long Syntax: SPF.058 Sending multicast type *OSPF_packet_type* dst *IP_destination* failed, reason *reason_code*, network *network*

Description: Sending of a multicast OSPF packet of specified type failed to the specified IP destination. The reason_code is the internal error code for the failure.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

SPF.059

Level: UI-ERROR

Short Syntax: SPF.059 Rxmit type *OSPF_packet_type* fld, *IP_source -> IP_destination*, rsn *reason_code*, net *network*

Long Syntax: SPF.059 Retransmitting packet failed, type *OSPF_packet_type*, *IP_source -> IP_destination*, reason *reason_code*, network *network*

Description: Retransmission of unicast OSPF packet of specified type failed. The reason_code is the internal error code for the failure.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

SPF.060

Level: UI-ERROR

Short Syntax: SPF.060 NBMA hello disc to dest *neighbor_IP_address*, rsn *reason_code*, net *network*

Long Syntax: SPF.060 NBMA hello disc to IP

destination *neighbor_IP_address*, reason *reason_code*, network *network*

Description: An OSPF hello has was discarded when attempting to send to the specified IP destination. This was attempted over a non-broadcast, multi-access interface. The reason_code is the internal error code for the failure.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

SPF.061

Level: U-INFO

Short Syntax: SPF.061 non-existent area *proposed_area*, interface *interface_IP_address* not installed

Long Syntax: SPF.061 OSPF area *proposed_area* not configured, interface *interface_IP_address* not installed

Description: Printed on router startup when an OSPF interface address is configured, but the attached area is not defined. Hence, the OSPF interface is not installed.

SPF.062

Level: UE-ERROR

Short Syntax: SPF.062 LS node alloc fld, ty *ls_node_type*, sz *ls_node_size*

Long Syntax: SPF.062 LS node allocation failed, type *ls_node_type*, size *ls_node_size*

Description: The router has run out of memory. As a result, OSPF is unable to allocate a node for later insertion into the link state database for either the advertisement that we have just received, or the advertisement that we are trying to originate.

SPF.063

Level: UI-ERROR

Short Syntax: SPF.063 No FSM match, state *interface_state*, event *interface_event*, unnum net *network ID*

Long Syntax: SPF.063 No FSM match, state *interface_state*, event *interface_event*, unnumbered net *network ID*

Description: The specified event occurred while an unnumbered interface was in the specified state. This occurrence was not covered by the interface Finite State Machine. The event is ignored.

Cause: Possible internal error

Action: Notify service

SPF.064

Level: U-INFO

Short Syntax: SPF.064 State change, new state *new_interface_state*, event *interface_event*, unnum net *network ID*

Long Syntax: SPF.064 State change, new state *new_interface_state*, event *interface_event*, unnumbered net *network ID*

Description: The specified event occurred on the specified interface, causing its state to transition.

SPF.066

Level: U-INFO

Short Syntax: SPF.066 SPF IP mismatch for unnum addr *interface_IP_address*, Interface not installed

Long Syntax: SPF.066 SPF IP mismatch for unnumbered address *interface_IP_address*, Interface not installed

Description: Printed on router startup when an unnumbered OSPF interface address is configured, yet this address has not also been configured in the IP console. OSPF interface is not installed.

SPF.067

Level: U-INFO

Short Syntax: SPF.067 DMD circuit support active for area *active_area*

Long Syntax: SPF.067 Demand circuit support active for area *active_area*

Description: Printed when there are no more DC bit clear LSA in any of the area's link state database's and it is valid to set the DoNotAge bit.

SPF.068

Level: U-INFO

Short Syntax: SPF.068 DMD circuit support not active for area *inactive_area*

Long Syntax: SPF.068 Demand circuit support not active for area *inactive_area*

Description: Printed when an LSA with the DC bit clear is added to one of the area's link state data bases and any LSA's with the DoNotAge bit set are purged.

SPF.069

Level: P-TRACE

Short Syntax: SPF.069 Unchanged adv: (*LS_type*, *advertisement_ID*) suppr for dmd int

Long Syntax: SPF.069 Unchanged advertisement: typ *LS_type* id *advertisement_ID* suppressed for demand interfaces

Description: Printed when an LSA is not flooded over one or more circuits configured as demand circuits because there is no change in the content of the LSA from a previous version.

SPF.070

Level: P-TRACE

Short Syntax: SPF.070 Hello's on int *interface_address* to neigh *neighbor_address.* suppressed

Long Syntax: SPF.070 Hello's on interface *interface_address* to neighbor *neighbor_address.* are being suppressed.

Description: Printed when hello suppression becomes active for some interface and neighbor.

SPF.071

Level: P-TRACE

Short Syntax: SPF.071 Cbit clear indicate received in area *area_address* from *router_id.*

Long Syntax: SPF.071 Cbit clear indicate LSA received in area *area_address* from *router_id.*.

Description: Printed when a special type 4 indicate LSA is received in a non stub area to indicate the presence of routers outside the area that do not support DoNotAge processing.

SPF.072

Level: P-TRACE

Short Syntax: SPF.072 Cbit clear indicate originated in area *area_address*

Long Syntax: SPF.072 Cbit clear indicate LSA originated in area *area_address*.

Description: Printed when the local router originates a special type 4 indicate LSA to indicate the presence of routers outside the area that do not support DoNotAge processing.

SPF.073

Level: UE-ERROR

Short Syntax: SPF.073 adv discarded, ovflo buf: (*LS_type*, *advertisement_ID*)

Long Syntax: SPF.073 advertisement discarded, overflows buffer: LS type *LS_type* id *advertisement_ID*

Description: A link state advertisement was discarded because it would be too large to fit in a routers data area.

Cause: A router links lsa has become excessively large due to a large number of direct connections to neighbor routers within a single area.

Action: Reconfigure the network to reduce the size of the largest link state advertisement or increase the size of router data areas to hold the lsa. The data area used to build lsa's can be increased to the size of the local router's buffer by configuring the maximum lsa size. The size of the local router's buffer can be enlarged by increasing the size of the largest mtu for a locally attached subnetwork.

SPF.074

Level: UE-ERROR

Short Syntax: SPF.074 Demand circuit *Advertisement_scope* LSA purge error - Area *area_id* count is *area_do_not_age_count*.

Long Syntax: SPF.074 Demand circuit *Advertisement_scope* LSA purge error - Area *area_id* count is *area_do_not_age_count*.

Description: A mismatch in the number of advertisements purged due to a change of status from the area supporting demand circuits to not supporting demand circuits.

SPF.075

Level: UE-ERROR

Short Syntax: SPF.075 DD pkt MTU mismatch for *Neighbor_Address* - ifc MTU *Interface_MTU*, ifc MRU *Interface_MRU*, nbr MTU *Neighbor*

Long Syntax: SPF.075 Database Description paket MTU mismatch for *Neighbor_Address* - interface MTU/MRU *Interface_MTU*/ *Interface_MRU* versus neighbor MTU *Neighbor*.

Description: A received data description packet was discarded due to an MTU mismatch with the advertised MTU and the interface MTU.

Cause: The MTU specified in the OSPF Database Description packet is larger than the interface MTU.

Action: Reconfigure all OSPF routers in the IP subnet to have the same interface MTU.

SPF.076

Level: UE-ERROR

Short Syntax: SPF.076 OSPF subsystem reset cannot increase heap from *Current_heap_allocation* to *Requested_heap_allocation*.

Long Syntax: SPF.076 OSPF subsystem reset cannot increase heap reservation from *Current_heap_allocation* to *Requested_heap_allocation*.

Description: An OSPF reset attempted to increase the heap allocation for OSPF.

Action: The router must be re-started to increase the OSPF heap reservation.

SPF.077

Level: UE-ERROR

Short Syntax: SPF.077 OSPF subsystem reset memory alloc failure for *Object_type - Object_id*.

Long Syntax: SPF.077 OSPF subsystem reset memory allocation failure for object *Object_type - Object_id*.

Description: An OSPF reset tried to allocate an object but failed. Check memory allocation of for other router protocols and features.

SPF.078

Level: C_INFO

Short Syntax: SPF.078 0x *Memory_op Memory_address* Length *Memory*.

Long Syntax: SPF.078 0x *Memory_op Memory_address* for length *Memory*.

Description: OSPF allocated or freed temporary memory.

SPF.079

Level: C_INFO

Short Syntax: SPF.079 LS Update (*LSA_type*, *LSA_id*, *LSA_org*) unicast to *Neighbor_ID*.

Long Syntax: SPF.079 LS Update (type *LSA_type*, id *LSA_id*, org *LSA_org*) unicast to *Neighbor_ID*.

Description: A more recent LSA was sent back toward the originator as per RFC 2178.

SPF.080

Level: U-INFO

Short Syntax: SPF.080 from *neighbor_IP_address*, MINLSARRIVAL reject: (*LS_type*, *advertisement_ID*, *new_lsa_seq*) seq *old_lsa_seq lsa_age* versus *lsa_received*.

Long Syntax: SPF.080 from *neighbor_IP_address*, MINLSARRIVAL reject: type *LS_type* id *advertisement_ID* seq *new_lsa_seq* - old seq *old_lsa_seq lsa_age* versus *lsa_received*.

Description: A link state advertisement has been received within the MINLSARRIVAL. It will be ignored.

Chapter 87. OS Interface to Router IP (IPIF)

This chapter describes OS Interface to Router IP (IPIF) messages. For information on message content and how to use the message, refer to the Introduction.

IPIF.001

Level: UI-ERROR

Short Syntax: IPIF.001 mosip_ *mosip_routine*: ifconfig *ifconfig_string* FAILED, errno = *errno*.

Long Syntax: IPIF.001 mosip_ *mosip_routine*: ifconfig *ifconfig_string* FAILED, errno = *errno*.

Description: IPIF ifconfig error message.

Cause: See errno.

IPIF.002

Level: UI-ERROR

Short Syntax: IPIF.002 mosip_ *mosip_routine*: route *route_string* FAILED, errno = *errno*.

Long Syntax: IPIF.002 mosip_ *mosip_routine*: route *route_string* FAILED, errno = *errno*.

Description: IPIF route error message.

Cause: See errno.

IPIF.003

Level: UI-ERROR

Short Syntax: IPIF.003 mosip_getport: *TCP/UDP* port (*requested_port*) not assigned.

Long Syntax: IPIF.003 mosip_getport: *TCP/UDP* port (*requested_port*) not assigned.

Description: IPIF port assignment error message.

Cause: TCP/UDP Port not available.

IPIF.004

Level: UI-ERROR

Short Syntax: IPIF.004 mosip_ *mosip_routine*: ERROR allocating memory (*memory_type*).

Long Syntax: IPIF.004 mosip_ *mosip_routine*: ERROR allocating memory (*memory_type*).

Description: IPIF memory allocation error message.

Cause: Memory not available.

IPIF.005

Level: C-TRACE

Short Syntax: IPIF.005 *trace_string*

Long Syntax: IPIF.005 *trace_string*

Description: IPIF trace message.

Cause: Traces are enabled in IPIF.

IPIF.006

Level: C-TRACE

Short Syntax: IPIF.006 mosip_ *mosip_routine*: route *route_string*

Long Syntax: IPIF.006 mosip_ *mosip_routine*: route *route_string*

Description: IPIF trace message.

Cause: Traces are enabled in IPIF.

IPIF.007

Level: C-TRACE

Short Syntax: IPIF.007 mosip_ *output/rcv*: UDP (*(ip_address,port)*) len *length*

Long Syntax: IPIF.007 mosip_ *output/rcv*: UDP (*(ip_address,port)*) length *length*

Description: IPIF trace message.

Cause: Traces are enabled in IPIF.

IPIF.008

Level: C-TRACE

Short Syntax: IPIF.008 mosip_getport: Ignoring error on UDP port *port_number* registration.

Long Syntax: IPIF.008 mosip_getport: Ignoring error on UDP port *port_number* registration.

Description: IPIF trace message.

Cause: Traces are enabled in IPIF.

IPIF.009

Level: C-TRACE

Short Syntax: IPIF.009 mosip_ *add/del*: ifconfig *ifconfig_string*

Long Syntax: IPIF.009 mosip_ *add/del*: ifconfig *ifconfig_string*

Description: IPIF trace message.

Cause: Traces are enabled in IPIF.

IPIF.010

Level: C-TRACE

Short Syntax: IPIF.010 mosip_port_allocate: (protocol *requested_port*,port *assigned_port*,addr *protocol*) ==> port

Long Syntax: IPIF.010 mosip_port_allocate: (protocol *requested_port*,port *assigned_port*,addr *protocol*) ==> port

Description: IPIF trace message.

Cause: Traces are enabled in IPIF.

IPIF.011

Level: C-TRACE

Short Syntax: IPIF.011 mosip_port_free: freeing UDP port *port_to_free*

Long Syntax: IPIF.011 mosip_port_free: freeing UDP port *port_to_free*

Description: IPIF trace message.

Cause: Traces are enabled in IPIF.

IPIF.012

Level: C-TRACE

Short Syntax: IPIF.012 mosip_ *output/rcv*: TCP (*(ip_address,port)*) len *length*

Long Syntax: IPIF.012 mosip_ *output/rcv*: TCP (*(ip_address,port)*) length *length*

Description: IPIF trace message.

Cause: Traces are enabled in IPIF.

IPIF.013

Level: C-TRACE

Short Syntax: IPIF.013 mosip_ *output/rcv*: ICMP (*source_ip_address*) -> (*dest_ip_address*) type *icmp_type* code *icmp_code* len *length*

Long Syntax: IPIF.013 mosip_ *output/rcv*: ICMP (*source_ip_address*) -> (*dest_ip_address*) type *icmp_type* code *icmp_code* length *length*

Description: IPIF trace message.

Cause: Traces are enabled in IPIF.

IPIF.014

Level: C-TRACE

Short Syntax: IPIF.014 mosip_ *output/rcv*: (*source_ip_address*) -> (*dest_ip_address*) protocol *protocol* len *length*

Long Syntax: IPIF.014 mosip_ *output/rcv*: (*source_ip_address*) -> (*dest_ip_address*) protocol *protocol* length *length*

Description: IPIF trace message.

Cause: Traces are enabled in IPIF.

IPIF.015

Level: UI-ERROR

Short Syntax: IPIF.015 Error on *system_call* system call, errno = *errno*

Long Syntax: IPIF.015 Error on *system_call* system call, errno = *errno*

Description: IPIF trace message.

Cause: Traces are enabled in IPIF.

IPIF.016

Level: C-TRACE

Short Syntax: IPIF.016 mosip_rcv: *TCP/UDP* dest port *port_number* refused

Long Syntax: IPIF.016 mosip_rcv: *TCP/UDP* destination port *port_number* refused

Description: IPIF trace message.

Cause: Traces are enabled in IPIF.

Chapter 88. PCA Network Interface (PCA)

This chapter describes PCA Network Interface (PCA) messages. For information on message content and how to use the message, refer to the Introduction.

PCA.001

Level: ALWAYS

Short Syntax: PCA.001 bd frm LANtype *lan_type* LANnum *lan_num* on nt *network*

Long Syntax: PCA.001 frame received for unknown LAN type *lan_type*, LAN number *lan_num* on network *network*

Description: A frame was received from the channel destined for an unknown LAN type or LAN number.

PCA.002

Level: ALWAYS

Short Syntax: PCA.002 bd not *not_id* on nt *network*

Long Syntax: PCA.002 unknown notification *not_id* received from device driver on network *network*

Description: A notification was received from the device driver that was unknown.

PCA.003

Level: UE-ERROR

Short Syntax: PCA.003 bd 8232 cmd *cmd* on nt *network*

Long Syntax: PCA.003 unknown 8232 command *cmd* received on network *network*

Description: An 8232 command was received that was unknown.

PCA.004

Level: ALWAYS

Short Syntax: PCA.004 bd cmd *cmd* on nt *network*

Long Syntax: PCA.004 unknown IORB command *cmd* received on network *network*

Description: An IORB was received that contained an unknown command.

PCA.005

Level: ALWAYS

Short Syntax: PCA.005 no subch on nt *network*

Long Syntax: PCA.005 no subchannels are defined on network *network*, cannot pass self-test

Description: There are no subchannels defined for a Parallel Channel Adapter base net so the network cannot be activated (pass self-test).

Cause: The virtual net handler(s) for this base net handler has (have) not been defined correctly.

Action: Define subchannels for the virtual net handler(s) on this Parallel Channel Adapter.

PCA.006

Level: UI-ERROR

Short Syntax: PCA.006 STOP: no IORB on nt *network*

Long Syntax: PCA.006 network *network* was unable to send a STOP command to the device driver because an IORB was not available

Description: The network was unable to complete deactivation because there was no IORB available with which to send the STOP command to the device driver.

PCA.007

Level: P-TRACE

Short Syntax: PCA.007 frm sent to lt *lantype* ln *lannumber* on nt *network*

Long Syntax: PCA.007 A frame was sent to LAN type *lantype*, LAN number *lannumber* on network *network*

Description: A frame was received on the channel and sent to a virtual net handler.

PCA.008

Level: P-TRACE

Short Syntax: PCA.008 data frm rcvd from nt *network*

Long Syntax: PCA.008 A data frame was received from network *network*

Description: A data frame was received from a virtual net handler to send to the channel.

PCA.009

Level: P-TRACE

Short Syntax: PCA.009 cmd *cmd_code* in frm rcvd from nt *network*

Long Syntax: PCA.009 command *cmd_code* in frame received from network *network*

Description: A command frame was received from a virtual net handler to send to the channel.

PCA.010

Level: P-TRACE

Short Syntax: PCA.010 notif *notif_code* rcvd on nt *network*

Long Syntax: PCA.010 notification *notif_code* received from device driver on network *network*

Description: A notification was received from the device driver.

PCA.011

Level: P-TRACE

Short Syntax: PCA.011 8232 cmd *cmd_code* rcvd on nt *network*

Long Syntax: PCA.011 8232 command *cmd_code* received on network *network*

Description: An 8232 command was received by the base net handler.

PCA.012

Level: C-TRACE

Short Syntax: PCA.012 nt *virtual_net_number* reg on nt *network*

Long Syntax: PCA.012 Network number *virtual_net_number* registering on base network *network*

Description: A virtual net handler is registering with an Parallel Channel Adapter base net handler.

PCA.013

Level: P-TRACE

Short Syntax: PCA.013 Cmd *cmd_code* fail stat *cmd_status* on nt *network*

Long Syntax: PCA.013 Command *cmd_code* to device driver failed with status *cmd_status* on network *network*

Description: A command that the base net handler sent to the device driver has failed.

PCA.014

Level: P-TRACE

Short Syntax: PCA.014 Cmd *cmd_code* sent to DD on nt *network* (sub locaddr *locaddr* devaddr *devaddr*)

Long Syntax: PCA.014 Commands *cmd_code* was sent to the device driver on network *network* (subchannel local address *locaddr*, device address *devaddr*)

Description: A command was sent to the device driver.

PCA.015

Level: P-TRACE

Short Syntax: PCA.015 Snd 8232 resp *cmd_code* (rc *retcode*) on nt *network* (sub locaddr *locaddr* devaddr *devaddr*)

Long Syntax: PCA.015 Sending 8232 response for command *cmd_code* with return code *retcode* on network *network* (subchannel local address *locaddr*, device address *devaddr*)

Description: An 8232 response was sent to the host.

PCA.016

Level: P-TRACE

Short Syntax: PCA.016 Snd not *notification_id* to net *virt_net_number* on nt *network*

Long Syntax: PCA.016 Sending notification *notification_id* to net *virt_net_number* on network *network*

Description: A notification was sent to a virtual net handler from the base net handler.

PCA.017

Level: U-TRACE

Short Syntax: PCA.017 circdn for nt *net_num* on nt *network*

Long Syntax: PCA.017 circdown for net *net_num* called on network *network*

Description: The circuit down routine for a network has been called.

PCA.018

Level: U-TRACE

Short Syntax: PCA.018 circup for nt *net_num* on nt *network*

Long Syntax: PCA.018 circup for net *net_num* called on network *network*

Description: The circuit up routine for a network has been called.

PCA.019

Level: U-TRACE

Short Syntax: PCA.019 net up for nt *net_num* on nt *network*

Long Syntax: PCA.019 net up for net *net_num* called on network *network*

Description: The net up routine for a virtual network has been called.

PCA.020

Level: U-TRACE

Short Syntax: PCA.020 net dn for nt *net_num* on nt *network*

Long Syntax: PCA.020 net down for net *net_num* called on network *network*

Description: The net down routine for a virtual network has been called.

PCA.022

Level: UI-ERROR

Short Syntax: PCA.022 PCA DD(slot *slot* adapter): Transmit List full threshold= *count*.

Long Syntax: PCA.022 PCA DD(slot *slot* adapter): Transmit List full threshold= *count*.

Description: The device driver requires a transmit list entry in order to send data to the adapter. If the adapter cannot obtain an entry during initialization, the adapter will be restarted. If the adapter cannot obtain an entry at any other time, the internal software will attempt to recover. This error will only cause real problems for LSA.

Action: If the problem persists, contact Software Support.

PCA.028

Level: UI-ERROR

Short Syntax: PCA.028 PCA DD asked to free private buffer 0x *iob* twice for adapter in slot *slot*.

Long Syntax: PCA.028 PCA DD asked to free private buffer 0x *iob* twice for adapter in slot *slot*.

Description: The device driver has freed the same private buffer to the adapter twice. This could cause data corruption.

Action: If the problem persists, contact Software Support.

PCA.029

Level: UI_ERROR

Short Syntax: PCA.029 Unable to create dump file for PCA adapter in slot *slot*. RC=0x *rc*

Long Syntax: PCA.029 Unable to create dump file for PCA adapter in slot *slot*. RC=0x *rc*

Description: A fatal error was detected by the PCA adapter microcode, but the device drive was unable to create a dump file. Contact Software Support.

PCA.030

Level: UI_ERROR

Short Syntax: PCA.030 Dump file *' filename'* created for the PCA adapter in slot *slot*.

Long Syntax: PCA.030 Dump file *' filename'* created for the PCA adapter in slot *slot*.

Description: A fatal error was detected by the PCA adapter microcode, and a dump file was created containing failure information. Contact Software Support.

PCA.034

Level: ALWAYS

Short Syntax: PCA.034 PCA in slot *slot*. AIB FLASH mismatch: code at 0x *codelev*, adapter at 0x *adaplev*

Long Syntax: PCA.034 Parallel Channel Adapter in slot *slot*. AIB FLASH mismatch: code at 0x *codelev*, adapter at 0x *adaplev*

Description: The Parallel Channel Adapter has FLASH code that is different from the level available with the current load image.

Action: Contact Software Support to determine if the FLASH code on the adapter should be updated.

PCA.035

Level: C-INFO

Short Syntax: PCA.035 PCA in slot *slot* is operational.

Long Syntax: PCA.035 Parallel Channel Adapter in slot *slot* is operational.

Description: The Parallel Channel Adapter is operational.

PCA.036

Level: UI-ERROR

Short Syntax: PCA.036 PCA error, slot= *slot*, subchan= *subchan*, rc= *rc*, origcmd= *origcmd*, sev= *sev*, correl=0x *correl*.

Long Syntax: PCA.036 Parallel Channel Adapter DD received an Error notif from slot *slot* PCA; subchan= *subchan*, rc = *rc*, origcmd= *origcmd*, severity= *sev*, correl=0x *correl*.

Description: The Parallel Channel Adapter is reporting an error to the Parallel Channel Adapter device driver.

Action: Typically, no action is required. If the problem persists, contact Software Support. Refer to the documentation for further information.

PCA.037

Level: UI-ERROR

Short Syntax: PCA.037 PCA in slot= *slot* is offline to the host.

Long Syntax: PCA.037 Parallel Channel Adapter in slot= *slot* is offline to the host.

Description: The Parallel Channel Adapter is reporting that it is offline to the host. The adapter will be automatically restarted. Diagnostics will be invoked if the adapter fails to restart.

Action: If the adapter passes diagnostics but fails to start, contact Software Support.

PCA.038

Level: UI-ERROR

Short Syntax: PCA.038 PCA DD received i960 Processor Fault notif from slot= *slot* PCA, Fault Type=0x *ft*.

Long Syntax: PCA.038 Parallel Channel Adapter DD received an i960 Processor Fault notif from slot *slot* PCA with Fault Type=0x *ft*.

Description: The Parallel Channel Adapter is reporting that it had an i960 processor fault. The adapter will be automatically restarted. Diagnostics will be invoked if the adapter fails to start.

Action: Contact Software Support.

PCA.040

Level: U-INFO

Short Syntax: PCA.040 PCA in slot *slot* had an unexpected interrupt.

Long Syntax: PCA.040 Parallel Channel Adapter DD received an Unexpected Interrupt notification from slot *slot* PCA.

Description: Parallel Channel Adapter had an unexpected interrupt. If the problem persists, contact Software Support.

PCA.042

Level: UI-ERROR

Short Syntax: PCA.042 Slot *slot* PCA microcode aborted with rc=0x *rc*.

Long Syntax: PCA.042 Parallel Channel Adapter DD received a Microcode Aborted notification from slot *slot* PCA, rc=0x *rc*.

Description: The Parallel Channel Adapter is reporting that the microcode aborted. The adapter will be automatically restarted. Diagnostics will be invoked if the adapter fails to restart.

Action: If the adapter fails to restart, contact Software Support.

PCA.044

Level: UI-ERROR

Short Syntax: PCA.044 PCA in slot *slot* had a POST error, error = 0x *error*.

Long Syntax: PCA.044 Parallel Channel Adapter in slot *slot* had a POST error, error = 0x *error*.

Description: The Parallel Channel Adapter had a POST error. The adapter will be automatically restarted. Diagnostics will be invoked if the adapter fails to restart.

Action: If adapter fails to restart, contact Software Support.

PCA.045

Level: UI-ERROR

Short Syntax: PCA.045 PCA in slot *slot* had a POST error, CBSP value=0x *error*.

Long Syntax: PCA.045 Parallel Channel Adapter in slot *slot* had a POST error, CBSP value=0x *error*.

Description: The Parallel Channel Adapter had a POST error. The adapter will be automatically restarted. Diagnostics will be invoked if the adapter fails to restart.

Action: If adapter fails to restart, contact Software Support.

PCA.046

Level: UI-ERROR

Short Syntax: PCA.046 PCA in slot *slot* did not complete POST.

Long Syntax: PCA.046 Parallel Channel Adapter in slot *slot* did not complete POST.

Description: The Parallel Channel Adapter did not complete POST. The adapter will be automatically restarted. Diagnostics will be invoked if the adapter fails to restart.

Action: If the adapter passes diagnostics but fails to restart, contact Software Support.

PCA.047

Level: UI-ERROR

Short Syntax: PCA.047 PCA in slot *slot* had a PrePOST error = 0x *error*.

Long Syntax: PCA.047 Parallel Channel Adapter in slot *slot* had a PrePOST error = 0x *error*.

Description: The Parallel Channel Adapter had a

PrePOST error. The adapter will be automatically restarted. Diagnostics will be invoked if the adapter fails to restart.

Action: If the adapter does not restart, contact Software Support.

PCA.048

Level: UI-ERROR

Short Syntax: PCA.048 Slot *slot* does not contain a PCA, identifier = *id*.

Long Syntax: PCA.048 Slot *slot* does not contain a Parallel Channel Adapter, identifier = *id*.

Description: The slot does not contain a Parallel Channel Adapter and the software has been configured for a Parallel Channel Adapter in that slot.

Action: Correct the configuration. If the problem occurs after reconfiguration, contact Software Support.

PCA.049

Level: UI-ERROR

Short Syntax: PCA.049 Slot *slot* PCA timed-out during initialization, cmd=0x *cmd*.

Long Syntax: PCA.049 Slot *slot* Parallel Channel Adapter timed-out during initialization, cmd=0x *cmd*.

Description: The adapter will be automatically restarted.

Action: If the adapter does not restart, contact Software Support.

PCA.050

Level: UI-ERROR

Short Syntax: PCA.050 Slot *slot* PCA Control Unit table did not load correctly, rc=0x *rc*, tbl=0x *tbl_num*.

Long Syntax: PCA.050 Slot *slot* Parallel Channel Adapter Control Unit table did not load correctly, rc=0x *rc*, tbl=0x *tbl_num*.

Description: The Parallel Channel Adapter cannot start properly without these tables. The adapter will be automatically restarted.

Action: If the adapter does not restart, contact Software Support.

PCA.051

Level: UI-ERROR

Short Syntax: PCA.051 PCA DD could not obtain a Control Buffer from slot *slot* adapter.

Long Syntax: PCA.051 Parallel Channel Adapter DD could not obtain a Control Buffer from adapter in slot *slot*.

Description: The device driver requires a buffer from the adapter. If the adapter cannot provide the buffer then the adapter is not functioning properly. The adapter will be restarted automatically.

Action: If the problem persists, contact Software Support.

PCA.052

Level: U-INFO

Short Syntax: PCA.052 *file*(*line*): PCA DD encountered an internal error for slot *slot*. Identifier = *id*.

Long Syntax: PCA.052 *file*(*line*): Parallel Channel Adapter DD encountered an internal error for slot *slot*. Identifier = *id*.

Description: The Parallel Channel Adapter device driver has encountered a condition that it cannot handle properly.

Action: If the problem persists, contact Software Support.

PCA.054

Level: UI-ERROR

Short Syntax: PCA.054 PCA DD could not obtain system memory; slot=0x *slot*, identifier= *id*.

Long Syntax: PCA.054 Parallel Channel Adapter DD could not obtain system memory; slot=0x *slot*, identifier= *id*.

Description: If this error occurred during initialization, the adapter will be restarted.

Action: If the problem persists, contact Software Support.

PCA.055

Level: UI-ERROR

Short Syntax: PCA.055 PCA DD could not open dump files on harddrive. Dumps not available for slot *slot* adapter.

Long Syntax: PCA.055 Parallel Channel Adapter DD could not open the dump files on the harddrive. The dumps are not available for slot *slot* adapter

Description: The device driver attempted to open a file on the harddrive but was unsuccessful. The dump of the Parallel Channel Adapter is not available.

Action: If problems with the adapter persist, contact Software Support.

PCA.056

Level: UI-ERROR

Short Syntax: PCA.056 PCA DD could not dump all slot *slot* PCA *data_type* data to the dump file.

Long Syntax: PCA.056 Parallel Channel Adapter DD could not dump all of the slot *slot* PCA *data_type* data to the dump file on the harddrive.

Description: The device driver attempted to dump the Parallel Channel Adapter data to a file on the harddrive. The IRAM dump may be partially available in c:\PCAIx.DMP, where x is the slot number. The DRAM dump may be partially available in c:\PCADx.DMP, where x is the slot number.

Action: Contact Software Support.

PCA.057

Level: C-INFO

Short Syntax: PCA.057 PCA DD received a reset subchannel notif for subchannel 0x *sc*, slot= *slot*.

Long Syntax: PCA.057 Parallel Channel Adapter DD received a reset subchannel notification for subchannel 0x *sc*, slot= *slot*.

Description: The device driver received a reset subchannel notification.

PCA.058

Level: C-INFO

Short Syntax: PCA.058 Incorrect subchannel configuration detected for slot *slot* PCA.

Long Syntax: PCA.058 Incorrect subchannel configuration detected for slot *slot* Parallel Channel Adapter.

Description: The device driver has detected that a subchannel configuration is incorrect. Correctly configured subchannels should not be affected by this problem.

Action: Correct the configuration.

PCA.059

Level: UI-ERROR

Short Syntax: PCA.059 PCA DD(slot *slot* adapter): Command FIFO full threshold= *count*.

Long Syntax: PCA.059 PCA DD(slot *slot* adapter): Command FIFO full threshold= *count*.

Description: The device driver requires a Command FIFO entry in order to communicate with the adapter. If the adapter cannot obtain an entry during initialization, the adapter will be restarted. If the adapter cannot obtain an entry at any other time, the internal software will attempt to recover. This error will only cause real problems for LSA.

Action: If the problem persists, contact Software Support.

PCA.060

Level: P-TRACE

Short Syntax: PCA.060 PCA DD sending frame from slot= *slot*,, subchan= *subchan*,, LT= *lantype*,, LN= *lannum*, to base net.

Long Syntax: PCA.060 Parallel Channel Adapter DD rcvd frame from slot *slot*, PCA, subchan= *subchan*,, LanType= *lantype*,, and LanNum= *lannum*; sending it to base net.

Description: A frame was received by the channel and was sent to the Parallel Channel Adapter base net handler.

PCA.061

Level: P-TRACE

Short Syntax: PCA.061 PCA DD rcvd frame from net handler for slot= *slot*,, subchan= *subchan*,, LT= *lantype*,, LN= *lannum*,,PDU-hdr= *pdu_len*

Long Syntax: PCA.061 Parallel Channel Adapter DD received a frame from a net handler destined for slot *slot*, PCA, subchan= *subchan*,, LanType= *lantype*,, and LanNum= *lannum*,, PDU-header len= *pdu_len*.

Description: A Parallel Channel Adapter-related nethandler sent the Parallel Channel Adapter DD a frame to transmit.

PCA.062

Level: P-TRACE

Short Syntax: PCA.062 PCA DD rcvd *cmd*, cmd from net handler for slot *slot* PCA.

Long Syntax: PCA.062 Parallel Channel Adapter DD received *cmd*, command from net handler for slot *slot* PCA.

Description: A Parallel Channel Adapter-related net handler sent the Parallel Channel Adapter DD a command.

PCA.063

Level: P-TRACE

Short Syntax: PCA.063 PCA DD rcvd *cmd*, cmd from nethandler for slot *slot*, PCA, subchan= *subchan*.

Long Syntax: PCA.063 Parallel Channel Adapter DD received *cmd*, command from a nethandler for slot *slot*, PCA, subchan= *subchan*.

Description: A Parallel Channel Adapter-related net

handler sent the Parallel Channel Adapter DD a command.

PCA.064

Level: P-TRACE

Short Syntax: PCA.064 PCA DD sent *notif,* notif for slot *slot,* PCA, subchan= *subchan,*, LT= *lantype,*, LN= *lannum*, to nethandler.

Long Syntax: PCA.064 Parallel Channel Adapter DD sent *notif,* notif for slot *slot,* PCA, subchan= *subchan,*, LT= *lantype,*, LN= *lannum*, to nethandler.

Description: The Parallel Channel Adapter device driver sent a notification to a Parallel Channel Adapter-related net handler

---

PCA.065

Level: U-INFO

Short Syntax: PCA.065 PCA ran out of rcv buffers, LCS frame discarded, slot= *slot*, local sc= *subchan*

Long Syntax: PCA.065 Parallel Channel Adapter ran out of receive buffers and discarded an LCS frame; slot= *slot* local subchan= *subchan*.

Description: The Parallel Channel Adapter is reporting that it discarded an LCS frame because it could not obtain a receive buffer.

Action: Typically, no action is required. If the problem persists, increase the number of receive buffers for this Parallel Channel Adapter.

PCA.066

Level: UI-ERROR

Short Syntax: PCA.066 PCA ran out of rcv buffers, LSA frame discarded, slot= *slot*, local sc= *subchan*

Long Syntax: PCA.066 Parallel Channel Adapter ran out of receive buffers and discarded an LSA frame; slot= *slot* local subchan= *subchan*.

Description: The Parallel Channel Adapter is reporting that it discarded an LSA frame because it could not obtain a receive buffer.

Action: Increase the number of receive buffers for this Parallel Channel Adapter.

PCA.067

Level: U-INFO

Short Syntax: PCA.067 PCA ran out of rcv buffers, MPC+ frame discarded, slot= *slot*, local sc= *subchan*

Long Syntax: PCA.067 Parallel Channel Adapter ran out of receive buffers and discarded an MPC+ frame; slot= *slot* local subchan= *subchan*.

Description: The Parallel Channel Adapter is reporting that it discarded an MPC+ frame because it could not obtain a receive buffer.

Action: Typically, no action is required. If the problem persists, increase the number of receive buffers for this Parallel Channel Adapter.

Panic pcanomem

Short Syntax: pcanomem: Parallel Channel Adapter handler no memory

Description: An Parallel Channel Adapter handler cannot allocate memory for control block(s).

Action: Contact customer service.

Panic pcansram

Short Syntax: pcansram: Parallel Channel Adapter SRAM not found

Description: The SRAM record for an Parallel Channel Adapter handler could not be found.

Action: Contact customer service.

Panic pcabprt

Short Syntax: pcabprt: bad prot init

Description: An unsupported Network Layer protocol tried to initialize an Parallel Channel Adapter handler.

Action: Contact customer service.

Panic pcadreg

Short Syntax: pcadreg: virt net already reg

Description: An Parallel Channel Adapter virtual net handler has already registered with the base.

Action: Contact customer service.

Panic pcabreq

Short Syntax: pcabreq: bad xmit rqst

Description: An unsupported protocol packet was given to the Parallel Channel Adapter handler for transmission.

Action: Contact customer service.

Panic pcanosub

Short Syntax: pcanosub: subch not found

Description: The requested device address was not found in the PCA base handler subchannel table.

Action: Contact customer service.

Panic pcabcall

Short Syntax: pcabcall: bad call to routine.

Description: An invalid call was made to a routine.

Action: Contact customer service.

Panic pcabprd

Short Syntax: pcabprt: bad prot down

Description: An unsupported Network Layer protocol tried to uninitialize a Parallel Channel Adapter handler.

Action: Contact customer service.

Chapter 89. PKI

This chapter describes PKI messages. For information on message content and how to use the message, refer to the Introduction.

PKI.001

Level: U-INFO

Short Syntax: PKI.001 PKI TFTP file *file_name* from *server_IP_addr* status *status*.

Long Syntax: PKI.001 PKI TFTP file *file_name* from *server_IP_addr* status *status*.

Description: PKI tftp status

PKI.002

Level: P_TRACE

Short Syntax: PKI.002 LDAP signal check status: *LDAP_signal_check_status*.

Long Syntax: PKI.002 LDAP signal check status: *LDAP_signal_check_status*.

Description: LDAP signal status report

PKI.003

Level: U-INFO

Short Syntax: PKI.003 LDAP connection opening completion.

Long Syntax: PKI.003 LDAP connection opening completion.

Description: Upcall for LDAP connection opening completion.

PKI.004

Level: U-INFO

Short Syntax: PKI.004 LDAP binding completion.

Long Syntax: PKI.004 LDAP binding completion.

Description: Upcall for LDAP simple binding completion.

PKI.005

Level: U-INFO

Short Syntax: PKI.005 LDAP cert file size *cert_size* too big, over 2048 bytes

Long Syntax: PKI.005 LDAP cert file size *cert_size* too big, over 2048 bytes

Description: The cert file LDAp downloaded is over the max cert size we support.

PKI.006

Level: U-INFO

Short Syntax: PKI.006 RSA key pair gen init status: *status*.

Long Syntax: PKI.006 RSA key pair generation initialization status: *status*.

Description: The status of RSA key pair generation initialization.

PKI.007

Level: U-INFO

Short Syntax: PKI.007 RSA key pair (*size_bits* bits) gen status: *status*.

Long Syntax: PKI.007 RSA key pair (*size_bits* bits) generation status: *status*.

Description: The status of RSA key pair generation.

PKI.008

Level: U-INFO

Short Syntax: PKI.008 BCert C_CreateExtensionsObject() *status* status code *error_code*.

Long Syntax: PKI.008 BCert C_CreateExtensionsObject() *status* status code *error_code*.

Description: The status of Bcert call C_CreateExtensionsObject()

PKI.009

Level: UI-ERROR

Short Syntax: PKI.009 Validity check: *status*. Current date: *year*/ *month*/ *day*, Time: *hour*: *min*: *sec*. Cert valid date: *start_year*/ *month*/ *day hour*: *min*: *sec* -- *end_year*/ *month*/ *day hour*: *min*: *sec*.

Long Syntax: PKI.009 Validity check: *status*. Current date: *year*/ *month*/ *day*, Time: *hour*: *min*: *sec*. Cert valid date: *start_year*/ *month*/ *day hour*: *min*: *sec* -- *end_year*/ *month*/ *day hour*: *min*: *sec* .

Description: Certificate validity check status

PKI.010

Level: U-INFO

Short Syntax: PKI.010 Root CA in cache? *status* length= *length*.

Long Syntax: PKI.010 Root CA in cache? *status* length= *length*.

Description: Status of whether the root ca cert is in cache.

PKI.011

Level: UI-ERROR

Short Syntax: PKI.011 BSafe B_CreateKeyObject() error code *bsafe_error_code*.

Long Syntax: PKI.011 BSafe B_CreateKeyObject() error code *bsafe_error_code*.

Description: The B_CreateKeyObject() failed.

PKI.012

Level: UI-ERROR

Short Syntax: PKI.012 BSafe B_SetKeyInfo() error code *bsafe_error_code*.

Long Syntax: PKI.012 BSafe B_SetKeyInfo() error code *bsafe_error_code*

Description: The B_SetKeyInfo() failed.

PKI.013

Level: UI-ERROR

Short Syntax: PKI.013 BCert C_VerifyCertSignature() error code *bcert_error_code*.

Long Syntax: PKI.013 BCert C_VerifyCertSignature() error code *bcert_error_code*.

Description: The C_VerifyCertSignature() failed.

PKI.014

Level: U-INFO

Short Syntax: PKI.014 BCert C_CreateCertObject() *status* status code *status_code*.

Long Syntax: PKI.014 BCert C_CreateCertObject() *status* status code *status_code*.

Description: The status of Bcert call C_CreateCertObject()

PKI.015

Level: U-INFO

Short Syntax: PKI.015 BCert C_SetCertBER() *status* status code *code*.

Long Syntax: PKI.015 BCert C_SetCertBER() *status* status code *code*.

Description: The status of Bcert call C_SetCertBER()

PKI.016

Level: U-INFO

Short Syntax: PKI.016 BCert C_GetCertFields() *status* status code *status_code*.

Long Syntax: PKI.016 BCert C_GetCertFields() *status* status code *status_code*.

Description: The status of Bcert callC_GetCertFields()

PKI.017

Level: U-INFO

Short Syntax: PKI.017 BCert C_GetNameDER() *status* status code *status_code*.

Long Syntax: PKI.017 BCert C_GetNameDER() *status* status code *status_code*.

Description: The status of Bcert callC_GetNameDER()

PKI.018

Level: UI-ERROR

Short Syntax: PKI.018 X500DNBER2Str() error.

Long Syntax: PKI.018 X500DNBER2Str() error

Description: X500DNBER2Str() return error

PKI.019

Level: U-INFO

Short Syntax: PKI.019 BCert C_GetExtensionValue() *status* status code *status_code*.

Long Syntax: PKI.019 BCert C_GetExtensionValue() *status* status code *status_code*.

Description: The status of Bcert callC_GetExtensionValue()

PKI.020

Level: UI-ERROR

Short Syntax: PKI.020 BSafe B_GetKeyInfo() error code *bsafe_error_code*.

Long Syntax: PKI.020 BSafe B_GetKeyInfo() error code *bsafe_error_code*

Description: The B_GetKeyInfo() failed.

PKI.021

Level: U-INFO

Short Syntax: PKI.021 BCert C_FindExtensionByType() *status* status code *status_code*.

Long Syntax: PKI.021 BCert C_FindExtensionByType() *status* status code *status_code*.

Description: The status of Bcert callC_FindExtensionByType()

PKI.022

Level: U-INFO

Short Syntax: PKI.022 BCert C_GetExtensionInfo() *status* status code *status_code*.

Long Syntax: PKI.022 BCert C_GetExtensionInfo() *status* status code *status_code*.

Description: The status of Bcert callC_GetExtensionInfo()

PKI.023

Level: U-INFO

Short Syntax: PKI.023 BCert C_CreateAttributesObject() *status* code *error_code*.

Long Syntax: PKI.023 BCert C_CreateAttributesObject() *status* code *error_code*.

Description: The status of Bcert call C_CreateAttributesObject()

PKI.024

Level: U-INFO

Short Syntax: PKI.024 BCert C_GetAttributeInExtensionsObj() *status* status code *status_code*.

Long Syntax: PKI.024 BCert C_GetAttributeInExtensionsObj() *status* status code *status_code*.

Description: The status of Bcert call C_GetAttributeInExtensionsObj()

PKI.025

Level: U-INFO

Short Syntax: PKI.025 BCert C_CreateExtension() *status* status code *code*.

Long Syntax: PKI.025 BCert C_CreateExtension() *status* status code *code*.

Description: The status of Bcert call C_CreateExtension()

PKI.026

Level: U-INFO

Short Syntax: PKI.026 BCert C_AddExtensionValue() *status* status code *error_code*.

Long Syntax: PKI.026 BCert C_AddExtensionValue() *status* status code *error_code*.

Description: The status of Bcert call C_AddExtensionValue()

PKI.027

Level: U-INFO

Short Syntax: PKI.027 BCert C_CreateNameObject() *status* status code *error_code*.

Long Syntax: PKI.027 BCert C_CreateNameObject() *status* status code *error_code*.

Description: The status of Bcert call C_CreateNameObject()

PKI.028

Level: U-INFO

Short Syntax: PKI.028 BCert C_CreateCertRequestObject() *status* code *error_code*.

Long Syntax: PKI.028 BCert C_CreateCertRequestObject() *status* code *error_code*.

Description: The status of Bcert call C_CreateCertRequestObject()

PKI.029

Level: U-INFO

Short Syntax: PKI.029 BSafe B_GetKeyInfo() *status* code *error_code*.

Long Syntax: PKI.029 BCert B_GetKeyInfo() *status* code *error_code*.

Description: The status of Bcert call B_GetKeyInfo()

PKI.030

Level: U-INFO

Short Syntax: PKI.030 BCert C_SetCertRequestFields() *status* code *error_code*.

Long Syntax: PKI.030 BCert C_SetCertRequestFields() *status* code *error_code*.

Description: The status of Bcert call C_SetCertRequestFields()

PKI.031

Level: U-INFO

Short Syntax: PKI.031 BCert C_SignCertRequest() *status* code *error_code*.

Long Syntax: PKI.031 BCert C_SignCertRequest() *status* code *error_code*.

Description: The status of Bcert call C_SignCertRequest()

PKI.032

Level: U-INFO

Short Syntax: PKI.032 BCert C_GetCertRequestDER() *status* code *error_code*.

Long Syntax: PKI.032 BCert C_GetCertRequestDER() *status* code *error_code*.

Description: The status of Bcert call C_GetCertRequestDER()

PKI.033

Level: UI-ERROR

Short Syntax: PKI.033 Warning: *cert_type* Cert is loaded more than once. cert ID= *cert_ID*

Long Syntax: PKI.033 Warning: *cert_type* Cert is loaded more than once. cert ID= *cert_ID*

Description: The default cert is loaded more than once

PKI.034

Level: U-INFO

Short Syntax: PKI.034 Load *cert_type* Cert *status*. cert ID= *cert_ID*

Long Syntax: PKI.034 Load *cert_type* Cert *status*. cert ID= *cert_ID*

Description: The status of loading certificate into cache.

PKI.035

Level: U-INFO

Short Syntax: PKI.035 Default *cert_type* Cert not found, not loaded.

Long Syntax: PKI.035 Default *cert_type* Cert not found, not loaded.

Description: The status of the default cert loaded into cache.

PKI.036

Level: U-INFO

Short Syntax: PKI.036 BSafe B_CreateAlgorithmObject() *status* status code *bsafe_code*.

Long Syntax: PKI.036 BSafe B_CreateAlgorithmObject() *status* status code *bsafe_code*.

Description: The status of calling Bsafe B_CreateAlgorithmObject().

PKI.037

Level: U-INFO

Short Syntax: PKI.037 BSafe B_SetAlgorithmObject() *status* status code *bsafe_code*.

Long Syntax: PKI.037 BSafe B_SetAlgorithmObject() *status* status code *bsafe_code*.

Description: The status of calling Bsafe B_SetAlgorithmObject().

PKI.038

Level: U-INFO

Short Syntax: PKI.038 BSafe B_GenerateInit() *status* status code *bsafe_code*.

Long Syntax: PKI.038 BSafe B_GenerateInit() *status* status code *bsafe_code*.

Description: The status of calling Bsafe B_GenerateInit().

PKI.039

Level: U-INFO

Short Syntax: PKI.039 BSafe B_GenerateKeypair() *status* error code *bsafe_code*.

Long Syntax: PKI.039 BSafe B_GenerateKeypair() *status* error code *bsafe_code*.

Description: The status of calling Bsafe B_GenerateKeypair().

PKI.040

Level: U-INFO

Short Syntax: PKI.040 BCert C_AddNameAVA() *status* code *error_code* Name str: *type*.

Long Syntax: PKI.040 BCert C_AddNameAVA() *status* code *error_code* Name str: *type*.

Description: The status of Bcert call C_AddNameAVA()

PKI.041

Level: U-INFO

Short Syntax: PKI.041 X500DN initialized. Total X500 Attribute name *total_attr_name*

Long Syntax: PKI.041 X500DN initialized. Total X500 Attribute name *total_attr_name*

Description: The X500 Attribute name function is initialized.

PKI.042

Level: U-INFO

Short Syntax: PKI.042 BCert C_SetNameBER() *status* code *error_code*.

Long Syntax: PKI.042 BCert C_SetNameBER() *status* code *error_code*.

Description: The status of Bcert call C_SetNameBER()

PKI.043

Level: U-INFO

Short Syntax: PKI.043 BCert C_IsSubjectSubordinateToIssuer() *status* code *error_code*.

Long Syntax: PKI.043 BCert C_IsSubjectSubordinateToIssuer() *status* code *error_code*.

Description: The status of Bcert call C_IsSubjectSubordinateToIssuer()

PKI.044

Level: U-INFO

Short Syntax: PKI.044 BCert C_GetNameAVACount() *status* code *error_code*.

Long Syntax: PKI.044 BCert C_GetNameAVACount() *status* code *error_code*.

Description: The status of Bcert call C_GetNameAVACount()

PKI.045

Level: U-INFO

Short Syntax: PKI.045 BCert C_GetNameAVA() *status* code *error_code*.

Long Syntax: PKI.045 BCert C_GetNameAVA() *status* code *error_code*.

Description: The status of Bcert call C_GetNameAVA()

PKI.046

Level: UI-ERROR

Short Syntax: PKI.046 Cert size (*size* bytes) not correct.

Long Syntax: PKI.046 Certificate size (*size* bytes) not correct. Too big.

Description: the certificate size is not right.

PKI.047

Level: UI-ERROR

Short Syntax: PKI.047 Private key size (*size* bytes) not correct.

Long Syntax: PKI.047 Private Key size (*size* bytes) not correct. Too big.

Description: The private key size is not right.

PKI.048

Level: U-INFO

Short Syntax: PKI.048 PKI initialized.

Long Syntax: PKI.048 PKI initialized.

Description: PKI initialization finished.

PKI.049

Level: UI-ERROR

Short Syntax: PKI.049 PKI DN attr table not initialized properly. Table szie= *size*.

Long Syntax: PKI.049 PKI DN attribute table is not initialized properly. Table szie= *size*.

Description: The attribute table is not initilized.

PKI.050

Level: UI-ERROR

Short Syntax: PKI.050 PKI X500 DN attr Name not found.

Long Syntax: PKI.050 PKI X500 attribute name is not found in the current attribute table.

Description: X500 DN attribute name not found.

PKI.051

Level: UI-ERROR

Short Syntax: PKI.051 PKI X500 Name type not found.

Long Syntax: PKI.051 PKI X500 Name type is not found in the current attribute table.

Description: X500 DN name type not found.

PKI.052

Level: UI-ERROR

Short Syntax: PKI.052 PKI *ID_type* type unknown.

Long Syntax: PKI.052 PKI *ID_type* type unknown.

Description: PKI ID|alt-name type unknown.

PKI.053

Level: U-INFO

Short Syntax: PKI.053 PKI *status* Value *alt-name*

Long Syntax: PKI.053 PKI *status* Value *alt-name*

Description: PKI subject alt-name status

PKI.054

Level: UI-ERROR

Short Syntax: PKI.054 PKI ID bad IP addr

Long Syntax: PKI.054 PKI ID bad IP addr in subj-alt-name in cert

Description: PKI ID bad IP addr in subj-alt-name in cert

PKI.055

Level: UI-ERROR

Short Syntax: PKI.055 PKI subj-alt-name not parsed correctly.

Long Syntax: PKI.055 PKI subj-alt-name not parsed correctly.

Description: PKI subj-alt-name not parsed correctly.\n\r

PKI.056

Level: U-INFO

Short Syntax: PKI.056 PKI cert validity check status *status*

Long Syntax: PKI.056 PKI certificate validity check status *status*

Description: PKI certificate validity check status

PKI.057

Level: U-INFO

Short Syntax: PKI.057 PKI cert root CA check status *status*

Long Syntax: PKI.057 PKI certificate root CA check status *status*

Description: PKI certificate root CA check status

PKI.058

Level: U-INFO

Short Syntax: PKI.058 PKI store peer public key status *status*

Long Syntax: PKI.058 PKI store peer public key status *status*

Description: PKI store peer public key status

PKI.059

Level: U-INFO

Short Syntax: PKI.059 PKI store *cert_type* cert ID status *status*

Long Syntax: PKI.059 PKI store *cert_type* cert ID status *status*

Description: PKI store peer|router cert ID status

PKI.060

Level: U-INFO

Short Syntax: PKI.060 PKI processing *cert_type* cert successful

Long Syntax: PKI.060 PKI processing *cert_type* cert successful

Description: PKI successully processed cert.

PKI.061

Level: U-INFO

Short Syntax: PKI.061 PKI cert alt-name processing status: *status*

Long Syntax: PKI.061 PKI cert alt-name processing status: *status*

Description: status of PKI cert alt-name processing status

PKI.062

Level: U-INFO

Short Syntax: PKI.062 PEM->DER conversion: Header *status*, Footer *status*

Long Syntax: PKI.062 PEM->DER conversion: Header *status*, Footer *status*

Description: The status of extracting PEM file header and fotter

---

PKI.063

Level: U-INFO

Short Syntax: PKI.063 PEM->DER buffer conversion: *status*

Long Syntax: PKI.063 PEM->DER buffer conversion: *status*

Description: The status of converting PEM buffer to DER buffer.

PKI.064

Level: UI-ERROR

Short Syntax: PKI.064 PKI unable to get mem: *location*

Long Syntax: PKI.064 PKI unable to get memory: *location*

Description: PKI was unable to allocate the necessary memory PKI is unable to run because of this.

Cause: There is a shortage in heap memory, possibly because too many memory intensive forwarders/protocols are running.

Action: Disable unnecesary forwarders/protocols or get more memory.

PKI.065

Level: UI-ERROR

Short Syntax: PKI.065 PKI Cert key usuage *key_usuage* check *status*

Long Syntax: PKI.065 PKI Cert key usuage *key_usuage* check *status*

Description: Check on Cert key usuage

PKI.066

Level: U-INFO

Short Syntax: PKI.066 Store *cert_type(remote|local)* cert life from router start *cert_lif_sec*

Long Syntax: PKI.066 Store *cert_type(remote|local)* certificate life router start *cert_lif_sec*

Description: The status of storing certificate lifetime

PKI.067

Level: U-INFO

Short Syntax: PKI.067 not used now

Long Syntax: PKI.067 not used now

Description: not used now

PKI.068

Level: U-INFO

Short Syntax: PKI.068 Retrieve peer cert public key *status*

Long Syntax: PKI.068 Retrieve peer certificate public key *status*.

Description: The status of Retrieving peer certificate's public key.

PKI.069

Level: U-INFO

Short Syntax: PKI.069 Store my cert into cache *status*. cert length= *len*

Long Syntax: PKI.069 Store my cert into cache *status*. cert length= *len*

Description: The status of storing my certificate into runtime cache

---

PKI.070

Level: U-INFO

Short Syntax: PKI.070 Store my private key into cache *status*. key buffer length= *len*

Long Syntax: PKI.070 Store my private key into cache *status*. key buffer length= *len*

Description: The status of storing my private key into runtime cache

---

PKI.071

Level: U-INFO

Short Syntax: PKI.071 Store root cert into cache *status*. cert length= *len*

Long Syntax: PKI.071 Store root cert into cache *status*. cert length= *len*

Description: The status of storing root certificate into runtime cache

---

PKI.072

Level: U-INFO

Short Syntax: PKI.072 Get my cert from cache *status*. cert length= *len*

Long Syntax: PKI.072 Get my cert from cache *status*. cert length= *len*

Description: The status of getting my certificate from runtime cache

---

PKI.073

Level: U-INFO

Short Syntax: PKI.073 BSafe B_EncryptInit() *status* status code *bsafe_code*.

Long Syntax: PKI.073 BSafe B_EncryptInit() *status* status code *bsafe_code*.

Description: The status of calling Bsafe B_EncryptInit()

PKI.074

Level: U-INFO

Short Syntax: PKI.074 BSafe B_EncryptUpdate() *status* status code *bsafe_code*.

Long Syntax: PKI.074 BSafe B_EncryptUpdate() *status* status code *bsafe_code*.

Description: The status of calling Bsafe B_EncryptUpdate()

PKI.075

Level: U-INFO

Short Syntax: PKI.075 BSafe B_EncryptFinal() *status* status code *bsafe_code*.

Long Syntax: PKI.075 BSafe B_EncryptFinal() *status* status code *bsafe_code*.

Description: The status of calling Bsafe B_EncryptFinal()

PKI.076

Level: U-INFO

Short Syntax: PKI.076 BSafe B_DecryptInit() *status* status code *bsafe_code*.

Long Syntax: PKI.076 BSafe B_DecryptInit() *status* status code *bsafe_code*.

Description: The status of calling Bsafe B_DecryptInit()

PKI.077

Level: U-INFO

Short Syntax: PKI.077 BSafe B_DecryptUpdate() *status* status code *bsafe_code*.

Long Syntax: PKI.077 BSafe B_DecryptUpdate() *status* status code *bsafe_code*.

Description: The status of calling Bsafe B_DecryptUpdate()

PKI.078

Level: U-INFO

Short Syntax: PKI.078 BSafe B_DecryptFinal() *status* status code *bsafe_code*.

Long Syntax: PKI.078 BSafe B_DecryptFinal() *status* status code *bsafe_code*.

Description: The status of calling Bsafe B_DecryptFinal()

PKI.079

Level: U-INFO

Short Syntax: PKI.079 BSafe B_SetAlgorithmInfo() *status* status code *bsafe_code*.

Long Syntax: PKI.079 BSafe B_SetAlgorithmInfo() *status* status code *bsafe_code*.

Description: The status of calling Bsafe B_SetAlgorithmInfo().

PKI.080

Level: U-INFO

Short Syntax: PKI.080 BSafe B_RandomInit() *status* status code *bsafe_code*.

Long Syntax: PKI.080 BSafe B_RandomInit() *status* status code *bsafe_code*.

Description: The status of calling Bsafe B_RandomInit().

PKI.081

Level: U-INFO

Short Syntax: PKI.081 BSafe B_RandomUpdate() *status* status code *bsafe_code*.

Long Syntax: PKI.081 BSafe B_RandomUpdate() *status* status code *bsafe_code*.

Description: The status of calling Bsafe B_RandomUpdate().

Chapter 90. Point to Point Protocol Network Interface (PPP)

This chapter describes Point to Point Protocol Network Interface (PPP) messages. For information on message content and how to use the message, refer to the Introduction.

PPP.001

Level: C-INFO

Short Syntax: PPP.001 Req brng up IP, addr = *ip_address* nt *network ID*

Long Syntax: PPP.001 Request to bring up IP, local address = *ip_address*, on network *network ID*

Description: ppp_prinit routine called for IP protocol

PPP.002

Level: C-INFO

Short Syntax: PPP.002 Srl prt up, nt *network ID*

Long Syntax: PPP.002 Serial port came up sucessfully, on network *network ID*

Description: ppp_slftst2 routine liked the results of the load and init.

PPP.003

Level: C-TRACE

Short Syntax: PPP.003 Mnt nt *network ID*

Long Syntax: PPP.003 Doing maint, on network *network ID*

Description: Entering ppp_mnt

PPP.004

Level: P-TRACE

Short Syntax: PPP.004 Nt opn fr outb *protocol_name,* nt *network ID*

Long Syntax: PPP.004 Outbound data discarded, not open for protocol *protocol_name,* on network *network ID*

Description: ppp_send was called for IP data when IP state is not open (OK).

PPP.005

Level: U-INFO

Short Syntax: PPP.005 Bd IP pkt xmt typ= *type,* nt *network ID*

Long Syntax: PPP.005 Bad IP packet to transmit: type = *type*,, on network *network ID*

Description: slhc returned bad code for IP packet.

PPP.006

Level: CE-ERROR

Short Syntax: PPP.006 I_ERR on rcv nt *network ID*

Long Syntax: PPP.006 Packet received with I_ERR set, on network *network ID*

Description: ppp_in received packet with I_ERR set.

PPP.007

Level: UE-ERROR

Short Syntax: PPP.007 Rcv Bd fr addr *bad_address,* nt *network ID*

Long Syntax: PPP.007 Received packet with bad frame address = *bad_address*,, on network *network ID*

Description: ppp_in got a frame with address byte not 0xff.

PPP.008

Level: UE-ERROR

Short Syntax: PPP.008 Rcv Bd fr cntrl *bad_control,* nt *network ID*

Long Syntax: PPP.008 Received packet with bad frame control field = *bad_control*,, on network *network ID*

Description: ppp_in got a frame with control byte not = 3 (UI).

PPP.009

Level: UE-ERROR

Short Syntax: PPP.009 Rcv inv prtcl *bad_protocol,* nt *network ID*

Long Syntax: PPP.009 Received packet with invalid protocol = *bad_protocol*,, on network *network ID*

Description: ppp_in got a frame with protocol not valid (as opposed to unknown).

PPP.010

Level: CE-ERROR

Short Syntax: PPP.010 Nt opn fr inb *protocol_name,* nt *network ID*

Long Syntax: PPP.010 Inbound data discarded, not open for protocol *protocol_name,* on network *network ID*

Description: ppp_in received data when protocol state is not open.

PPP.011

Level: CE-ERROR

Short Syntax: PPP.011 Nt opn fr inb *control_protocol_name,* nt *network ID*

Long Syntax: PPP.011 Inbound *control_protocol_name,* discarded, not open for IPCP on network *network ID*

Description: ppp_in received control protocol data when LCP state is not open.

PPP.012

Level: UE-ERROR

Short Syntax: PPP.012 Multi Encrypt - discard nt *network ID*

Long Syntax: PPP.012 Multiple Encrypt. - discard on network *network ID*

Description: For export reasons, we do not support encrypting a packet twice within the box. This ELS message is logged before discarding already encrypted packets when PPP encryption has been negotiated.

Cause: Encrypted packet passed to PPP when PPP encryption is negotiated

Action: Reconfigure IPSEC or PPP so that encryption is not attempted more than once.

PPP.014

Level: C-TRACE

Short Syntax: PPP.014 *fsm_name,/ fsm_state, routine_name,* nt *network ID*

Long Syntax: PPP.014 FSM = *fsm_name,*, state = *fsm_state,*, called *routine_name,*, on network *network ID*

Description: Called the specified fsm routine.

PPP.015

Level: UI-ERROR

Short Syntax: PPP.015 *fsm_name,/ fsm_state,* snd bd cd *code,* xmt, nt *network ID*

Long Syntax: PPP.015 FSM = *fsm_name,*, state = *fsm_state,*, tried to send bad code *code,*, on network *network ID*

Description: fsm_send called to send packet with bad code.

Cause: Control blocks and/or memory corruption

Action: Restart the router and call customer service

PPP.016

Level: P-TRACE

Short Syntax: PPP.016 *fsm_name,/ fsm_state,* snd *code,*, id *id,* len *len,*, nt *network ID*

Long Syntax: PPP.016 FSM = *fsm_name,*, state = *fsm_state,*, sending *code,*, id *id,*, len *len,*, on network *network ID*

Description: fsm_send about to send fsm message.

PPP.017

Level: P-TRACE

Short Syntax: PPP.017 *fsm_name,/ fsm_state,* rcv *code,*, id *id,* len *len,*, nt *network ID*

Long Syntax: PPP.017 FSM = *fsm_name,*, state = *fsm_state,*, received *code,*, id *id,*, len *len,*, on network *network ID*

Description: fsm_proc received fsm message.

PPP.018

Level: CE-ERROR

Short Syntax: PPP.018 *fsm_name, msg_type,* retr exc nt *network ID*

Long Syntax: PPP.018 *fsm_name,* FSM, *msg_type,* retries exceeded, on network *network ID*

Description: Too many retries of a config request or terminate request.

PPP.019

Level: C-TRACE

Short Syntax: PPP.019 LCP/ *lcp_state, routine_name,* nt *network ID*

Long Syntax: PPP.019 LCP, state = *lcp_state,*, called *routine_name,*, on network *network ID*

Description: Called the specified lcp routine.

PPP.020

Level: UE-ERROR

Short Syntax: PPP.020 Bd lcp rej id, exp *exp_id,* gt *got_id,*, nt *network ID*

Long Syntax: PPP.020 Bad LCP reject id, expected *exp_id,*, got *got_id,*, on network *network ID*

Description: lcp_rej got reject with bad id.

PPP.021

Level: UE-ERROR

Short Syntax: PPP.021 Bd lcp rej lngth, nt *network ID*

Long Syntax: PPP.021 Bad LCP reject length, on network *network ID*

Description: lcp_rej got reject with bad length.

PPP.022

Level: UE-ERROR

Short Syntax: PPP.022 Bd lcp rej opt *lcp_option,*, nt *network ID*

Long Syntax: PPP.022 Bad LCP reject option = *lcp_option,*, on network *network ID*

Description: lcp_rej got reject containing out-of-range option.

PPP.023

Level: UE-ERROR

Short Syntax: PPP.023 out-ordr lcp rej opt *lcp_option,*, nt *network ID*

Long Syntax: PPP.023 Bad LCP reject option = *lcp_option,*, on network *network ID*

Description: lcp_rej got reject containing out-of-order option.

PPP.024

Level: UE-ERROR

Short Syntax: PPP.024 Bd lcp req hdr lngth, nt *network ID*

Long Syntax: PPP.024 Bad LCP request header length, on network *network ID*

Description: lcp_req got request with bad header length.

PPP.025

Level: UE-ERROR

Short Syntax: PPP.025 Bd lcp req opt *lcp_option,* shrt, nt *network ID*

Long Syntax: PPP.025 Bad LCP request option = *lcp_option,*, data too short, on network *network ID*

Description: lcp_req got request containing option with insufficient data.

PPP.026

Level: C-TRACE

Short Syntax: PPP.026 lcp req rslt: *lcp_rslt,*, opt *lcp_option,*, ln *opt_len,*, nt *network ID*

Long Syntax: PPP.026 lcp request result: *lcp_rslt,*, option = *lcp_option,*, length = *opt_len,*, on network *network ID*

Description: Result, so far, of processing one option.

PPP.027

Level: UE-ERROR

Short Syntax: PPP.027 Bd lcp ack id, exp *exp_id,* gt *got_id,*, nt *network ID*

Long Syntax: PPP.027 Bad lcp ack id, expected *exp_id,*, got *got_id,*, on network *network ID*

Description: lcp_ack got config ack with bad id.

PPP.028

Level: UE-ERROR

Short Syntax: PPP.028 Bd lcp ack lngth, nt *network ID*

Long Syntax: PPP.028 Bad lcp ack length, on network *network ID*

Description: lcp_ack got config ack with bad length.

PPP.029

Level: UE-ERROR

Short Syntax: PPP.029 msmtchd lcp ack, nt *network ID*

Long Syntax: PPP.029 mis-matched data in lcp ack, on network *network ID*

Description: lcp_ack got ack whose data doesn't match our request.

PPP.030

Level: UE-ERROR

Short Syntax: PPP.030 Bd lcp nak id, exp *exp_id,* gt *got_id,*, nt *network ID*

Long Syntax: PPP.030 Bad LCP nak id, expected *exp_id,*, got *got_id,*, on network *network ID*

Description: lcp_nak got nak with bad id.

PPP.031

Level: UE-ERROR

Short Syntax: PPP.031 Bd lcp nak lngth, nt *network ID*

Long Syntax: PPP.031 Bad LCP nak length, on network *network ID*

Description: lcp_nak got nak with bad length.

PPP.032

Level: UE-ERROR

Short Syntax: PPP.032 Bd lcp nak opt *lcp_option,*, nt *network ID*

Long Syntax: PPP.032 Bad LCP nak option = *lcp_option,*, on network *network ID*

Description: lcp_nak got nak containing out-of-range option.

PPP.033

Level: UE-ERROR

Short Syntax: PPP.033 out-ordr lcp nak opt *lcp_option,*, nt *network ID*

Long Syntax: PPP.033 Bad LCP nak option = *lcp_option,*, on network *network ID*

Description: lcp_nak got nak containing out-of-order option.

PPP.034

Level: UE-ERROR

Short Syntax: PPP.034 Bd lcp nak opt *lcp_option,* shrt, nt *network ID*

Long Syntax: PPP.034 Bad LCP nak option = *lcp_option,*, data too short, on network *network ID*

Description: lcp_nak got nak containing option with insufficient data.

PPP.035

Level: P_TRACE

Short Syntax: PPP.035 mk mru *mru*

Long Syntax: PPP.035 making max receive unit with value *mru*

Description: lcp_option built mru.

PPP.036

Level: P_TRACE

Short Syntax: PPP.036 mk accm 0x *accm*

Long Syntax: PPP.036 making accm = 0x *accm*

Description: lcp_option built accm.

PPP.037

Level: P_TRACE

Short Syntax: PPP.037 mk aut 0x *auth*

Long Syntax: PPP.037 making authorization protocol with value 0x *auth*

Description: lcp_option built authorization.

PPP.038

Level: P_TRACE

Short Syntax: PPP.038 mk mag 0x *magic_number*

Long Syntax: PPP.038 making magic number with value 0x *magic_number*

Description: lcp_option built magic number.

PPP.039

Level: P_TRACE

Short Syntax: PPP.039 mk pfc

Long Syntax: PPP.039 making protocol compression

Description: lcp_option built protocol compression.

PPP.040

Level: P_TRACE

Short Syntax: PPP.040 mk acfc

Long Syntax: PPP.040 making address/control field compression

Description: lcp_option built address/control compression.

PPP.041

Level: P_TRACE

Short Syntax: PPP.041 mk qp 0x *protocol, period*

Long Syntax: PPP.041 making quality protocol = 0x *protocol,*, period = *period*

Description: lcp_option built quality.

PPP.042

Level: P_TRACE

Short Syntax: PPP.042 mk fcs

Long Syntax: PPP.042 making 32-bit fcs

Description: lcp_option built 32-bit fcs.

PPP.043

Level: P_TRACE

Short Syntax: PPP.043 mk lcp unk *option*

Long Syntax: PPP.043 making unknown lcp option *option*

Description: lcp_option built an unrecognized option.

PPP.044

Level: P_TRACE

Short Syntax: PPP.044 ck mru *mru*

Long Syntax: PPP.044 checking max receive unit with value *mru*

Description: lcp_check processed mru.

PPP.045

Level: P_TRACE

Short Syntax: PPP.045 ck accm 0x *accm*

Long Syntax: PPP.045 checking accm = 0x *accm*

Description: lcp_check processed accm.

PPP.046

Level: P_TRACE

Short Syntax: PPP.046 ck aut 0x *auth*

Long Syntax: PPP.046 checking authorization protocol with value 0x *auth*

Description: lcp_check processed authorization.

PPP.047

Level: P_TRACE

Short Syntax: PPP.047 ck mag 0x *magic_number*

Long Syntax: PPP.047 checking magic number with value 0x *magic_number*

Description: lcp_check processed magic number.

PPP.048

Level: P_TRACE

Short Syntax: PPP.048 ck pfc

Long Syntax: PPP.048 checking protocol compression

Description: lcp_check processed protocol compression.

PPP.049

Level: P_TRACE

Short Syntax: PPP.049 ck acfc

Long Syntax: PPP.049 checking address/control field compression

Description: lcp_check processed address/control compression.

PPP.050

Level: P_TRACE

Short Syntax: PPP.050 ck qp 0x *protocol, period*

Long Syntax: PPP.050 checking quality protocol = 0x *protocol*,, period = *period*

Description: lcp_check processed quality.

PPP.051

Level: P_TRACE

Short Syntax: PPP.051 ck fcs

Long Syntax: PPP.051 checking 32-bit fcs

Description: lcp_check processed 32-bit fcs.

PPP.052

Level: P_TRACE

Short Syntax: PPP.052 ck lcp unk *option*

Long Syntax: PPP.052 checking unknown lcp option *option*

Description: lcp_check processed an unrecognized option.

PPP.053

Level: C-TRACE

Short Syntax: PPP.053 *state, routine_name,* nt *network ID*

Long Syntax: PPP.053 state = *state*,, called *routine_name*,, on network *network ID*

Description: Called the specified cp routine.

PPP.054

Level: UE-ERROR

Short Syntax: PPP.054 Bd *control_protocol,* ack id, exp *exp_id,* gt *got_id*,, nt *network ID*

Long Syntax: PPP.054 Bad *control_protocol,* ack id, expected *exp_id*,, got *got_id*,, on network *network ID*

Description: xxcp_ack got config ack with bad id.

PPP.055

Level: UE-ERROR

Short Syntax: PPP.055 Bad *control_protocol,* ack lngth nt *network ID*

Long Syntax: PPP.055 Bad *control_protocol,* ack length on network *network ID*

Description: xxcp_ack got config ack with bad length.

PPP.056

Level: UE-ERROR

Short Syntax: PPP.056 msmtchd *control_protocol,* ack, nt *network ID*

Long Syntax: PPP.056 mis-matched data in *control_protocol,* ack, on network *network ID*

Description: xxcp_ack got ack whose data doesn't match our request.

PPP.057

Level: UE-ERROR

Short Syntax: PPP.057 Bd *control_protocol,* nak id, exp *exp_id,* gt *got_id,,* nt *network ID*

Long Syntax: PPP.057 Bad *control_protocol,* nak id, expected *exp_id,,* got *got_id,,* on network *network ID*

Description: xxcp_nak got nak with bad id.

PPP.058

Level: UE-ERROR

Short Syntax: PPP.058 Bd *control_protocol,* nak lngth, nt *network ID*

Long Syntax: PPP.058 Bad *control_protocol,* nak length, on network *network ID*

Description: xxcp_nak got nak with bad length.

PPP.059

Level: UE-ERROR

Short Syntax: PPP.059 Bd *control_protocol,* nak opt *ipcp_option,,* nt *network ID*

Long Syntax: PPP.059 Bad *control_protocol,* nak option = *ipcp_option,,* on network *network ID*

Description: xxcp_nak got nak containing out-of-range option.

PPP.060

Level: UE-ERROR

Short Syntax: PPP.060 out-ordr *control_protocol,* nak opt *ipcp_option,,* nt *network ID*

Long Syntax: PPP.060 Bad *control_protocol,* nak option = *ipcp_option,,* on network *network ID*

Description: xxcp_nak got nak containing out-of-order option.

PPP.061

Level: UE-ERROR

Short Syntax: PPP.061 Bd *control_protocol,* nak opt *ipcp_option,* shrt, nt *network ID*

Long Syntax: PPP.061 Bad *control_protocol,* nak option = *ipcp_option,,* data too short, on network *network ID*

Description: xxcp_nak got nak containing option with insufficient data.

PPP.062

Level: UE-ERROR

Short Syntax: PPP.062 Bd *control_protocol,* rej id, exp *exp_id,* gt *got_id,,* nt *network ID*

Long Syntax: PPP.062 Bad *control_protocol,* reject id, expected *exp_id,,* got *got_id,,* on network *network ID*

Description: xxcp_rej got reject with bad id.

PPP.063

Level: UE-ERROR

Short Syntax: PPP.063 Bd *control_protocol,* rej lngth, nt *network ID*

Long Syntax: PPP.063 Bad *control_protocol,* reject length, on network *network ID*

Description: xxcp_rej got reject with bad length.

PPP.064

Level: UE-ERROR

Short Syntax: PPP.064 Bd *control_protocol,* rej opt *ipcp_option,,* nt *network ID*

Long Syntax: PPP.064 Bad *control_protocol,* reject option = *ipcp_option,,* on network *network ID*

Description: xxcp_rej got reject containing out-of-range option.

PPP.065

Level: UE-ERROR

Short Syntax: PPP.065 out-ordr *control_protocol,* rej opt *ipcp_option,,* nt *network ID*

Long Syntax: PPP.065 Bad *control_protocol,* reject option = *ipcp_option,,* on network *network ID*

Description: xxcp_rej got reject containing out-of-order option.

PPP.066

Level: P_TRACE

Short Syntax: PPP.066 mk ads *src_addr, dest_addr*

Long Syntax: PPP.066 making IPCP addresses option, addresses = *src_addr, dest_addr*

Description: ipcp_option built (deprecated) IPCP addresses option.

PPP.067

Level: P_TRACE

Short Syntax: PPP.067 mk cmp 0x *comp_protocol, slots,/ slot_compress*

Long Syntax: PPP.067 making compression option 0x *comp_protocol*,, slots = *slots*,, slot_compress = *slot_compress*

Description: ipcp_option built compression option.

PPP.068

Level: P_TRACE

Short Syntax: PPP.068 mk ad *ip_address*

Long Syntax: PPP.068 making IPCP address option, address = *ip_address*

Description: ipcp_option built ipcp address option.

PPP.069

Level: P_TRACE

Short Syntax: PPP.069 mk *protocol,* unk *option*

Long Syntax: PPP.069 making unknown *protocol,* option *option*

Description: {ipcp,dncp}_option built unrecognized option.

PPP.070

Level: P_TRACE

Short Syntax: PPP.070 ck ads *src_addr, dest_addr*,, nt *network ID*

Long Syntax: PPP.070 checking IPCP addresses option, addresses = *src_addr, dest_addr*,, on network *network ID*

Description: ipcp_check processed (deprecated) IPCP addresses option.

PPP.071

Level: P_TRACE

Short Syntax: PPP.071 ck cmp 0x *comp_protocol, slots,/ slot_compress*,, nt *network ID*

Long Syntax: PPP.071 checking compression option 0x *comp_protocol*,, slots = *slots*,, slot_compress = *slot_compress*,, on network *network ID*

Description: ipcp_check processed compression option.

PPP.072

Level: P_TRACE

Short Syntax: PPP.072 ck ad *ip_address*,, nt *network ID*

Long Syntax: PPP.072 checking IPCP address option, address = *ip_address*,, on network *network ID*

Description: ipcp_check processed ipcp address option.

PPP.073

Level: P_TRACE

Short Syntax: PPP.073 ck *control_protocol,* unk *option*

Long Syntax: PPP.073 checking unknown *control_protocol,* option *option*

Description: ipcp_check processed ipcp unrecognized option.

PPP.074

Level: UE-ERROR

Short Syntax: PPP.074 Bd *control_protocol,* req hdr lngth, nt *network ID*

Long Syntax: PPP.074 Bad *control_protocol,* request header length, on network *network ID*

Description: xxcp_req got request with bad header length.

PPP.075

Level: UE-ERROR

Short Syntax: PPP.075 Bd *control_protocol,* req opt *ipcp_option,* shrt, nt *network ID*

Long Syntax: PPP.075 Bad *control_protocol,* request option = *ipcp_option*,, data too short, on network *network ID*

Description: xxcp_req got request containing option with insufficient data.

PPP.076

Level: C-TRACE

Short Syntax: PPP.076 *control_protocol,* req rslt: *ipcp_rslt*,, opt *ipcp_option*,, ln *opt_len*,, nt *network ID*

Long Syntax: PPP.076 *control_protocol,* request result:

ipcp_rslt,, option = *ipcp_option*,, length = *opt_len*,, on network *network ID*

Description: Result, so far, of processing one option.

PPP.077

Level: UE-ERROR

Short Syntax: PPP.077 bd rcv len, pk *hdr_len,* dr *i_bxfr,* nt *network ID*

Long Syntax: PPP.077 bad length on received data, packet length = *hdr_len*,, driver says *i_bxfr*,, on network *network ID*

Description: The length field of an LCP, AP, or NCP packet didn't match the i_bxfr of the iorb.

PPP.078

Level: C-INFO

Short Syntax: PPP.078 Mdm sts chg, DCD *dcd* CTS *cts* nt *network ID*

Long Syntax: PPP.078 Modem status changed DCD = *dcd* CTS = *cts* on network *network ID*

Description: A modem status change has occurred. The present state is described.

PPP.079

Level: UE-ERROR

Short Syntax: PPP.079 prt rej rcv, prt 0x *protocol*) nt *network ID*

Long Syntax: PPP.079 protocol reject received for protocol 0x *protocol*) on network *network ID*

Description: Got a protocol reject packet from the link.

PPP.080

Level: UE-ERROR

Short Syntax: PPP.080 rc bd cd *packet_type,* prt *prot_type,* nt *network ID*

Long Syntax: PPP.080 Received bad code (*packet_type*,) for prot *prot_type*,, on network *network ID*

Description: A packet from the net had a type which is not supported for that protocol.

PPP.081

Level: UE-ERROR

Short Syntax: PPP.081 rc bd mgc 0x *rcv_magic_num*,, ours 0x *our_magic_num,* nt *network ID*

Long Syntax: PPP.081 Received bad magic number 0x *rcv_magic_num*,, ours is 0x *our_magic_num*,, on network *network ID*

Description: Didn't get magic number we wanted. If we got our own (the two args match) the link is looped back.

PPP.082

Level: UE-ERROR

Short Syntax: PPP.082 lpbk nt *network ID*

Long Syntax: PPP.082 link appears to be looped back on network *network ID*

Description: Excessive magic number collisions while trying to configure link.

PPP.083

Level: UI-ERROR

Short Syntax: PPP.083 Srl prt fl: 0x *status,* nt *network ID*

Long Syntax: PPP.083 Serial port failed init, stat: 0x *status*,, network *network ID*

Description: ppp_slftst2 observed bad status in (netp->n_idctp)->d_flg after init.

PPP.084

Level: C-INFO

Short Syntax: PPP.084 Req brng up DN nt *network ID*

Long Syntax: PPP.084 Request to bring up DECNET IV, on network *network ID*

Description: ppp_prinit routine called for Decnet IV protocol.

PPP.085

Level: UE-ERROR

Short Syntax: PPP.085 rc no mgc nt *network ID*

Long Syntax: PPP.085 Received message without a magic number, on network *network ID*

Description: A received packet which should have had a magic number (ECHO REQ, ECHO ACK, QUALITY REPORT, DISC REQ), didn't.

PPP.086

Level: C-INFO

Short Syntax: PPP.086 Req brng up IPX nt *network ID*

Long Syntax: PPP.086 Request to bring up IPX, on network *network ID*

Description: ppp_prinit routine called for IPX protocol.

PPP.087

Level: C-INFO

Short Syntax: PPP.087 Req brng up SRT nt *network ID*

Long Syntax: PPP.087 Request to bring up SRT, on network *network ID*

Description: ppp_prinit routine called for SRT protocol.

PPP.088

Level: C-INFO

Short Syntax: PPP.088 BNCP changed SR seg num from *oldsegnum* to *newsegnum* on port *port* nt *network ID*

Long Syntax: PPP.088 Bridging control protocol changed source route segment number from *oldsegnum* to *newsegnum* on port *port*, network *network ID*

Description: As a result of negotiating the source route line ID, the local side of the link changed its source route segment number.

PPP.089

Level: C-TRACE

Short Syntax: PPP.089 DROP: rcvd STB bdgd pkt but bdging dsbld on nt *network*

Long Syntax: PPP.089 Dropping the received Spanning Tree Bridged packet but bridging is disabled on network *network*

Description: A Bridged packet is received on this PPP interface even though the Spanning Tree Bridging is not enabled on this interface or STB is disabled in the box.

PPP.090

Level: C-INFO

Short Syntax: PPP.090 Req brng up AppleTalk nt *network ID*

Long Syntax: PPP.090 Request to bring up AppleTalk, on network *network ID*

Description: ppp_prinit routine called for Appletalk protocol.

PPP.091

Level: UE-ERROR

Short Syntax: PPP.091 ATCP add opt rejected on nt *network ID* - no common net num

Long Syntax: PPP.091 ATCP address option rejected on network *network ID* - no common network number

Description: An ATCP configuration reject will be sent because the AppleTalk Address option did not contain a common network number for the PPP link.

PPP.092

Level: UE-ERROR

Short Syntax: PPP.092 ATCP add opt rejected on nt *network ID* - remote's node ID invalid *node_id*

Long Syntax: PPP.092 ATCP address option rejected on network *network ID* - remote side's node ID is invalid *node_id*

Description: An ATCP configuration reject will be sent because the AppleTalk Address option from the remote side contained an invalid node ID.

PPP.093

Level: C-INFO

Short Syntax: PPP.093 Req brng up OSI nt *network ID*

Long Syntax: PPP.093 Request to bring up OSI, on network *network ID*

Description: ppp_prinit routine called for OSI protocol.

PPP.094

Level: C-TRACE

Short Syntax: PPP.094 CCP rec reset-req nt *network ID*

Long Syntax: PPP.094 CCP received compression reset-req on network *network ID*

Description: CCP received a reset request from the remote host. This is likely due to lost or corrupted packets.

PPP.095

Level: C-TRACE

Short Syntax: PPP.095 CCP snd reset-req nt *network ID*

Long Syntax: PPP.095 CCP sent compression reset-req on network *network ID*

Description: CCP sent a reset request to the remote host. This is due to lost or corrupted packets.

PPP.096

Level: C-TRACE

Short Syntax: PPP.096 CCP rec reset-ack nt *network ID*

Long Syntax: PPP.096 CCP received compression reset-ack on network *network ID*

Description: CCP received a reset acknowledge from the remote host.

PPP.097

Level: UE-ERROR

Short Syntax: PPP.097 Bd *control_protocol,* reset-ack id, exp *exp_id,* gt *got_id,*, nt *network ID*

Long Syntax: PPP.097 Bad *control_protocol,* reset-ack id, expected *exp_id,*, got *got_id,*, on network *network ID*

Description: xxcp_reset_ack got reset ack with bad id.

PPP.098

Level: UE-ERROR

Short Syntax: PPP.098 Bad *alg_name,* seq/FCS, exp *exp_id,* gt *got_id,*, nt *network ID*

Long Syntax: PPP.098 *alg_name,* decompress, bad sequence or FCS, expected *exp_id,*, got *got_id,*, on network *network ID*

Description: A compressed packet was receive with an unexpected sequence number or a check value that doesn't match the received data. This is due to missing or corrupted packets.

PPP.099

Level: C-TRACE

Short Syntax: PPP.099 CCP mk *opt_id,* sz *len,* opt *optval*

Long Syntax: PPP.099 CCP make option *opt_id,* length *len,* optionval *optval*

Description: CCP created an option of this type.

PPP.100

Level: UE-ERROR

Short Syntax: PPP.100 CCP bad packet nt *network ID*

Long Syntax: PPP.100 CCP decompressor dropped a bad packet, network *network ID*

Description: PPP data decompress dropped a bad packet.

PPP.101

Level: C-INFO

Short Syntax: PPP.101 ccinit *typename,* will *will_neg,* mem *mem,*, nt *network ID*.

Long Syntax: PPP.101 CCP init: *typename,* will_negotiate *will_neg,* maxmem *mem,*, net *network ID*.

Description: Boot time list of CCP's available compressors and their cost.

PPP.103

Level: C-INFO

Short Syntax: PPP.103 CCP dis nt *network ID*.

Long Syntax: PPP.103 CCP data compression disabled at boot time, net *network ID*.

Description: CCP data compression is disabled on this interface.

PPP.104

Level: C-TRACE

Short Syntax: PPP.104 uncmp pkt; cmp len *cmp_len,* > orig *orig_len,* (*action,*); nt *network ID*

Long Syntax: PPP.104 Uncompressible packet: compressed len *cmp_len,* >= uncompressed len *orig_len,* (*action,*), nt *network ID*.

Description: CCP Compressor found an incompressible packet. Normally the original uncompressed packet is sent instead.

PPP.105

Level: C-TRACE

Short Syntax: PPP.105 CCP have *proto,* got *opt,* (*ob1, ob2, ob3*)

Long Syntax: PPP.105 CCP proto *proto,* option is *opt,* (*ob1, ob2, ob3*).

Description: Another router sent a configuration request containing options.

PPP.106

Level: UE-ERROR

Short Syntax: PPP.106 CDP gnt pkt *got,* (> *mru*).

Long Syntax: PPP.106 CDP saw a giant packet of length *got,* (> *mru*).

Description: The router received a compressed packet with too much data.

PPP.107

Level: C-TRACE

Short Syntax: PPP.107 LZS pkt after reset (*res_id,*) nt *network ID*.

Long Syntax: PPP.107 LZS compression received a packet after reset (*res_id,*) net *network ID*.

Description: A packet was discarded while waiting for Reset Acknowledge.

PPP.108

Level: UI-ERROR

Short Syntax: PPP.108 Ignoring extra IP addr: addr = *ip_address* nt *network ID*

Long Syntax: PPP.108 Ignoring multiple IP addresses configured on single PPP link, local address = *ip_address*, on network *network ID*

Description: IP/PPP can only support one IP address per PPP interface. When you configure multiple IP addresses on a single PPP interface, the router ignores all but the first IP address. This message indicates that the router is ignoring an IP address.

PPP.109

Level: UI-ERROR

Short Syntax: PPP.109 CCP rcv CODE_REJ *code,* nt *network ID*.

Long Syntax: PPP.109 CCP received CODE_REJ for code *code,*, net *network ID*.

Description: CCP received a CODE_REJect for a CCP packet. Code 14 is RESET-REQ and remote hosts not supporting compression reset may reject it. The router terminates the CCP connection and may restart on its own.

PPP.110

Level: C-INFO

Short Syntax: PPP.110 CCP *dir,* no buf net *network ID*.

Long Syntax: PPP.110 CCP *dir,*put no buffers available net *network ID*.

Description: CCP tried to allocate an input or output buffer and failed.

PPP.111

Level: C-TRACE

Short Syntax: PPP.111 CCP R-req timeout nt *network ID*.

Long Syntax: PPP.111 CCP Reset-req timeout expired net *network ID*.

Description: CCP sent a reset request and timed out waiting for an acknowledgement.

PPP.112

Level: C-INFO

Short Syntax: PPP.112 Req brng up APPN ISR nt *network ID*

Long Syntax: PPP.112 Request to bring up APPN ISR, on network *network ID*

Description: ppp_prinit routine called for APPN ISR protocol.

PPP.113

Level: C-INFO

Short Syntax: PPP.113 Req brng up APPN HPR nt *network ID*

Long Syntax: PPP.113 Request to bring up APPN HPR, on network *network ID*

Description: ppp_prinit routine called for APPN HPR protocol.

PPP.114

Level: C-INFO

Short Syntax: PPP.114 *message*

Long Syntax: PPP.114 *message*

Description: Special event messages - used for internal development.

PPP.116

Level: C-INFO

Short Syntax: PPP.116 PAP Rcv *packet_type,* nt *network ID*.

Long Syntax: PPP.116 PAP Received *packet_type,* net *network ID*.

Description: PAP received an ACK or NAK on an authentication request.

PPP.117

Level: UE-ERROR

Short Syntax: PPP.117 Bd id, exp *exp_id,* gt *got_id,*, nt *network ID*

Long Syntax: PPP.117 Bad id, expected *exp_id,*, got *got_id,*, on network *network ID*

Description: PAP or CHAP or MSCHAP packet with id different than expected.

PPP.118

Level: UE-ERROR

Short Syntax: PPP.118 *protocol,* Bd typ *type,* nt *network ID*.

Long Syntax: PPP.118 *protocol,* Bad Packet Type *type,* net *network ID*.

Description: PAP or CHAP or MSCHAP got a packet type that was illegal.

PPP.119

Level: P-TRACE

Short Syntax: PPP.119 CHAP rcv pkt *packet_type*, nt *network ID*.

Long Syntax: PPP.119 CHAP receive packet type *packet_type*, on net *network ID*.

Description: CHAP or MSCHAP received a packet.

PPP.120

Level: P-TRACE

Short Syntax: PPP.120 CHAP snd pkt *packet_type*, nt *network ID*.

Long Syntax: PPP.120 CHAP sent packet type *packet_type*, on net *network ID*.

Description: CHAP or MSCHAP sent a packet.

PPP.121

Level: UE-ERROR

Short Syntax: PPP.121 CHAP bad len nt *network ID*.

Long Syntax: PPP.121 CHAP bad response length net *network ID*.

Description: CHAP or MSCHAP received a response packet that was too short.

PPP.124

Level: C-INFO

Short Syntax: PPP.124 *protocol*, msg *message*, nt *network ID*.

Long Syntax: PPP.124 *protocol*, messsage *message*, net *network ID*.

Description: PAP or CHAP or MSCHAP reply packet contained a plaintext message.

PPP.125

Level: C-INFO

Short Syntax: PPP.125 CHAP CHAL *direction*, name= *name*, nt *network ID*.

Long Syntax: PPP.125 CHAP CHAL *direction*, name= *name*, nt *network ID*.

Description: A CHAP or MSCHAP challenge was issued or received.

PPP.126

Level: C-INFO

Short Syntax: PPP.126 CHAP RESP *direction*, name= *name*, nt *network ID*.

Long Syntax: PPP.126 CHAP RESP *direction*, name= *name*, nt *network ID*.

Description: A response to a CHAP or MSCHAP challenge was issued or received.

PPP.127

Level: C-INFO

Short Syntax: PPP.127 PAP REQ *direction*, name= *name*, nt *network ID*.

Long Syntax: PPP.127 PAP REQ *direction*, name= *name*, nt *network ID*.

Description: A PAP request was issued or received.

PPP.128

Level: CE-ERROR

Short Syntax: PPP.128 Rcvd illegal *protocol*, nt *network ID*.

Long Syntax: PPP.128 Rcvd illegal *protocol*, nt *network ID*.

Description: received a PAP or CHAP or MSCHAP packet that did not correspond to the negotiated options.

PPP.129

Level: C-TRACE

Short Syntax: PPP.129 *protocol*, *flag*, done nt *network ID*.

Long Syntax: PPP.129 *protocol*, *flag*, Authentication complete net *network ID*.

Description: Either local or remote authentication completed successfully.

PPP.130

Level: C-INFO

Short Syntax: PPP.130 Auth done nt *network ID*.

Long Syntax: PPP.130 Authentication completed successfully on net *network ID*.

Description: All authentication completed successfully.

PPP.131

Level: C-TRACE

Short Syntax: PPP.131 *protocol*, *fcn_name*, nt *network ID*.

Long Syntax: PPP.131 *protocol*, *fcn_name*, net *network ID*.

Description: Called the specified authentication function.

PPP.132

Level: C-TRACE

Short Syntax: PPP.132 No name cfgrd nt *network ID*.

Long Syntax: PPP.132 No name configured net *network ID*.

Description: No name is configured on this PPP interface

PPP.133

Level: P-TRACE

Short Syntax: PPP.133 LCP Ident: *message*, nt *network ID*.

Long Syntax: PPP.133 LCP Identification: *message*, on net *network ID*

Description: LCP Identification Packet Received

PPP.134

Level: C-INFO

Short Syntax: PPP.134 *seconds*, seconds remaining pkt typ *packet_type*, nt *network ID*.

Long Syntax: PPP.134 *seconds*, seconds remaining packet type *packet_type*, on net *network ID*

Description: TIME REMAINING Packet Received

PPP.135

Level: P-TRACE

Short Syntax: PPP.135 time rem start *seconds*, seconds net *network ID*.

Long Syntax: PPP.135 Time Remaining Started with *seconds*, Seconds Remaining net *network ID*.

Description: Time Remaining Started

PPP.136

Level: P-TRACE

Short Syntax: PPP.136 time rem sent *seconds*, seconds left net *network ID*..

Long Syntax: PPP.136 Time Remaining Packet Sent *seconds*, Seconds Remaining net *network ID*

Description: TIME REMAINING Packet Sent

PPP.137

Level: C-INFO

Short Syntax: PPP.137 No Time Remaining! Forced Shutdown net *network ID*.

Long Syntax: PPP.137 No Time Remaining! Forced Shutdown net *network ID*.

Description: No Time Remaining! Forced Shutdown

PPP.138

Level: P-TRACE

Short Syntax: PPP.138 *protocol*, snd pkt *packet_type*, id *id*, nt *network ID*.

Long Syntax: PPP.138 *protocol*, sent packet type *packet_type*, id= *id*, on net *network ID*.

Description: Authentication protocol sent a packet.

PPP.139

Level: P-TRACE

Short Syntax: PPP.139 *protocol*, rcv pkt *packet_type*, nt *network ID*.

Long Syntax: PPP.139 *protocol*, receive packet type *packet_type*, on net *network ID*.

Description: Authentication protocol received a packet.

PPP.140

Level: C-INFO

Short Syntax: PPP.140 *protocol*, Rcv *packet_type*, nt *network ID*.

Long Syntax: PPP.140 *protocol*, Received *packet_type*, net *network ID*.

Description: Authentication protocol received an ACK or NAK on an authentication request.

PPP.141

Level: P-TRACE

Short Syntax: PPP.141 Rcvd *what*, prtcl=0x *protocol*), len= *length*, nt *network ID*.

Long Syntax: PPP.141 Packet (*what*) Received, protocol=0x *protocol*), length= *length*, net *network ID*.

Description: PPP received a packet. The "what" parameter indicates whether the packet was really received as a regular packet "pkt" or whether it was received as a compressed data packet "CDP". The protocol and lengths shown are after decompression has taken place.

PPP.143

Level: P-TRACE

Short Syntax: PPP.143 Unlimited Seconds Remaining for net *network ID*.

Long Syntax: PPP.143 Unlimited Seconds Remaining net *network ID*.

Description: Unlimited Time Remaining

PPP.144

Level: P-TRACE

Short Syntax: PPP.144 Unlimited Time Remaining Packet Sent net *network ID*..

Long Syntax: PPP.144 Unlimited Time Remaining Packet Sent net *network ID*

Description: Unlimited Time Remaining Packet Sent

PPP.145

Level: U-INFO

Short Syntax: PPP.145 Test # *test_number* triggered, nt *network ID*. --> *test_description*

Long Syntax: PPP.145 Test # *test_number* triggered on net *network ID*. Description: *test_description*

Description: A special diagnostic test has been triggered.

PPP.146

Level: UE-ERROR

Short Syntax: PPP.146 CDP decomp err: data exceeds MRU, nt *network ID*.

Long Syntax: PPP.146 CDP decompression error; expanded data length > MRU, net *network ID*

Description: Data decompression expanded a packet to produce a PPP information field which exceeds the negotiated MRU value. This could be due to a problem at the sending end rather than the receiver, or due to corrupt data in the received packet.

PPP.147

Level: U-INFO

Short Syntax: PPP.147 MRU reduced (was *old_mru*,, now *new_mru*,), nt *network ID*.

Long Syntax: PPP.147 MRU has been reduced, from *old_mru*, to *new_mru*,, net *network ID*.

Description: The MRU value being used on a PPP link has been reduced from the configured value, because the underlying link won't support the configured MRU. The base cause of this is that the underlying link framesize is not large enough to contain a complete PPP packet with the specified MRU. This most likely will occur on an ISDN dial circuit, where the framesize configured for the ISDN base net is too small for PPP packets with the specified MRU to be sent.

Action: As long as the new MRU value is acceptable, no action needed. Otherwise, reconfigure the PPP interface to have a smaller MRU which fits in the base link framing, or increase the framesize parameters for the underlying link to handle the specified PPP MRU value.

PPP.148

Level: UI-ERROR

Short Syntax: PPP.148 Init MRU= *mru*, too small (<1500) for PPP nt *network ID*.

Long Syntax: PPP.148 Initial MRU value of *mru*, is too small, net *network ID*.

Description: The initial MRU value being used on a PPP link is too small to allow proper operation of the link. This error indicates that the internal input data buffers are too small to receive PPP frames with 1500 bytes of data. PPP requires the ability to handle 1500 bytes of data -- smaller MRU values can be negotiated via LCP, but until this is done the MRU is 1500. The base cause of this message is that the underlying link framesize is not large enough to contain a complete PPP packet with the default MRU size of 1500. Note that the problem here is *not* with the configured PPP MRU value, as this is merely the value which gets negotiated via LCP and can ultimately be less than 1500; instead, the problem is that the data buffers aren't large enough for PPP to revert to 1500 byte packets in case of loss of sync or renegotiation of the MRU. The network will probably function though as long as a smaller MRU is negotiated, since control packets would rarely be long enough to pose a problem. This most likely will occur on an ISDN dial circuit, where the framesize configured for the ISDN base net is too small to carry a PPP packet with 1500 data bytes.

Action: Reconfigure the underlying link parameters (such as the ISDN framesize). If the problem persists, contact customer service.

PPP.149

Level: P-TRACE

Short Syntax: PPP.149 Rcv pkt discard, rsn= *reason*,, nt *network ID*

Long Syntax: PPP.149 Input packet discarded, reason= *reason*,, nt *network ID*

Description: PPP discarded a packet it received.

Action: None; informational message only.

PPP.150

Level: P_TRACE

Short Syntax: PPP.150 Pkt data= *data*... nt *network ID*

Long Syntax: PPP.150 Packet data= *data*... net *network ID*

Description: This informational message simply displays the first several bytes of data in a packet. It always comes out in conjunction with some other ELS message and should never be produced as a standalone message. The exact data displayed, and where it comes from within the packet, are dependent on the event which resulted in producing this message.

Action: None; informational message only.

PPP.151

Level: C-INFO

Short Syntax: PPP.151 Net dwn, *why,*, nt *network ID*

Long Syntax: PPP.151 Signalling a net down on network, cause= *why,*, *network ID*

Description: PPP signalling a net-down event to higher layers.

PPP.152

Level: C_INFO

Short Syntax: PPP.152 Effective MRU changed from *old_mru,* to *new_mru*,, nt *network ID*

Long Syntax: PPP.152 Effective MRU changed from *old_mru,* to *new_mru*,, net *network ID*

Description: The effective MRU has changed on a link which is already marked as 'up'; PPP is signalling a special 'net up' type of event to indicate that the MRU size has altered. This typically occurs when encryption is activated, or on dial- on-demand circuits when the ends negotiate a value for the MRU which differs from the configured value.

PPP.153

Level: P-TRACE

Short Syntax: PPP.153 Sent pkt, prtcl=0x *protocol*), len= *length*, rc= *rc* (*status*), nt *network ID*.

Long Syntax: PPP.153 Packet Sent, protocol=0x *protocol*), length= *length*, status= *rc* (*status*), net *network ID*.

Description: PPP sent a packet. This actually means it handed off the packet to be delivered by the underlying device driver. It is possible that the device driver or the Bandwidth Reservation system blocked the actual transmission of the packet - this would be indicated by a non-zero value for the status (return code) value. The protocol and length values are the values prior to data compression, if a packet is sent in compressed form.

PPP.154

Level: C-INFO

Short Syntax: PPP.154 CCP start cmp *algorithm options*, nt *network ID*

Long Syntax: PPP.154 CCP start compressor *algorithm options*, on network *network ID*

Description: CCP has successfully negotiated a compression algorithm.

PPP.155

Level: C-INFO

Short Syntax: PPP.155 CCP start dcmp *algorithm options*, nt *network ID*

Long Syntax: PPP.155 CCP start decompressor *algorithm options*, on network *network ID*

Description: CCP has successfully negotiated a decompression algorithm.

PPP.156

Level: C-INFO

Short Syntax: PPP.156 CCP stop cmp *algorithm*, nt *network ID*

Long Syntax: PPP.156 CCP stop compressor *algorithm*, on network *network ID*

Description: CCP has shutdown compression.

PPP.157

Level: C-INFO

Short Syntax: PPP.157 CCP stop dcmp *algorithm*, nt *network ID*

Long Syntax: PPP.157 CCP stop decompressor *algorithm*, on network *network ID*

Description: CCP has shutdown decompression.

PPP.158

Level: C-INFO

Short Syntax: PPP.158 PPP net down, nt *network ID*

Long Syntax: PPP.158 PPP net down, on network *network ID*

Description: The PPP link has gone down. This may be due to an externally signalled event, or due to some internally generated PPP event which will be reported via PPP_151.

PPP.159

Level: C-INFO

Short Syntax: PPP.159 PPP link down (disconnect on dial link), nt *network ID*

Long Syntax: PPP.159 PPP link down (disconnect on dial link), on network *network ID*

Description: The PPP link has gone down on a dial circuit. This differs from event PPP_158 in that the link is down from PPP's perspective, but the network interface is still up from the layer-3 protocols' perspective (for example, on a dial-on-demand circuit which has disconnected because the ISDN idle timer has expired).

PPP.160

Level: C-INFO

Short Syntax: PPP.160 PPP link disabled, nt *network ID*

Long Syntax: PPP.160 PPP link disabled, on network *network ID*

Description: A PPP link was disabled.

PPP.161

Level: P_TRACE

Short Syntax: PPP.161 ck mrru *mrru*

Long Syntax: PPP.161 checking max reconstructed receive unit with value *mrru*

Description: lcp_check processed mrru.

PPP.162

Level: P_TRACE

Short Syntax: PPP.162 ck short seqnos

Long Syntax: PPP.162 checking short sequence numbers

Description: lcp_check processed short seqnos.

PPP.163

Level: P_TRACE

Short Syntax: PPP.163 ck endpt discr. cls= *class* addr= *addr*

Long Syntax: PPP.163 checking endpoint discriminator class= *class*,addr= *addr*

Description: lcp_check processed endpoint discriminator.

PPP.164

Level: P_TRACE

Short Syntax: PPP.164 ck link discr= *LD*

Long Syntax: PPP.164 checking link discriminator = *LD*

Description: lcp_check processed link discriminator.

PPP.165

Level: C-INFO

Short Syntax: PPP.165 Assigning IP Address *ip_address*, nt *network ID*

Long Syntax: PPP.165 Assigning IP Address *ip_address*, nt *network ID*

Description: IP address for IPCP negotation assigned

PPP.167

Level: C-TRACE

Short Syntax: PPP.167 ECP mk *opt_id*, sz *len*, opt *optval*

Long Syntax: PPP.167 ECP make option *opt_id*, length *len*, optionval *optval*

Description: ECP created an option of this type.

PPP.168

Level: C-TRACE

Short Syntax: PPP.168 ECP have *proto*, got *opt*, (*ob1*, *ob2*, *ob3*)

Long Syntax: PPP.168 ECP proto *proto*, option is *opt*, (*ob1*, *ob2*, *ob3*).

Description: Another router sent a configuration request containing options.

PPP.169

Level: C-TRACE

Short Syntax: PPP.169 ECP rec reset-req nt *network ID*

Long Syntax: PPP.169 ECP received encryption reset-req on network *network ID*

Description: ECP received a reset request from the remote host. This is likely due to lost or corrupted packets.

PPP.170

Level: C-TRACE

Short Syntax: PPP.170 ECP rec reset-ack nt *network ID*

Long Syntax: PPP.170 ECP received encryption reset-ack on network *network ID*

Description: ECP received a reset acknowledge from the remote host.

PPP.171

Level: UI-ERROR

Short Syntax: PPP.171 ECP rcv CODE_REJ *code,* nt *network ID*.

Long Syntax: PPP.171 ECP received CODE_REJ for code *code,*, net *network ID*.

Description: ECP received a CODE_REJect for a ECP packet. Code 14 is RESET-REQ and remote hosts not supporting encryption reset may reject it. The router terminates the ECP connection and may restart on its own.

PPP.172

Level: C-INFO

Short Syntax: PPP.172 ECP dis nt *network ID*.

Long Syntax: PPP.172 ECP data encryption disabled at boot time, net *network ID*.

Description: ECP data encryption is disabled on this interface.

PPP.173

Level: C-INFO

Short Syntax: PPP.173 ccinit *typename,* will *will_neg,* mem *mem,*, nt *network ID*.

Long Syntax: PPP.173 ECP init: *typename,* will_negotiate *will_neg,* maxmem *mem,*, net *network ID*.

Description: Boot time list of ECP's available encrypters and their cost.

PPP.174

Level: C-INFO

Short Syntax: PPP.174 ECP *dir,* no buf net *network ID*.

Long Syntax: PPP.174 ECP *dir,*put no buffers available net *network ID*.

Description: ECP tried to allocate an input or output buffer and failed.

PPP.175

Level: C-TRACE

Short Syntax: PPP.175 ECP snd reset-req nt *network ID*

Long Syntax: PPP.175 ECP sent encryption reset-req on network *network ID*

Description: ECP sent a reset request to the remote host. This is due to lost or corrupted packets.

PPP.176

Level: UE-ERROR

Short Syntax: PPP.176 Bad *alg_name,* seq, exp *exp_id,* gt *got_id,*, nt *network ID*

Long Syntax: PPP.176 *alg_name,* decrypt, bad sequence id, expected *exp_id,*, got *got_id,*, on network *network ID*

Description: ADC data decrypt got bad sequence number. This is due to missing packets.

PPP.177

Level: UE-ERROR

Short Syntax: PPP.177 Malformed encrypted packet, nt *network ID*

Long Syntax: PPP.177 Decrypter dropped a malformed packet, network *network ID*

Description: An encrypted packet was received but discarded because it was malformed (bad length or similar) and could not be decoded.

PPP.178

Level: P_TRACE

Short Syntax: PPP.178 mk mrru *mrru*

Long Syntax: PPP.178 making max reconstructed receive unit with value *mrru*

Description: lcp_option built mrru.

PPP.179

Level: P_TRACE

Short Syntax: PPP.179 mk endpt discriminator

Long Syntax: PPP.179 making endpoint discriminator

Description: lcp_option built endpt discriminator.

PPP.180

Level: P_TRACE

Short Syntax: PPP.180 mk link discriminator

Long Syntax: PPP.180 making link discriminator

Description: lcp_option built link discriminator.

PPP.181

Level: CE-ERROR

Short Syntax: PPP.181 Duplicate address *address* nt *network ID*

Long Syntax: PPP.181 IPCP negotiated IP address *address* is being used by another host nt *network ID*

Description: Another host responded to an ARP for

this IP address, route will not be added.

PPP.182

Level: C-INFO

Short Syntax: PPP.182 Added route from *from* to *to* mask *mask*

Long Syntax: PPP.182 Added route from *from* to *to* mask *mask*

Description: Added new static route for dialin client or LAN

PPP.183

Level: C-INFO

Short Syntax: PPP.183 IPCP no buf net *network ID*

Long Syntax: PPP.183 IPCP no buffers available net *network ID*

Description: IPCP tried to allocate an input or output buffer and failed.

PPP.184

Level: C-INFO

Short Syntax: PPP.184 Add static rte to *address* failed nt *network ID*

Long Syntax: PPP.184 Add static route to *address* failed nt *network ID*

Description: Failed arp to check and see if this address is in use

---

PPP.185

Level: CI-ERROR

Short Syntax: PPP.185 Can't do cmprs on new intfc, tlrsiz too small, nt *network ID*

Long Syntax: PPP.185 Can't do compression on activated interface due to limited trailer size on allocated packet buffers, network *network ID*

Description: Data compression could not be enabled on an interface due to size constraints on the buffers which have already been allocated in the box. Data compression requires that the trailersize on buffers be somewhat larger than normal. This normally occurs when an ACTIVATE INTERFACE has been done on a PPP interface, but no existing circuit in the box had compression enabled on it. If the router is restarted at this point, it will allocate buffers with larger trailer sizes, and compression should be operable.

PPP.186

Level: C-INFO

Short Syntax: PPP.186 ARP ent deleted for prt add *protocol_address*

Long Syntax: PPP.186 ARP entry deleted for IP address *protocol_address*

Description: ARP entry for the dial-in IP address has now been deleted

---

PPP.187

Level: UI_ERROR

Short Syntax: PPP.187 No available Mac Addr - disabling *type*

Long Syntax: PPP.187 No available Mac Addr - disabling *type*

Description: Could not get a mac address

PPP.188

Level: C_TRACE

Short Syntax: PPP.188 Net state change, net *network ID*) is *state*.

Long Syntax: PPP.188 Net state change, net *network ID*) is *state*.

Description: A PPP net was set to a (possibly) new state.

Action: None; informational message only.

PPP.189

Level: C_TRACE

Short Syntax: PPP.189 Protocol *protocol* marked down, net *network ID*).

Long Syntax: PPP.189 Protocol *protocol* marked down, net *network ID*).

Description: A layer-3 protocol was marked down on a PPP link. Typically this occurs when a RESET PROTOCOL is performed.

Action: None; informational message only.

PPP.190

Level: C_TRACE

Short Syntax: PPP.190 Protocol *protocol action* on net *network ID*).

Long Syntax: PPP.190 Protocol *protocol action* on net *network ID*).

Description: A layer-3 protocol registered or unregistered itself as eligible to run on a PPP interface.

The "action" parameter indicates whether it registered or unregistered.

Action: None; informational message only.

PPP.191

Level: C_TRACE

Short Syntax: PPP.191 *sent_or_rcvd* LCP *lcp_packet_type*, ID= *id*, net *network ID*) LCP State = < *fsm_state*>, PktLen= *pkt_len*, LcpLen= *contents*, Contents:

Long Syntax: PPP.191 *sent_or_rcvd* LCP *lcp_packet_type*, ID= *id*, net *network ID*) LCP State = < *fsm_state*>, PktLen= *pkt_len*, LcpLen= *contents*, Contents:

Description: An LCP Configuration packet was sent or received. This refers to Config-Request, -Ack, -Nak, and -Reject packets. The "contents" field will describe the options present in the packet (or as much as can be fit in a single ELS message). The <fsm_state> shows the state of LCP at the time the action is logged. The PktLen is the length of the packet as received (not including framing/HDLC/FCS bytes) whereas LcpLen is the "stated" length of the packet included in the LCP header. These values are usually the same, but PktLen might be larger if the packet included some padding (if it is SMALLER than LcpLen then the packet is truncated and invalid).

Action: None; informational message only.

PPP.193

Level: U_INFO

Short Syntax: PPP.193 *algorithm* decomp/decrypt fld (sts= *status*), pkt lost, net *network ID*.

Long Syntax: PPP.193 *algorithm* decompression / decryption failed (status= *status*), packet lost, net *network ID*.

Description: A packet could not be decompressed or decrypted properly and was lost. The status value is the return code provided by the decompression or decryption routine.

PPP.194

Level: U_INFO

Short Syntax: PPP.194 *algorithm* encryption failed (status= *status*), net *network ID*.

Long Syntax: PPP.194 *algorithm* encryption failed (status= *status*), net *network ID*.

Description: A packet could not be encrypted properly and was not transmitted. The status value is the return code provided by the encryption routine.

PPP.195

Level: UE-ERROR

Short Syntax: PPP.195 MSCHAP usr ' *username*' bad chgpwd len *length* nt *network ID*.

Long Syntax: PPP.195 MSCHAP user ' *username*' supplied a bad change password length of *length* on net *network ID*.

Description: MSCHAP received a change password packet that was too short.

PPP.196

Level: UE_ERROR

Short Syntax: PPP.196 Auth svr does not support MSCHAP nt *net_number*.

Long Syntax: PPP.196 The authentication server does not support MSCHAP on net *net_number*.

Description: The current configuration uses a authentication server which does not support MSCHAP. All MSCHAP attempts to authenticate a user with this server will fail. Either disable MSCHAP or use an MSCHAP supported authentication server.

PPP.197

Level: UI_ERROR

Short Syntax: PPP.197 Cannot gt remote pwd for user ' *username*' nt *net_number*.

Long Syntax: PPP.197 Authentication server returned an empty password for user ' *username*' on net *net_number*.

Description: MSCHAP needs the current peer password to decrypt the new password included in the Change Password packet. MSCHAP could not obtain this password and therefore could not change the peer's Local List password.

PPP.198

Level: UI_ERROR

Short Syntax: PPP.198 Bad ptr.

Long Syntax: PPP.198 MSCHAP encountered an invalid internal value.

Description: MSCHAP encountered an invalid pointer value. This is an internal error.

PPP.199

Level: UE_ERROR

Short Syntax: PPP.199 MSCHAP user ' *username*' sent unexpected chg pwd nt *net_number*.

Long Syntax: PPP.199 MSCHAP user ' *username*' sent

an unexpected/unsupported change password pkt on net *net_number*.

Description: The peer sent an unauthorized change password packet. The packet is ignored.

PPP.200

Level: P_TRACE

Short Syntax: PPP.200 ck CBCP

Long Syntax: PPP.200 checking Callback Control Protocol (CBCP).

Description: lcp_check processed CBCP request

PPP.201

Level: P_TRACE

Short Syntax: PPP.201 Callback PPP user *username* at *dial_number* in *delay* seconds on net *network ID*.

Long Syntax: PPP.201 Callback PPP user *username* at *dial_number* in *delay* seconds on net *network ID*.

Description: A PPP user is being called back for roaming or required callback.

PPP.202

Level: P_TRACE

Short Syntax: PPP.202 Callback call start for user *username*, at *destination*, on net *network ID*.

Long Syntax: PPP.202 Callback call start for user *username*, at *destination*, on net *network ID*.

Description: The specified user is about to be called back at the number shown.

PPP.203

Level: P_TRACE

Short Syntax: PPP.203 Callback successful reconnection of user *username*, on net *network ID*.

Long Syntax: PPP.203 Callback successful reconnection of user *username*, on net *network ID*.

Description: Successful callback for listed user.

PPP.204

Level: P_TRACE

Short Syntax: PPP.204 Callback collision. User *username*, interrupt callback of *username*, on net *network ID*.

Long Syntax: PPP.204 Callback collision. User *username*, interrupt callback of *username*, on net *network ID*.

Description: While attempting to callback the second user, the first user called. The first user is disconnected and callback of user two continues.

PPP.205

Level: UE_ERROR

Short Syntax: PPP.205 MSCHAP disabled while MPPE is enabled nt *net_number*.

Long Syntax: PPP.205 MSCHAP was disabled while MPPE is still enabled on net *net_number*.

Description: The current configuration has MPPE encryption enabled. This requires MSCHAP to be enabled also. An attempt to disable MSCHAP without disabling MPPE (encryption) may cause all PPP links to fail.

PPP.206

Level: P_TRACE

Short Syntax: PPP.206 MPPE is active for nt *net_number*..

Long Syntax: PPP.206 Net *net_number* has negotiated MPPE.

Description: Indicates whether MPPE is active for the particular net.

PPP.207

Level: P_TRACE

Short Syntax: PPP.207 Link dropped for net *net_number* because MPPE was enabled and could not negotiate..

Long Syntax: PPP.207 Link dropped for net *net_number* because MPPE was enabled and could not negotiate..

Description: If MPPE does not negotiate and the net is enabled for mandatory encryption, then the link must drop.

PPP.208

Level: P-TRACE

Short Syntax: PPP.208 Getting ready to send packet through compression and encryption routines.

Long Syntax: PPP.208 Getting ready to send packet through compression and encryption routines.

Description: Almost a complete packet. Does not have protocol or HDLC header. Mainly for Packet tracing.

PPP.209

Level: P-TRACE

Short Syntax: PPP.209 Code reject. PPP packet not supported.

Long Syntax: PPP.209 Code reject. PPP packet not supported.

Description: This can be normal during PPP negotiations. A Code reject reject occurs when we or our peer does not support a particular packet code. For example, a peer which does not support LCP time-remaining would code reject this packet. For packet tracing.

PPP.210

Level: P-TRACE

Short Syntax: PPP.210 Authentication packet received

Long Syntax: PPP.210 Authentication packet received

Description: Received an authentication packet. Mainly for Packet tracing.

PPP.211

Level: P-TRACE

Short Syntax: PPP.211 Termination packet sent/received

Long Syntax: PPP.211 Termination packet sent/received

Description: A termination request or termination ack was sent/received. Mainly for Packet tracing.

PPP.212

Level: UI_ERROR

Short Syntax: PPP.212 incompatible LCP negotiations exceeded net *network ID*

Long Syntax: PPP.212 incompatible LCP negotiations exceeded net *network ID*

Description: The number of incompatible LCP negotiations was exceeded and the link will be terminated. Check LCP exchange in talk 2 to determine if a parameter config change can alleviate the problem.

Panic pppimem

Short Syntax: PPP interface initialization failed, no memory.

Description: The PPP interface failed to allocate sufficient memory to complete initialization.

Action: Contact customer service.

Panic pppiprt

Short Syntax: PPP: unsupported protocol during initialization

Description: The PPP network handler detected an unsupported protocol during initialization.

Action: Contact customer service.

Panic pppidev

Short Syntax: PPP: wrong device type

Description: The PPP network handler detected PPP configured on a device other than I_LOUIE or I_ATC2 during init.

Action: Contact customer service.

Chapter 91. Policy Database and Search Engine (PLCY)

This chapter describes Policy Database and Search Engine (PLCY) messages. For information on message content and how to use the message, refer to the Introduction.

PLCY.001

Level: UE-ERROR

Short Syntax: PLCY.001 Generic Error: *errorString*

Long Syntax: PLCY.001 Generic Error: *errorString*

Description: PLCY: An error during the policy search. The error string will signify the type of error that occurred.

Action: Contact IBM Service and report the error message

PLCY.002

Level: U-INFO

Short Syntax: PLCY.002 Begin building policy database, default rule configuration is *defaultConfig*

Long Syntax: PLCY.002 Begin building policy database, default rule configuration is *defaultConfig*

Description: PLCY: Policy search engine has been started, step 1 is to create the rules for the default configuration

PLCY.003

Level: U-INFO

Short Syntax: PLCY.003 *number* policies read from local configuration

Long Syntax: PLCY.003 *number* policies read from local configuration

Description: PLCY: X number of policies read locally from SRAM

PLCY.004

Level: U-INFO

Short Syntax: PLCY.004 Next refresh of policy DB in *hour* hour(s), *min* min(s) *second* second(s)

Long Syntax: PLCY.004 Next refresh of policy DB in *hour* hour(s), *min* min(s) *second* second(s)

Description: PLCY: A database refresh will automatically occur in the when the shown time has elapsed.

PLCY.005

Level: U-INFO

Short Syntax: PLCY.005 Automatic refresh of policy database is disabled

Long Syntax: PLCY.005 Automatic refresh of policy database is disabled

Description: PLCY: Policy Refresh has been disabled by the user

PLCY.006

Level: U-INFO

Short Syntax: PLCY.006 Marked list of valid policies, next check in *seconds* seconds

Long Syntax: PLCY.006 Marked list of valid policies, next check in *seconds* seconds

Description: PLCY: Performed check of valid policies and marked policy either valid or invalid based on the current time and the policy validity period. Will perform check again when the next policy is scheduled to become invalid or valid or an hour later, whichever time is smaller.

PLCY.007

Level: U-INFO

Short Syntax: PLCY.007 LDAP Policy Search FSM State *state*/Event *event*

Long Syntax: PLCY.007 LDAP Policy Search FSM State *state*/Event *event*

Description: PLCY: Information about events and the corresponding state changes that drive the policy state machine

PLCY.008

Level: UE-ERROR

Short Syntax: PLCY.008 DeviceProfile *deviceProfileDN*, *objectClassType* in LDAP Directory does not have a DeviceRulesReference

Long Syntax: PLCY.008 DeviceProfile *deviceProfileDN*, *objectClassType* in LDAP Directory does not have a DeviceRulesReference

Description: PLCY: The deviceprofile object in the

directory does not have a attribute for DeviceRulesReference. This attribute must be present as it specifies the list of rules that should be fetched for this device.

Action: Update the deviceprofile object for this device with the reference to the device rules object.

PLCY.009

Level: UE-ERROR

Short Syntax: PLCY.009 Error occured during LDAP processing, error msg: *errorMsg*

Long Syntax: PLCY.009 Error occured during LDAP processing, error msg: *errorMsg*

Description: PLCY: An error occurred connect to, binding to, searching from or retrieving results from the LDAP directory. Please review the specific error message for more information.

PLCY.010

Level: U-INFO

Short Syntax: PLCY.010 Completed bind to LDAP server successfully

Long Syntax: PLCY.010 Completed bind to LDAP server successfully

Description: PLCY: Successfully completed the bind to the ldap server, can now begin to search directory for policy information

---

PLCY.011

Level: UE-ERROR

Short Syntax: PLCY.011 Cannot access LDAP Server, route not available to IP Address *ipAddr*

Long Syntax: PLCY.011 Cannot access LDAP Server, route not available to IP Address *ipAddr*

Description: PLCY: Cannot perform a TCP connect for LDAP Connection until a route to destination address for LDAP Server is available. Will continue to try periodically. Note this is normal when a net is first trying to pass selftest or when the routing protocols have not yet come up. Also try to add a static route to the ldap server in the IP Configuration

---

PLCY.012

Level: U-INFO

Short Syntax: PLCY.012 Attempting connection to LDAP server at IP Address *ipAddr*, port number *portnum*

Long Syntax: PLCY.012 Attempting connection to LDAP server at IP Address *ipAddr*, port number *portnum*

Description: PLCY: Attempting to open a TCP connection to the LDAP Server on the specified port number

---

PLCY.013

Level: UE-ERROR

Short Syntax: PLCY.013 Error searching for policy info from LDAP, performing default action

Long Syntax: PLCY.013 Error searching for policy info from LDAP, performing default action

Description: PLCY: Some error occurred searching the directory for policy information. Since an error occuring, the policy search algorithm will perform the configured error handling procedure

---

PLCY.014

Level: UE-ERROR

Short Syntax: PLCY.014 Referring object not found in the directory, Referring DN *errorMsg*, Error msg:

Long Syntax: PLCY.014 Referring object not found in the directory, Referring DN *errorMsg*, Error msg:

Description: PLCY: A referring object specified by a reference attribute was not found in the directory. Check to make sure the reference is type correctly and is indeed populated in the directory

---

PLCY.015

Level: U-INFO

Short Syntax: PLCY.015 Searching LDAP for Object with dn: *refDn*

Long Syntax: PLCY.015 Searching LDAP for Object with dn: *refDn*

Description: PLCY: Informational message about the next object being searched for from LDAP Directory

---

PLCY.016

Level: UE-ERROR

Short Syntax: PLCY.016 DeviceRules object (DN = *deviceRuleDN*) not found in the directory, Error msg: *errorMsg*

Long Syntax: PLCY.016 DeviceRules object (DN = *deviceRuleDN*) not found in the directory, Error msg: *errorMsg*

Description: PLCY: The device rules object was not found in the directory. Please make sure this is configured in feature policy under talk 6 and that the object does indeed exist in the directory

---

PLCY.017

Level: UE-ERROR

Short Syntax: PLCY.017 No PolicyRuleReference attribute found in DeviceRule object *deviceRuleDN*

Long Syntax: PLCY.017 No PolicyRuleReference attribute found in DeviceRule object *deviceRuleDN*

Description: PLCY: The device rules object did not specify any policy rule references. There must be PolicyRuleReference attributes defined in the DeviceRules. This multi-valued attribute specifies which rules the device should fetch and load into the policy database for the device.

PLCY.018

Level: UE-ERROR

Short Syntax: PLCY.018 *objectType* (*objectName*) retrieved from the LDAP server was in error

Long Syntax: PLCY.018 *objectType* (*objectName*) retrieved from the LDAP server was in error

Description: PLCY: A policy object in the LDAP Directory had an error associated with it. Please check the object with the displayed name and make sure that the information it contains is correct

PLCY.019

Level: UE-ERROR

Short Syntax: PLCY.019 Value(*value*) out of range for LDAP Attribute *attrString*, valid range is *lowVal* to *highVal*

Long Syntax: PLCY.019 Value(*value*) out of range for LDAP Attribute *attrString*, valid range is *lowVal* to *highVal*

Description: PLCY: An attribute of an object being fetched from the directory was out of range. The valid range should have been displayed in the event log message. Please check this attribute and modify its value to be in this range.

PLCY.020

Level: UE-ERROR

Short Syntax: PLCY.020 Error occured while parsing attribute *attrName*, objectclass *objName*, value was *value*

Long Syntax: PLCY.020 Error occured while parsing attribute *attrName*, objectclass *objName*, value was *value*

Description: PLCY: An attribute of an object being fetched from the directory was in error. Either it was out of range, had an invalid value or some other error. Please check this attribute and modify its value to be correct.

PLCY.021

Level: U-INFO

Short Syntax: PLCY.021 Ignoring Attribute *attrName* in Class *objName* since it is not a recognized attribute

Long Syntax: PLCY.021 Ignoring Attribute *attrName* in Class *objName* since it is not a recognized attribute

Description: PLCY: An object was retrieved from the directory with an unrecognized attribute. This is not necessary an error since not all attributes are supported or recognized in a given object class and class definitions may have changed since this release of code. However, please check this attribute definition and make sure that it is indeed unnecessary for this device's operation. If you feel it is necessary contact IBM Service with the information and have someone look into the problem.

PLCY.022

Level: U-INFO

Short Syntax: PLCY.022 Found object (DN: *dn*), parse using class def *objName*

Long Syntax: PLCY.022 Found object (DN: *dn*), parse using class def *objName*

Description: PLCY: An object was retrieved from the directory and the search algorithm is about to begin parsing the attributes for this class.

PLCY.023

Level: U-INFO

Short Syntax: PLCY.023 Policy DB query (*queryType,*), src: *srcIPAddr,/ srcPortNum,*,dst: *dstIPAddr,/ dstPortNum,*,prot: *protocol,*,DS: *DiffServByte*

Long Syntax: PLCY.023 Policy DB query (*queryType,*), src: *srcIPAddr,/ srcPortNum,*,dst: *dstIPAddr,/ dstPortNum,*,prot: *protocol,*,DS: *DiffServByte*

Description: PLCY: Received an Policy query of the indicated type, with packet information specified via the parameters

PLCY.024

Level: U-INFO

Short Syntax: PLCY.024 *qType* has been returned

Long Syntax: PLCY.024 *qType* has been returned

Description: PLCY: The Decision Object contain an action of the idicated type.

PLCY.025

Level: U-INFO

Short Syntax: PLCY.025 *qType* Rule matched: *ruleName*

Long Syntax: PLCY.025 *qType* Rule matched: *ruleName*

Description: PLCY: A specific rule has been matched.

PLCY.026

Level: U-INFO

Short Syntax: PLCY.026 Completed building policy DB, *numRules* rules loaded

Long Syntax: PLCY.026 Completed building policy DB, *numRules* rules loaded

Description: PLCY: Finished building the policy database with numRules

PLCY.027

Level: UE-ERROR

Short Syntax: PLCY.027 Not enough memory to build tree for policy database

Long Syntax: PLCY.027 Not enough memory to build tree for policy database

Description: PLCY: If this error message is encountered then the user must upgrade to more memory or substantially reduce the number of policies this device is attempting to enforce.

PLCY.028

Level: UE-ERROR

Short Syntax: PLCY.028 Dropping pkt, did not arrive in a secure ipsec tunnel, rule matched *ruleName*

Long Syntax: PLCY.028 Dropping pkt, did not arrive in a secure ipsec tunnel, rule matched *ruleName*

Description: PLCY: A packet should have arrived in a secure tunnel but the policy has detected the packet came in the clear. The packet will be dropped as a result. This could result from someone attempting to get into the protected network.

PLCY.029

Level: U-INFO

Short Syntax: PLCY.029 Phase 1 *QueryType,* Query returning no match for Phase 1 Rule *RuleName,* (*ConfiguredPolicyRole*)

Long Syntax: PLCY.029 Phase 1 *QueryType,* Query returning no match for Phase 1 Rule *RuleName,* (*ConfiguredPolicyRole*)

Description: PLCY: A match was found for the phase 1 rule but we can not return the match because the configured phase 1 action only supported being an initiator or responder and this query is the opposite.

PLCY.030

Level: U-INFO

Short Syntax: PLCY.030 Proxy does not match selectors. Will not create neg item.

Long Syntax: PLCY.030 Proxy does not match selectors. Will not create neg item.

Description: PLCY: The first packet of a flow matches a profile in rule but does not match proxy values in IPSEC action. The CPE acts as if there was no match to the any IKE rule.

PLCY.031

Level: U-INFO

Short Syntax: PLCY.031 Created IKE Phase2 negotiated item.

Long Syntax: PLCY.031 Created IKE Phase2 negotiated item.

Description: PLCY: The first packet of a flow matches a profile in rule and the correspondig proxy values in IPSEC action. A new IKE Phase negotiated item is created to hold tunnel ID info.

PLCY.032

Level: U-INFO

Short Syntax: PLCY.032 Cannot Access LDAP Server, set the dflt tunnel endpoint info (set default)

Long Syntax: PLCY.032 Cannot Access LDAP Server, set the dflt tunnel endpoint info (set default)

Description: PLCY: The user must configure the tunnel endpoint (local and remote) information using the set default command in talk 6, feature policy. This is only necessary when the default rule behavior is drop all but LDAP or secure LDAP.

PLCY.033

Level: U-INFO

Short Syntax: PLCY.033 Phase 1 query, checked rule *ruleName,* for remote ID *remoteId,, matchFound*

Long Syntax: PLCY.033 Phase 1 query, checked rule *ruleName,* for remote ID *remoteId,, matchFound*

Description: PLCY: Informational message about the remote Id check during either the AGGRESSIVE MODE responder, MSG5 Responder, or MSG6 initiator case during the phase 1 ISAKMP negotiations. The match may fail for the rule being checked. If no match is found in any rule in the policy database, then this user is indeed not allowed access to the network or the user

should be allowed access to the network and must be added to the user group for a policy.

PLCY.034

Level: U-INFO

Short Syntax: PLCY.034 Created IKE Phase1 negotiated item, cpeP1Handle 0x *cpeP1Handle*

Long Syntax: PLCY.034 Created IKE Phase1 negotiated item, cpeP1Handle 0x *cpeP1Handle*

Description: PLCY: A new IKE Phase negotiated item is created to hold Phase 1 negotiated information.

PLCY.035

Level: UE-ERROR

Short Syntax: PLCY.035 Phase2 IDCi/IDCr Proxy match failed, error: *mismatchReason*

Long Syntax: PLCY.035 Phase2 IDCi/IDCr Proxy match failed, error: *mismatchReason*

Description: PLCY: The IDci and IDcr match that is performed as a ISAKMP Phase2 responder failed. The information in this message will tell the user what failed. If there is no match then make sure the proxy configuration on the remote side matches the proxy configuration on the local side.

PLCY.036

Level: U-INFO

Short Syntax: PLCY.036 tPeerIDci= *remoteIDci*,tRuleIDci= *ruleIDci*,tPeerIDcr= *remoteIDcr*,tRuleIDcr= *ruleIDCr*

Long Syntax: PLCY.036 tPeerIDci= *remoteIDci*,tRuleIDci= *ruleIDci*,tPeerIDcr= *remoteIDcr*,tRuleIDcr= *ruleIDCr*

Description: PLCY: Displayes the remote ISAKMP peer's value for Proxy local and remote informtion. Also displays idci and idcr configured in the policy being checked against. If these values dont match, an additional ELS message will be displayed. If they do not match, please ensure that both policies for each end of the tunnel are configured to match each other. A typical mistake is to configure different protocols or ports to put into the tunnel, they should match.

PLCY.037

Level: UE-ERROR

Short Syntax: PLCY.037 Encryption Algorithm *encrAlgorithm* for *objectName* not supported in this image

Long Syntax: PLCY.037 Encryption Algorithm *encrAlgorithm* for *objectName* not supported in this image

Description: PLCY: The message will be displayed whenever the policy database is being built and a configured encryption algorithm is detected that is not supported in the load. This can occur when you configure the box with a policy that contains an encryption algorithm that is supported in the current load image but then at a later point load the box with an image that does not support the configured encryption algorithm.

PLCY.038

Level: UE-ERROR

Short Syntax: PLCY.038 Object *objectName*, IP Address *ipaddr* is not a configured IP Address

Long Syntax: PLCY.038 Object *objectName*, IP Address *ipaddr* is not a configured IP Address

Description: PLCY: This message will be displayed whenever the policy engine detects a configuration that includes an IP Address that is not a valid IP Address on the box. This includes the tunnelStart in the default policy, in the IPSEC Action, and the interface pairs that are part of the profile.

PLCY.039

Level: U-INFO

Short Syntax: PLCY.039 Maximum number of IPSEC phase 2 tunnels reached, limit = *limit*

Long Syntax: PLCY.039 Maximum number of IPSEC phase 2 tunnels reached, limit = *limit*

Description: PLCY: This message will be displayed whenever the policy engine has a request to bringup a new tunnel and the number of active IKE negotiated IPSEC tunnels is already at the maximum

---

PLCY.040

Level: U-INFO

Short Syntax: PLCY.040 Request to *addOrDelete,* an IKE phase *phase1or2,* tunnel, totals active *totalActiveForPhase1or2*

Long Syntax: PLCY.040 Request to *addOrDelete,* an IKE phase *phase1or2,* tunnel, totals active *totalActiveForPhase1or2*

Description: PLCY: This message will be displayed whenever the policy engine has a request to add or delete a phase 1 or phase 2 ike tunnel. The number of active tunnels of that type is also displayed.

Chapter 92. Presence Manager (PM)

This chapter describes Presence Manager (PM) messages. For information on message content and how to use the message, refer to the Introduction.

PM.001

Level: UI-ERROR

Short Syntax: PM.001 Fan *fan* failed.

Long Syntax: PM.001 Cooling fan *fan* has failed.

Description: A cooling fan has stopped spinning at the minimum RPM required to provide adequate cooling.

PM.002

Level: U-INFO

Short Syntax: PM.002 Fan *fan* up to speed.

Long Syntax: PM.002 Cooling fan *fan* is up to speed.

Description: A cooling fan which had previously failed is now spinning at the minimum RPM required to provide adequate cooling.

PM.003

Level: U-INFO

Short Syntax: PM.003 Pwr Supp *power_supply* OFF.

Long Syntax: PM.003 Power Supply *power_supply* is OFF or has failed.

Description: A power supply has been powered-off or has failed.

PM.004

Level: U-INFO

Short Syntax: PM.004 Pwr Supp *power_supply* ON.

Long Syntax: PM.004 Power Supply *power_supply* is ON.

Description: A power supply has been powered-on.

PM.005

Level: U-INFO

Short Syntax: PM.005 Thermal *thermal* Overtemp.

Long Syntax: PM.005 Thermal Sensor *thermal* is over-temp.

Description: A thermal sensor reading has exceeded the specified threshold.

PM.006

Level: U-INFO

Short Syntax: PM.006 Thermal *thermal* below thresh.

Long Syntax: PM.006 Thermal Sensor *thermal* is below warning level.

Description: A thermal sensor reading which had previously exceeded the specified threshold is now below the specified threshold.

PM.007

Level: U-INFO

Short Syntax: PM.007 LIC2 *at0 at1 - lic_name* detected in slot *slot*.

Long Syntax: PM.007 LIC2 *at0 at1 - lic_name* detected in slot *slot*.

Description: A LIC of the type indicated has been detected in the slot indicated.

PM.008

Level: U-INFO

Short Syntax: PM.008 LIC2 *at0 at1 - lic_name* extracted from slot *slot*.

Long Syntax: PM.008 LIC2 *at0 at1 - lic_name* extracted from slot *slot*.

Description: A LIC of the type indicated has been extracted from the slot indicated.

PM.009

Level: UE-ERROR

Short Syntax: PM.009 Mechanical Insertion Error, slot *slot*.

Long Syntax: PM.009 Mechanical Insertion Error in slot *slot*.

Description: A mechanical insertion error has occurred indicating that the LIC type could not be detected. Re-inserting the LIC is required.

PM.010

Level: UE-ERROR

Short Syntax: PM.010 Unknown LIC Type in slot *slot*.

Long Syntax: PM.010 Unknown LIC Type detected in slot *slot*.

Description: The LIC type plugged into the slot does not match any known LIC type.

PM.011

Level: UE-ERROR

Short Syntax: PM.011 LIC2 *at0 at1 - lic_name* is not valid in slot *slot*.

Long Syntax: PM.011 LIC2 *at0 at1 - lic_name* is not valid in slot *slot*.

Description: The type of the LIC plugged into the slot is not compatible with a LIC present in an adjacent slot. As a result, the LIC will not be enabled and the ″Wrong Slot″ indicator will be on.

PM.012

Level: UI-ERROR

Short Syntax: PM.012 LIC2 *at0 at1 - lic_name* in slot *slot* is defective.

Long Syntax: PM.012 LIC2 *at0 at1 - lic_name* in slot *slot* is defective.

Description: The LIC type plugged into the slot is defective.

Chapter 93. Protocol Independent Multicast (PIM)

This chapter describes Protocol Independent Multicast (PIM) messages. For information on message content and how to use the message, refer to the Introduction.

PIM.001

Level: C-TRACE

Short Syntax: PIM.001 Add phyint *IP_interface*

Long Syntax: PIM.001 Add physical interface *IP_interface*

Description: PIM has been enabled on the specified physical interface.

PIM.002

Level: C-TRACE

Short Syntax: PIM.002 Add tunnel *tunnel_source*-> *tunnel_destination*

Long Syntax: PIM.002 Add tunnel *tunnel_source*-> *tunnel_destination*

Description: A PIM tunnel has been configured between the given source and destination addresses.

PIM.003

Level: U-TRACE

Short Syntax: PIM.003 Nbr *IP_neighbor* removed, ifc down

Long Syntax: PIM.003 Neighbor *IP_neighbor* removed due to interface going down

Description: The PIM virtual interface is going down and this neighbor has been removed from the neighbor list.

PIM.004

Level: UE-ERROR

Short Syntax: PIM.004 bd hdr cks 0x *checksum* (exp 0x *expected_checksum*) *source_ip_address* -> *destination_ip_address*

Long Syntax: PIM.004 Bad header checksum 0x *checksum* (expected 0x *expected_checksum*) in packet from *source_ip_address* for *destination_ip_address*

Description: This message is generated when a PIM control message has an invalid checksum. The received checksum, together with the correct checksum, are displayed.

Cause: Most likely, this is a damaged packet. It may be that another node is building an incorrect PIM control message.

Action: If the problem persists, examine a line trace to determine where the packet is being damaged.

PIM.005

Level: U-TRACE

Short Syntax: PIM.005 Rcvd PIM unk ver= *PIM_version IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: PIM.005 Received PIM Unknown Version= *PIM_version*, *IP_source* -> *IP_destination* net *network ID*

Description: A PIM Control Message was received with an unknown version number. Packet is discarded.

PIM.006

Level: P-TRACE

Short Syntax: PIM.006 Rcvd Hello Msg *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: PIM.006 Received PIM Hello Message, *IP_source* -> *IP_destination* net *network ID*

Description: A PIM Hello Message has been received.

PIM.007

Level: P-TRACE

Short Syntax: PIM.007 Rcvd Jn/Prn Msg *IP_source* => *PIM_upstream*, nt *network ID*

Long Syntax: PIM.007 Received PIM Join/Prune Message, *IP_source* upstream *PIM_upstream* net *network ID*

Description: A PIM Join/Prune Message has been received. The displayed address is not the IP packet destination address, but the upstream router address contained in the PIM protocol packet.

PIM.008

Level: P-TRACE

Short Syntax: PIM.008 Rcvd Assert Msg, grp *IP_group*, src *IP_source*, nt *network ID*

Long Syntax: PIM.008 Received PIM Assert Message,

Group *IP_group*, Source *IP_source*, net *network ID*

Description: A PIM Assert Message has been received.

PIM.009

Level: P-TRACE

Short Syntax: PIM.009 Rcvd Graft Msg *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: PIM.009 Received PIM Graft Message, *IP_source* -> *IP_destination* net *network ID*

Description: A PIM Graft Message has been received.

PIM.010

Level: P-TRACE

Short Syntax: PIM.010 Rcvd GraftAck Msg *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: PIM.010 Received PIM Graft Ack Message, *IP_source* -> *IP_destination* net *network ID*

Description: A PIM Graft Acknowledgement Message has been received.

PIM.011

Level: UE-ERROR

Short Syntax: PIM.011 Rcvd unk msg *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: PIM.011 Received PIM Unknown Message, *IP_source* -> *IP_destination* net *network ID*

Description: A PIM Control Message was received which has an unknown message type. The message is discarded.

PIM.012

Level: P-TRACE

Short Syntax: PIM.012 Send Hello Msg *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: PIM.012 Sending PIM Hello Message, *IP_source* -> *IP_destination* net *network ID*

Description: A PIM Hello Message has been created and scheduled for transmission.

PIM.013

Level: P-TRACE

Short Syntax: PIM.013 Send Jn/Prn Msg *IP_source* => *PIM_upstream*, nt *network ID*

Long Syntax: PIM.013 Sending PIM Join/Prune Message, *IP_source* upstream *PIM_upstream* net *network ID*

Description: A PIM Join/Prune Message has been created and scheduled for transmission. The displayed address is not the IP packet destination address (which is the all PIM router address, but the upstream router address contained in the PIM protocol packet.

PIM.014

Level: P-TRACE

Short Syntax: PIM.014 Send Assert Msg, grp *IP_group*, src *IP_source*, nt *network ID*

Long Syntax: PIM.014 Sending PIM Assert Message, Group *IP_group*, Source *IP_source* net *network ID*

Description: A PIM Assert Message has been created and scheduled for transmission. The Assert occurs when a multicast data packet was received on an output interface for a multicast entry in the forwarding tables or having received an assert from another router which should not be forwarding the multicast data in question. This action signifies that duplicate multicast data is being sent due to routing loops. The assert message resolves which router should be the forwarding router and ensures this condition does not persist.

PIM.015

Level: P-TRACE

Short Syntax: PIM.015 Send Graft Msg *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: PIM.015 Sending PIM Graft Message, *IP_source* -> *IP_destination* net *network ID*

Description: A PIM Graft Message has been created and scheduled for transmission.

PIM.016

Level: P-TRACE

Short Syntax: PIM.016 Send GraftAck Msg *IP_source* -> *IP_destination*, nt *network ID*

Long Syntax: PIM.016 Sending PIM Graft Ack Message, *IP_source* -> *IP_destination* net *network ID*

Description: A PIM Graft Acknowledgement Message has been created and scheduled for transmission.

PIM.017

Level: UI-ERROR

Short Syntax: PIM.017 Could not obtain iorb *IP_source* -> *IP_destination*

Long Syntax: PIM.017 Unable to obtain an iob for send, *IP_source* -> *IP_destination*

Description: No input/output request block (iorb) was returned by the system which was required by PIM to build and send a PIM control message.

Cause: The system has run out of resources. This usually occurs due to lack of system buffer memory.

Action: If not a problem due to an errant application, install more memory and assign more system buffers.

PIM.018

Level: UI-ERROR

Short Syntax: PIM.018 Failed iorb send rqst *IP_source -> IP_destination*

Long Syntax: PIM.018 IORB send request was unsuccessful, *IP_source -> IP_destination*

Description: A request to send an iorb containing a PIM control message returned a failure and the message was discarded.

Cause: The system refused to send an input/output request block. This may be due to an overburdened router or a errant application.

Action: Upgrade to a higer performance router.

PIM.019

Level: U-TRACE

Short Syntax: PIM.019 DR *PIM_old_DR* replaced by *PIM_new_DR* for nt *network ID*

Long Syntax: PIM.019 Designated Router *PIM_old_DR* is replaced by *PIM_new_DR* net *network ID*

Description: Designated Router timed out and is replaced by the new Designated Router.

PIM.020

Level: C-TRACE

Short Syntax: PIM.020 Nbr *PIM_neighbor* del for int *PIM_interface*

Long Syntax: PIM.020 Neighbor *PIM_neighbor* Deleted from Interface *PIM_interface*

Description: The Neighbor timed out since it did not send Hello message on time and will be removed.

PIM.021

Level: UI-ERROR

Short Syntax: PIM.021 No nbr blocks available for *PIM_neighbor*, int *PIM_interface*

Long Syntax: PIM.021 No neighbor control blocks are available for *PIM_neighbor*, Interface *PIM_interface*

Description: The neighbor array is full and no neighbor control blocks are available for the newly discovered neighbor on the interface.

Cause: The number of PIM neighbors has exceeded the limit supported by the PIM implementation. This may effect the designated router selection process.

Action: Reduce the number of routers running PIM on the same subnet.

PIM.022

Level: C-TRACE

Short Syntax: PIM.022 Nbr *PIM_neighbor* added to int *PIM_interface*

Long Syntax: PIM.022 Neighbor *PIM_neighbor* added for Interface *PIM_interface*

Description: A neighbor was added to interface due to receiving a Hello message.

PIM.023

Level: C-TRACE

Short Syntax: PIM.023 J/P Msg parsed, vif *PIM_interface* gcnt *Group_count*, jcnt *Join_count*, pcnt *Prune_count*

Long Syntax: PIM.023 Join/Prune message parsed, vif = *PIM_interface*, groups = *Group_count*, joins = *Join_count*, prunes = *Prune_count*

Description: A PIM Join/Prune formatted message has been received and parsed. The counts give the total number of groups, joins, and prunes that were parsed. The PIM virtual interface is the interface the packet was received on.

PIM.024

Level: UI-ERROR

Short Syntax: PIM.024 No mem for msg q

Long Syntax: PIM.024 Could not allocate memory for building a message queue element.

Description: A memory allocation failure occurred when creating a message queue element for the PIM message queues.

Cause: The system has run low on resources and is out of system memory.

Action: Install more memory or upgrade router.

PIM.025

Level: U-TRACE

Short Syntax: PIM.025 msg type unk, msg not queued

Long Syntax: PIM.025 Could not queue message element due to unknown message type.

Description: The message scheduler could not place the message element on the appropriate message queue due to an unknown message type passed by the caller. The message element is freed back to the memory pool and no other action taken.

PIM.026

Level: UI-ERROR

Short Syntax: PIM.026 No mem for state, src = *IP_source*, grp = *IP_group*, vif = *PIM_interface*

Long Syntax: PIM.026 Could not get memory for state element, source= *IP_source*, group= *IP_group*, interface = *PIM_interface*

Description: A memory allocation failure occurred when creating a state element for the state database for the reported source group pair on the PIM interface shown.

Cause: The system has run low on resources and is out of system memory.

Action: Install more memory or upgrade router.

PIM.027

Level: UI-ERROR

Short Syntax: PIM.027 No mem for sgnode, src = *IP_source*, grp = *IP_group*, vif = *PIM_interface*

Long Syntax: PIM.027 No memory for source-group node, source = *IP_source*, group = *IP_group*, interface = *PIM_interface*

Description: A memory allocation failure occurred when creating a source-group node element for the state database for the reported source group pair on the PIM interface shown.

Cause: The system has run low on resources and is out of system memory.

Action: Install more memory or upgrade router.

PIM.028

Level: C-TRACE

Short Syntax: PIM.028 New PRUNE state src *IP_source* grp *IP_group* vif *PIM_interface*

Long Syntax: PIM.028 A new PRUNE state has been created, source = *IP_source*, group = *IP_group*, vif = *PIM_interface*

Description: A new PIM PRUNE state has been created and placed in the state database.

PIM.029

Level: U-TRACE

Short Syntax: PIM.029 Del not required src *IP_source* grp *IP_group* vif *PIM_interface*

Long Syntax: PIM.029 Delete request for state not in database, source = *IP_source*, group = *IP_group*, vif = *PIM_interface*

Description: A request was processed to delete a state that does not exist in the PIM state database. No action was taken and control returned immediately to the caller.

PIM.030

Level: C-TRACE

Short Syntax: PIM.030 PRUNE state removed src *IP_source* grp *IP_group* vif *PIM_interface*

Long Syntax: PIM.030 A PRUNE was deleted from the state database, source = *IP_source*, group = *IP_group*, vif = *PIM_interface*

Description: A PRUNE state has been successfully removed from the state database and removed from all the appropriate program queues. This PRUNE state now no longer exists.

PIM.031

Level: UI-ERROR

Short Syntax: PIM.031 srte has rte type that is UNKNOWN by PIM, src net = *IP_source* type = *route_type*

Long Syntax: PIM.031 Unicast entry has route type UNKNOWN by PIM, source net = *IP_source*, route type = *route_type*

Description: PIM is attempting to determine local unicast routing metric preference for a particular multicast forwarding entry which is used in the PIM assert process. The route type of the unicast routing entry is not of any type known by PIM and the routing metric preference has been the 0x7FFFFFFF and the routing metric set to 0xFFFFFFFF, which are used for unknown routes. The result will usually lead to this router losing the assert process and pruning its oif from the multicast forwarding cache entry.

Cause: This is usually caused by running unicast routing protocols or other functions that update the unicast forwarding table that marks the unicast entry of a route type that PIM is not aware of. This should only occur if new protocols were added to the router and PIM implementation was not updated to support them. PIM may not be able to support the new functions or a newer version of PIM is required.

Action: Call customer support and inform them of problem. If all the unicast routing protocols and forwarding table modification mechanisms are supported by PIM, a new version is necessary.

PIM.032

Level: P-TRACE

Short Syntax: PIM.032 Rcv Pim Control Msg for net *network ID*

Long Syntax: PIM.032 Received Pim Ctl Message for

net *network ID*, which is not ready

Description: Received a PIM Control Message for an interface which is not ready to receive messages. The control message is discarded.

PIM.033

Level: C-TRACE

Short Syntax: PIM.033 PIM states have been cleared

Long Syntax: PIM.033 The PIM state database has been cleared

Description: The PIM state database has thrown away all of its states. This is most commonly performed when the multicast forwarding cache has been thrown away due a unicast routing update.

PIM.034

Level: U-TRACE

Short Syntax: PIM.034 Group address not multicast, addr = *IP_group*

Long Syntax: PIM.034 An invalid group address was encountered, address = *IP_group*

Description: During parsing of a PIM control message, a group address was encountered that did not qualify as a valid multicast address. The parser did not continue processing this address and ignored it, but continued further parsing of the packet.

PIM.035

Level: U-TRACE

Short Syntax: PIM.035 jp rcv pkt len err, len *error_length* vif *PIM_interface*

Long Syntax: PIM.035 join/prune parser error due to bad PIM packet counts, length = *error_length* vif = *PIM_interface*

Description: While parsing a received PIM Join/Prune message, the parser discovered a length error. This occurs when the group, join, and prune count fields in the PIM packet itself is erroneous, reporting an incorrect number of group and source addresses contained in the packet.

PIM.036

Level: C-TRACE

Short Syntax: PIM.036 New JOIN state src *IP_source* grp *IP_group* vif *PIM_Interface*

Long Syntax: PIM.036 A new JOIN state has been created, source = *IP_source*, group = *IP_group*, vif = *PIM_Interface*

Description: A new PIM PRUNE state has been created and placed in the state database.

PIM.037

Level: C-TRACE

Short Syntax: PIM.037 JOIN state removed src *IP_source* grp *IP_group* vif *PIM_Interface*

Long Syntax: PIM.037 A JOIN was deleted from the state database, source = *IP_source*, group = *IP_group*, vif = *PIM_Interface*

Description: A JOIN state has been successfully removed from the state database and removed from all the appropriate program queues. This JOIN state now no longer exists.

PIM.038

Level: U-TRACE

Short Syntax: PIM.038 No PIM vif, *IP_source* -> *IP_destination*, net *network ID*

Long Syntax: PIM.038 Could not find a PIM virtual interface, *IP_source* -> *IP_destination* net *network ID*

Description: A PIM control message has been received that could not be mapped to a PIM virtual interface. The packet is discarded.

PIM.039

Level: UI-ERROR

Short Syntax: PIM.039 No mem for rpf q

Long Syntax: PIM.039 Could not allocate memory for building an rpf neighbor block.

Description: A memory allocation failure occurred when creating a reverse path forwarding neighbor control block.

Cause: The system has run low on resources and is out of system memory.

Action: Install more memory or upgrade router.

PIM.040

Level: U-TRACE

Short Syntax: PIM.040 Rcv Assert for mfcache entry, no ifc, grp *IP_group*, src *IP_source*, fip *fip_index*

Long Syntax: PIM.040 Received Assert on interface not in mfcache entry, Group *IP_group*, Source *IP_source*, fip *fip_index*

Description: Received a PIM Assert message on the indicated multicast forwarding interface index. However, this interface is not present in the respective mfcache entry.

PIM.041

Level: P-TRACE

Short Syntax: PIM.041 Rcv Assert for unk src, grp *IP_group*, src *IP_source*, fip = *fip_index*

Long Syntax: PIM.041 Received an Assert for unknown source, Group *IP_group*, Source *IP_source*, fip = *fip_index*

Description: Received a PIM Assert message on the indicated multicast forwarding interface index. However, the source address could not be located in the unicast routing table.

PIM.042

Level: U-TRACE

Short Syntax: PIM.042 Discard packet due to bad addr, family = *encode_family*, type = *encode_type*

Long Syntax: PIM.042 Packet was discarded due to a bad address, family = *encode_family*, type = *encode_type*

Description: A PIM message packet has been received with an encoded address with either an unsupported family or type. The packet was discarded.

PIM.043

Level: U-TRACE

Short Syntax: PIM.043 Bad addr, ignored, family = *encode_family*, type = *encode_type*

Long Syntax: PIM.043 Address in packet ignored due to bad address, family = *encode_family*, type = *encode_type*

Description: During parsing of a PIM control message, an encoded address was encountered with an unsupported family or type. Did not continue processing this address and ignored it, but continued further parsing of the packet.

PIM.044

Level: U-TRACE

Short Syntax: PIM.044 Ignored aggregated addr, msklen = *mask_length*

Long Syntax: PIM.044 Ignored address with mask length less than maximum, mask length = *mask_length*

Description: During parsing of a PIM control message, an encoded address was encountered with a mask length less than the address maximum. This indicates aggregation, which is not supported by PIM, so the address was ignored.

PIM.045

Level: U-TRACE

Short Syntax: PIM.045 Net *net_index* not pt-to-pt, reset hello to *hello_period* sec

Long Syntax: PIM.045 The net *net_index* is not a point-to-point, reset hello period to *hello_period* seconds

Description: When reading configuration information for a PIM interface, no hello period was specified to prevent the transmissions of hello messages after adjacency has occured. This is used for point to point interfaces only. If the interface being set up is not a point to point, the hello period is changed to the default hello period.

Chapter 94. Protocol Independent Multicast IPv6 (PIM6)

This chapter describes Protocol Independent Multicast IPv6 (PIM6) messages. For information on message content and how to use the message, refer to the Introduction.

PIM6.001

Level: C-TRACE

Short Syntax: PIM6.001 Add phyint *IP6_interface*

Long Syntax: PIM6.001 Add physical interface *IP6_interface*

Description: PIM has been enabled on the specified physical interface.

PIM6.002

Level: C-TRACE

Short Syntax: PIM6.002 Add tunnel *tunnel6_source*-> *tunnel6_destination*

Long Syntax: PIM6.002 Add tunnel *tunnel6_source*-> *tunnel6_destination*

Description: A PIM tunnel has been configured between the given source and destination addresses.

PIM6.003

Level: U-TRACE

Short Syntax: PIM6.003 Nbr *IP6_neighbor* removed, ifc down

Long Syntax: PIM6.003 Neighbor *IP6_neighbor* removed due to interface going down

Description: The PIM virtual interface is going down and this neighbor has been removed from the neighbor list.

PIM6.004

Level: UE-ERROR

Short Syntax: PIM6.004 bd hdr cks 0x *checksum* (exp 0x *expected_checksum*) *source_ip_address* -> *destination_ip_address*

Long Syntax: PIM6.004 Bad header checksum 0x *checksum* (expected 0x *expected_checksum*) in packet from *source_ip_address* for *destination_ip_address*

Description: This message is generated when a PIM control message has an invalid checksum. The received checksum, together with the correct checksum, are displayed.

Cause: Most likely, this is a damaged packet. It may be that another node is building an incorrect PIM control message.

Action: If the problem persists, examine a line trace to determine where the packet is being damaged.

PIM6.005

Level: U-TRACE

Short Syntax: PIM6.005 Rcvd PIM unk ver= *PIM_version IP6_source* -> *IP6_destination*, nt *network ID*

Long Syntax: PIM6.005 Received PIM Unknown Version= *PIM_version*, *IP6_source* -> *IP6_destination* net *network ID*

Description: A PIM Control Message was received with an unknown version number. Packet is discarded.

PIM6.006

Level: P-TRACE

Short Syntax: PIM6.006 Rcvd Hello Msg *IP6_source* -> *IP6_destination*, nt *network ID*

Long Syntax: PIM6.006 Received PIM Hello Message, *IP6_source* -> *IP6_destination* net *network ID*

Description: A PIM Hello Message has been received.

PIM6.007

Level: P-TRACE

Short Syntax: PIM6.007 Rcvd Jn/Prn Msg *IP6_source* => *PIM6_upstream*, nt *network ID*

Long Syntax: PIM6.007 Received PIM Join/Prune Message, *IP6_source* upstream *PIM6_upstream* net *network ID*

Description: A PIM Join/Prune Message has been received. The displayed address is not the IP packet destination address, but the upstream router address contained in the PIM protocol packet.

PIM6.008

Level: P-TRACE

Short Syntax: PIM6.008 Rcvd Assert Msg, grp *IP6_group*, src *IP6_source*, nt *network ID*

Long Syntax: PIM6.008 Received PIM Assert Message,

Group *IP6_group*, Source *IP6_source*, net *network ID*

Description: A PIM Assert Message has been received.

PIM6.009

Level: P-TRACE

Short Syntax: PIM6.009 Rcvd Graft Msg *IP6_source* -> *IP6_destination*, nt *network ID*

Long Syntax: PIM6.009 Received PIM Graft Message, *IP6_source* -> *IP6_destination* net *network ID*

Description: A PIM Graft Message has been received.

PIM6.010

Level: P-TRACE

Short Syntax: PIM6.010 Rcvd GraftAck Msg *IP6_source* -> *IP6_destination*, nt *network ID*

Long Syntax: PIM6.010 Received PIM Graft Ack Message, *IP6_source* -> *IP6_destination* net *network ID*

Description: A PIM Graft Acknowledgement Message has been received.

PIM6.011

Level: UE-ERROR

Short Syntax: PIM6.011 Rcvd unk msg *IP6_source* -> *IP6_destination*, nt *network ID*

Long Syntax: PIM6.011 Received PIM Unknown Message, *IP6_source* -> *IP6_destination* net *network ID*

Description: A PIM Control Message was received which has an unknown message type. The message is discarded.

PIM6.012

Level: P-TRACE

Short Syntax: PIM6.012 Send Hello Msg *IP6_source* -> *IP6_destination*, nt *network ID*

Long Syntax: PIM6.012 Sending PIM Hello Message, *IP6_source* -> *IP6_destination* net *network ID*

Description: A PIM Hello Message has been created and scheduled for transmission.

PIM6.013

Level: P-TRACE

Short Syntax: PIM6.013 Send Jn/Prn Msg *IP6_source* => *PIM6_upstream*, nt *network ID*

Long Syntax: PIM6.013 Sending PIM Join/Prune Message, *IP6_source* upstream *PIM6_upstream* net *network ID*

Description: A PIM Join/Prune Message has been created and scheduled for transmission. The displayed address is not the IP packet destination address (which is the all PIM router address, but the upstream router address contained in the PIM protocol packet.

PIM6.014

Level: P-TRACE

Short Syntax: PIM6.014 Send Assert Msg, grp *IP6_group*, src *IP6_source*, nt *network ID*

Long Syntax: PIM6.014 Sending PIM Assert Message, Group *IP6_group*, Source *IP6_source*, net *network ID*

Description: A PIM Assert Message has been created and scheduled for transmission. The Assert occurs when a multicast data packet was received on an output interface for a multicast entry in the forwarding tables or having received an assert from another router which should not be forwarding the multicast data in question. This action signifies that duplicate multicast data is being sent due to routing loops. The assert message resolves which router should be the forwarding router and ensures this condition does not persist.

PIM6.015

Level: P-TRACE

Short Syntax: PIM6.015 Send Graft Msg *IP6_source* -> *IP6_destination*, nt *network ID*

Long Syntax: PIM6.015 Sending PIM Graft Message, *IP6_source* -> *IP6_destination* net *network ID*

Description: A PIM Graft Message has been created and scheduled for transmission.

PIM6.016

Level: P-TRACE

Short Syntax: PIM6.016 Send GraftAck Msg *IP6_source* -> *IP6_destination*, nt *network ID*

Long Syntax: PIM6.016 Sending PIM Graft Ack Message, *IP6_source* -> *IP6_destination* net *network ID*

Description: A PIM Graft Acknowledgement Message has been created and scheduled for transmission.

PIM6.017

Level: UI-ERROR

Short Syntax: PIM6.017 Could not obtain iorb *IP6_source* -> *IP6_destination*

Long Syntax: PIM6.017 Unable to obtain an iob for send, *IP6_source* -> *IP6_destination*

Description: No input/output request block (iorb) was returned by the system which was required by PIM to build and send a PIM control message.

Cause: The system has run out of resources. This usually occurs due to lack of system buffer memory.

Action: If not a problem due to an errant application, install more memory and assign more system buffers.

PIM6.018

Level: UI-ERROR

Short Syntax: PIM6.018 Failed iorb send rqst *IP6_source -> IP6_destination*

Long Syntax: PIM6.018 IORB send request was unsuccessful, *IP6_source -> IP6_destination*

Description: A request to send an iorb containing a PIM control message returned a failure and the message was discarded.

Cause: The system refused to send an input/output request block. This may be due to an overburdened router or a errant application.

Action: Upgrade to a higer performance router.

PIM6.019

Level: U-TRACE

Short Syntax: PIM6.019 DR *PIM6_old_DR* replaced by *PIM6_new_DR* for nt *network ID*

Long Syntax: PIM6.019 Designated Router *PIM6_old_DR* is replaced by *PIM6_new_DR* net *network ID*

Description: Designated Router timed out and is replaced by the new Designated Router.

PIM6.020

Level: C-TRACE

Short Syntax: PIM6.020 Nbr *PIM6_neighbor* del for int *PIM6_interface*

Long Syntax: PIM6.020 Neighbor *PIM6_neighbor* Deleted from Interface *PIM6_interface*

Description: The Neighbor timed out since it did not send Hello message on time and will be removed.

PIM6.021

Level: UI-ERROR

Short Syntax: PIM6.021 No nbr blocks available for *PIM6_neighbor*, int *PIM6_interface*

Long Syntax: PIM6.021 No neighbor control blocks are available for *PIM6_neighbor*, Interface *PIM6_interface*

Description: The neighbor array is full and no neighbor control blocks are available for the newly discovered neighbor on the interface.

Cause: The number of PIM neighbors has exceeded the limit supported by the PIM implementation. This may effect the designated router selection process.

Action: Reduce the number of routers running PIM on the same subnet.

PIM6.022

Level: C-TRACE

Short Syntax: PIM6.022 Nbr *PIM6_neighbor* added to int *PIM6_interface*

Long Syntax: PIM6.022 Neighbor *PIM6_neighbor* added for Interface *PIM6_interface*

Description: A neighbor was added to interface due to receiving a Hello message.

PIM6.023

Level: C-TRACE

Short Syntax: PIM6.023 J/P Msg parsed, vif *PIM6_interface* gcnt *Group_count*, jcnt *Join_count*, pcnt *Prune_count*

Long Syntax: PIM6.023 Join/Prune message parsed, vif = *PIM6_interface*, groups = *Group_count*, joins = *Join_count*, prunes = *Prune_count*

Description: A PIM Join/Prune formatted message has been received and parsed. The counts give the total number of groups, joins, and prunes that were parsed. The PIM virtual interface is the interface the packet was received on.

PIM6.024

Level: UI-ERROR

Short Syntax: PIM6.024 No mem for msg q

Long Syntax: PIM6.024 Could not allocate memory for building a message queue element.

Description: A memory allocation failure occurred when creating a message queue element for the PIM message queues.

Cause: The system has run low on resources and is out of system memory.

Action: Install more memory or upgrade router.

PIM6.025

Level: U-TRACE

Short Syntax: PIM6.025 msg type unk, msg not queued

Long Syntax: PIM6.025 Could not queue message element due to unknown message type.

Description: The message scheduler could not place the message element on the appropriate message queue due to an unknown message type passed by the caller.

The message element is freed back to the memory pool and no other action taken.

PIM6.026

Level: UI-ERROR

Short Syntax: PIM6.026 No mem for state, src = *IP6_source*, grp = *IP6_group*, vif = *PIM6_interface*

Long Syntax: PIM6.026 Could not get memory for state element, source = *IP6_source*, group = *IP6_group*, interface = *PIM6_interface*

Description: A memory allocation failure occurred when creating a state element for the state database for the reported source group pair on the PIM interface shown.

Cause: The system has run low on resources and is out of system memory.

Action: Install more memory or upgrade router.

PIM6.027

Level: UI-ERROR

Short Syntax: PIM6.027 No mem for sgnode, src = *IP6_source*, grp = *IP6_group*, vif = *PIM6_interface*

Long Syntax: PIM6.027 No memory for source-group node, source = *IP6_source*, group = *IP6_group*, interface = *PIM6_interface*

Description: A memory allocation failure occurred when creating a source-group node element for the state database for the reported source group pair on the PIM interface shown.

Cause: The system has run low on resources and is out of system memory.

Action: Install more memory or upgrade router.

PIM6.028

Level: C-TRACE

Short Syntax: PIM6.028 New PRUNE state src *IP6_source* grp *IP6_group* vif *PIM6_interface*

Long Syntax: PIM6.028 A new PRUNE state has been created, source = *IP6_source*, group = *IP6_group*, vif = *PIM6_interface*

Description: A new PIM PRUNE state has been created and placed in the state database.

PIM6.029

Level: U-TRACE

Short Syntax: PIM6.029 Del not required src *IP6_source* grp *IP6_group* vif *PIM6_interface*

Long Syntax: PIM6.029 Delete request for state not in database, source = *IP6_source*, group = *IP6_group*, vif = *PIM6_interface*

Description: A request was processed to delete a state that does not exist in the PIM state database. No action was taken and control returned immediately to the caller.

PIM6.030

Level: C-TRACE

Short Syntax: PIM6.030 PRUNE state removed src *IP6_source* grp *IP6_group* vif *PIM6_interface*

Long Syntax: PIM6.030 A PRUNE was deleted from the state database, source = *IP6_source*, group = *IP6_group*, vif = *PIM6_interface*

Description: A PRUNE state has been successfully removed from the state database and removed from all the appropriate program queues. This PRUNE state now no longer exists.

PIM6.031

Level: UI-ERROR

Short Syntax: PIM6.031 srte has rte type that is UNKNOWN by PIM, src net = *IP6_source* type = *route_type*

Long Syntax: PIM6.031 Unicast entry has route type UNKNOWN by PIM, source net = *IP6_source*, route type = *route_type*

Description: PIM is attempting to determine local unicast routing metric preference for a particular multicast forwarding entry which is used in the PIM assert process. The route type of the unicast routing entry is not of any type known by PIM and the routing metric preference has been the 0x7FFFFFFF and the routing metric set to 0xFFFFFFFF, which are used for unknown routes. The result will usually lead to this router losing the assert process and pruning its oif from the multicast forwarding cache entry.

Cause: This is usually caused by running unicast routing protocols or other functions that update the unicast forwarding table that marks the unicast entry of a route type that PIM is not aware of. This should only occur if new protocols were added to the router and PIM implementation was not updated to support them. PIM may not be able to support the new functions or a newer version of PIM is required.

Action: Call customer support and inform them of problem. If all the unicast routing protocols and forwarding table modification mechanisms are supported by PIM, a new version is necessary.

PIM6.032

Level: P-TRACE

Short Syntax: PIM6.032 Rcv Pim Control Msg for net *network ID*

Long Syntax: PIM6.032 Received Pim Ctl Message for net *network ID*, which is not ready

Description: Received a PIM Control Message for an interface which is not ready to receive messages. The control message is discarded.

PIM6.033

Level: C-TRACE

Short Syntax: PIM6.033 PIM states have been cleared

Long Syntax: PIM6.033 The PIM state database has been cleared

Description: The PIM state database has thrown away all of its states. This is most commonly performed when the multicast forwarding cache has been thrown away due a unicast routing update.

PIM6.034

Level: U-TRACE

Short Syntax: PIM6.034 Group address not multicast, addr = *IP6_group*

Long Syntax: PIM6.034 An invalid group address was encountered, address = *IP6_group*

Description: During parsing of a PIM control message, a group address was encountered that did not qualify as a valid multicast address. The parser did not continue processing this address and ignored it, but continued further parsing of the packet.

PIM6.035

Level: U-TRACE

Short Syntax: PIM6.035 jp rcv pkt len err, len *error_length* vif *PIM6_interface*

Long Syntax: PIM6.035 join/prune parser error due to bad PIM packet counts, length = *error_length* vif = *PIM6_interface*

Description: While parsing a received PIM Join/Prune message, the parser discovered a length error. This occurs when the group, join, and prune count fields in the PIM packet itself is erroneous, reporting an incorrect number of group and source addresses contained in the packet.

PIM6.036

Level: C-TRACE

Short Syntax: PIM6.036 New JOIN state src *IP6_source* grp *IP6_group* vif *PIM6_Interface*

Long Syntax: PIM6.036 A new JOIN state has been created, source = *IP6_source*, group = *IP6_group*, vif = *PIM6_Interface*

Description: A new PIM PRUNE state has been created and placed in the state database.

PIM6.037

Level: C-TRACE

Short Syntax: PIM6.037 JOIN state removed src *IP6_source* grp *IP6_group* vif *PIM6_Interface*

Long Syntax: PIM6.037 A JOIN was deleted from the state database, source = *IP6_source*, group = *IP6_group*, vif = *PIM6_Interface*

Description: A JOIN state has been successfully removed from the state database and removed from all the appropriate program queues. This JOIN state now no longer exists.

PIM6.038

Level: U-TRACE

Short Syntax: PIM6.038 No PIM vif, *IP6_source* -> *IP6_destination*, net *network ID*

Long Syntax: PIM6.038 Could not find a PIM virtual interface, *IP6_source* -> *IP6_destination* net *network ID*

Description: A PIM control message has been received that could not be mapped to a PIM virtual interface. The packet is discarded.

PIM6.039

Level: UI-ERROR

Short Syntax: PIM6.039 No mem for rpf q

Long Syntax: PIM6.039 Could not allocate memory for building an rpf neighbor block.

Description: A memory allocation failure occurred when creating a reverse path forwarding neighbor control block.

Cause: The system has run low on resources and is out of system memory.

Action: Install more memory or upgrade router.

PIM6.040

Level: U-TRACE

Short Syntax: PIM6.040 Rcv Assert for mfcache entry, no ifc, grp *IP6_group*, src *IP6_source*, fip *fip_index*

Long Syntax: PIM6.040 Received Assert on interface not in mfcache entry, Group *IP6_group*, Source *IP6_source*, fip *fip_index*

Description: Received a PIM Assert message on the indicated multicast forwarding interface index. However, this interface is not present in the respective mfcache entry.

PIM6.041

Level: P-TRACE

Short Syntax: PIM6.041 Rcv Assert for unk src, grp *IP6_group*, src *IP6_source*, fip = *fip_index*

Long Syntax: PIM6.041 Received an Assert for unknown source, Group *IP6_group*, Source *IP6_source*, fip = *fip_index*

Description: Received a PIM Assert message on the indicated multicast forwarding interface index. However, the source address could not be located in the unicast routing table.

PIM6.042

Level: U-TRACE

Short Syntax: PIM6.042 Discard packet due to bad addr, family = *encode_family*, type = *encode_type*

Long Syntax: PIM6.042 Packet was discarded due to a bad address, family = *encode_family*, type = *encode_type*

Description: A PIM message packet has been received with an encoded address with either an unsupported family or type. The packet was discarded.

PIM6.043

Level: U-TRACE

Short Syntax: PIM6.043 Bad addr, ignored, family = *encode_family*, type = *encode_type*

Long Syntax: PIM6.043 Address in packet ignored due to bad address, family = *encode_family*, type = *encode_type*

Description: During parsing of a PIM control message, an encoded address was encountered with an unsupported family or type. Did not continue processing this address and ignored it, but continued further parsing of the packet.

PIM6.044

Level: U-TRACE

Short Syntax: PIM6.044 Ignored aggregated addr, msklen = *mask_length*

Long Syntax: PIM6.044 Ignored address with mask length less than maximum, mask length = *mask_length*

Description: During parsing of a PIM control message, an encoded address was encountered with a mask length less than the address maximum. This indicates aggregation, which is not supported by PIM, so the address was ignored.

PIM6.045

Level: U-TRACE

Short Syntax: PIM6.045 Net *net_index* not pt-to-pt, reset hello to *hello_period* sec

Long Syntax: PIM6.045 The net *net_index* is not a point-to-point, reset hello period to *hello_period* seconds

Description: When reading configuration information for a PIM interface, no hello period was specified to prevent the transmissions of hello messages after adjacency has occured. This is used for point to point interfaces only. If the interface being set up is not a point to point, the hello period is changed to the default hello period.

Chapter 95. QLLC Layer (over X25) Messages

This chapter describes QLLC Layer (over X25) Messages messages. For information on message content and how to use the message, refer to the Introduction.

QLLC.001

Level: C-INFO

Short Syntax: QLLC.001 reset pkt rcvd: nt *cep* st lcn ev

Long Syntax: QLLC.001 reset pkt rcvd: network *cep* state lcn event

Description: reset pkt rcvd.

QLLC.002

Level: C-INFO

Short Syntax: QLLC.002 reset cnf pkt rcvd: nt *cep* st lcn ev

Long Syntax: QLLC.002 reset cnf pkt rcvd: network *cep* state lcn event

Description: reset cnf pkt rcvd.

QLLC.003

Level: C-INFO

Short Syntax: QLLC.003 ckt busy cleared: nt *cep* st lcn ev

Long Syntax: QLLC.003 ckt busy cleared: network *cep* state lcn event

Description: ckt busy cleared.

QLLC.004

Level: C-INFO

Short Syntax: QLLC.004 ckt busy establish: nt *cep* st lcn ev

Long Syntax: QLLC.004 ckt busy establish: network *cep* state lcn event

Description: ckt busy establish.

QLLC.005

Level: C-INFO

Short Syntax: QLLC.005 QTEST_RSP timeout exceeded: nt *cep* st lcn ev

Long Syntax: QLLC.005 QTEST_RSP timeout exceeded: network *cep* state lcn event

Description: QTEST_RSP timeout exceeded check check .

QLLC.006

Level: C-INFO

Short Syntax: QLLC.006 QXID_RSP timeout exceeded: nt *cep* st lcn ev

Long Syntax: QLLC.006 QXID_RSP timeout exceeded: network *cep* state lcn event

Description: QXID_RSP timeout exceeded.

QLLC.007

Level: C-INFO

Short Syntax: QLLC.007 QDISCONTACT timeout exceeded: nt *cep* st lcn ev

Long Syntax: QLLC.007 QDISCONTACT timeout exceeded: network *cep* state lcn event

Description: QDISCONTACT timeout exceeded.

QLLC.008

Level: C-INFO

Short Syntax: QLLC.008 QCONTACT timeout exceeded: nt *cep* st lcn ev

Long Syntax: QLLC.008 QCONTACT timeout exceeded: network *cep* state lcn event

Description: QCONTACT timeout exceeded.

QLLC.009

Level: C-INFO

Short Syntax: QLLC.009 PVC CIRCUIT ACTIVE: nt *cep* st lcn ev

Long Syntax: QLLC.009 PVC CIRCUIT ACTIVE: network *cep* state lcn event

Description: PVC CIRCUIT ACTIVE. check check cep->port->hp_handle may be useful.

QLLC.010

Level: C-INFO

Short Syntax: QLLC.010 PLC changed to down: nt *cep* st lcn ev

Long Syntax: QLLC.010 PLC changed to down: network *cep* state lcn event

Description: PLC changed to down.

QLLC.011

Level: C-INFO

Short Syntax: QLLC.011 Q_CONTACT_RCV: nt *cep* st lcn ev

Long Syntax: QLLC.011 Q_CONTACT_RCV: network *cep* state lcn event

Description: Q_CONTACT_RCV.

QLLC.012

Level: C-INFO

Short Syntax: QLLC.012 Q_EXCPTN - UNEXP_CFIELD_RCVD: nt *cep* st lcn ev

Long Syntax: QLLC.012 Q_EXCPTN - UNEXP_CFIELD_RCVD: network *cep* state lcn event

Description: Q_EXCPTN - UNEXP_CFIELD_RCVD.

QLLC.013

Level: C-INFO

Short Syntax: QLLC.013 Q_CONTACT_CNF: nt *cep* st lcn ev

Long Syntax: QLLC.013 Q_CONTACT_CNF: network *cep* state lcn event

Description: Q_CONTACT_CNF.

QLLC.014

Level: C-INFO

Short Syntax: QLLC.014 Q_DISCONTACT_RCV: nt *cep* st lcn ev

Long Syntax: QLLC.014 Q_DISCONTACT_RCV: network *cep* state lcn event

Description: Q_DISCONTACT_RCV.

QLLC.015

Level: C-INFO

Short Syntax: QLLC.015 Q_DISCONTACT_CNF: nt *cep* st lcn ev

Long Syntax: QLLC.015 Q_DISCONTACT_CNF: network *cep* state lcn event

Description: Q_DISCONTACT_CNF.

QLLC.016

Level: C-INFO

Short Syntax: QLLC.016 Q_CIRCUIT_BUSY: nt *cep* st lcn ev

Long Syntax: QLLC.016 Q_CIRCUIT_BUSY: network *cep* state lcn event

Description: Q_CIRCUIT_BUSY.

QLLC.017

Level: C-INFO

Short Syntax: QLLC.017 Q_XID_CMD_RCV: nt *cep* st lcn ev

Long Syntax: QLLC.017 Q_XID_CMD_RCV: network *cep* state lcn event

Description: Q_XID_CMD_RCV.

QLLC.018

Level: C-INFO

Short Syntax: QLLC.018 Q_XID_RSP_RCV: nt *cep* st lcn ev

Long Syntax: QLLC.018 Q_XID_RSP_RCV: network *cep* state lcn event

Description: Q_XID_RSP_RCV.

QLLC.019

Level: C-INFO

Short Syntax: QLLC.019 b001 plc chg to dwn. nt

Long Syntax: QLLC.019 b001 plc change to down network

Description: b001 plc change to down.

QLLC.020

Level: C-INFO

Short Syntax: QLLC.020 b002 plc ckt bsy chg: nt *cep* lcn cst st

Long Syntax: QLLC.020 b002 plc ckt busy change: network *cep* for lcn cep state state

Description: b002 plc ckt busy change.

QLLC.021

Level: C-INFO

Short Syntax: QLLC.021 b003 clear_reset pkt rcvd: nt *cep* lcn cst st

Long Syntax: QLLC.021 b003 clear_reset pkt rcvd: network *cep* lcn cep state state

Description: b003 clear_reset pkt rcvd.

QLLC.022

Level: C-INFO

Short Syntax: QLLC.022 b004 q_pkt rcvd: nt *qa_field* lcn cst qa-field

Long Syntax: QLLC.022 b004 q_pkt rcvd: network *qa_field* lcn cep state qa_field

Description: b004 q_pkt rcvd.

QLLC.023

Level: C-INFO

Short Syntax: QLLC.023 b004 q_pkt rcvd: nt *qc_field* lcn cst qc-field

Long Syntax: QLLC.023 b004 q_pkt rcvd: network *qc_field* lcn cep state qc_field

Description: c_field.

QLLC.024

Level: C-INFO

Short Syntax: QLLC.024 b004 user data pkt rcvd: nt *cep* lcn st

Long Syntax: QLLC.024 b004 user data pkt rcvd: network *cep* lcn state

Description: b004 user data pkt rcvd.

QLLC.025

Level: C-INFO

Short Syntax: QLLC.025 b005 ckt active: nt *cep* lcn st

Long Syntax: QLLC.025 b005 ckt active: network *cep* lcn state

Description: b005 ckt active.

QLLC.026

Level: C-INFO

Short Syntax: QLLC.026 b006 plc change to up: nt

Long Syntax: QLLC.026 b006 plc change to up: network

Description: b006 plc change to up.

QLLC.027

Level: C-INFO

Short Syntax: QLLC.027 b019 link busy change: nt *link_status* lnk stat

Long Syntax: QLLC.027 b019 link busy change: network *link_status* link status

Description: b019 link busy change.

QLLC.028

Level: C-INFO

Short Syntax: QLLC.028 b020 Rx Incoming call: nt *peer* lcn st

Long Syntax: QLLC.028 b020 Rx Incoming call: nt *peer* lcn st

Description: b020 Rx Incoming call.

QLLC.029

Level: C-INFO

Short Syntax: QLLC.029 b022 get qdata: nt *peer* lcn st tsk

Long Syntax: QLLC.029 b022_get_qdata_pkt: network *peer* lcn state transmit task

Description: b019 Rxk call connected.

QLLC.030

Level: C-INFO

Short Syntax: QLLC.030 b023 rcv pkt: nt *tx_task* lcn st

Long Syntax: QLLC.030 b023_rcv_pkt_state_dr : network *tx_task* lcn state

Description: b023 receive packet state DR

QLLC.031

Level: C-INFO

Short Syntax: QLLC.031 b023 pkt rcvd state DR: nt *tx_task* lcn st

Long Syntax: QLLC.031 b023 pkt rcvd state DR: network *tx_task* lcn state

Description: b023 pkt rcvd state DR.

QLLC.032

Level: C-INFO

Short Syntax: QLLC.032 b024 crt rst or clr pkt: nt *cep* lcn st code

Long Syntax: QLLC.032 b024 create reset or clear pkt: network *cep* lcn state code

Description: b024 create reset or clear pkt.

QLLC.033

Level: C-INFO

Short Syntax: QLLC.033 s003 open port: nt *prtcl* prtcl

Long Syntax: QLLC.033 s003 open port: network *prtcl* protocol

Description: s003 open port: protocol.

QLLC.034

Level: C-INFO

Short Syntax: QLLC.034 s003 close port: nt *protocol* prtcl

Long Syntax: QLLC.034 s003 close port: network *protocol* protocol

Description: s003 close port.

QLLC.035

Level: C-INFO

Short Syntax: QLLC.035 s005 register station: nt *lcn* prtcl lcn hndl

Long Syntax: QLLC.035 s005 register station: network *lcn* protocol lcn handle

Description: s005 register station.

QLLC.036

Level: C-INFO

Short Syntax: QLLC.036 s006 unregister station: nt *cep* lcn

Long Syntax: QLLC.036 s006 unregister station: network *cep* lcn

Description: s006 unregister station.

QLLC.037

Level: C-INFO

Short Syntax: QLLC.037 s007 call req: nt *cep* prtcl

Long Syntax: QLLC.037 s007 call req: network *cep* protocol

Description: s007 call request

QLLC.038

Level: C-INFO

Short Syntax: QLLC.038 s008 clr call req: nt *cep* lcn code

Long Syntax: QLLC.038 s008 clr call req: network *cep* lcn code

Description: s011 data request prim.

QLLC.039

Level: C-INFO

Short Syntax: QLLC.039 s009 xid req: nt *cep* lcn state

Long Syntax: QLLC.039 s009 xid request : network *cep* lcn cep state :

Description: s009 xid request prim.

QLLC.040

Level: C-INFO

Short Syntax: QLLC.040 s011 data req: nt *cep* lcn state

Long Syntax: QLLC.040 s011 data req: network *cep* lcn cep state

Description: s011 data request prim

QLLC.041

Level: C-INFO

Short Syntax: QLLC.041 s012 xid resp: nt *cep* lcn modifier

Long Syntax: QLLC.041 s012 xid response : nt *cep* lcn modifier

Description: s017 q_rsp timer expired.

QLLC.042

Level: C-INFO

Short Syntax: QLLC.042 s013 tst resp: nt *cep* lcn state modifier

Long Syntax: QLLC.042 s013 tst resp: nt *cep* lcn state modifier

Description: s013 test response

QLLC.043

Level: C-INFO

Short Syntax: QLLC.043 s016 ckt bsy req: nt *cep* lcn state modifier

Long Syntax: QLLC.043 s016 ckt bsy req: nt *cep* lcn state modifier

Description: s016 circuit busy request

QLLC.044

Level: C-INFO

Short Syntax: QLLC.044 s017 rsp tmr expr: nt *cep* lcn state

Long Syntax: QLLC.044 s017 q response timer expired: network *cep* lcn state

Description: s022 contact request prim.

QLLC.045

Level: C-INFO

Short Syntax: QLLC.045 s018 set stn role: nt *cep* lcn state modifier

Long Syntax: QLLC.045 s018 set station role : network *cep* lcn state modifier

Description: s018 set station role

QLLC.046

Level: C-INFO

Short Syntax: QLLC.046 s019 inc call resp: nt *cep* lcn hnd modifier

Long Syntax: QLLC.046 s019 incoming call response: network *cep* lcn handle modifier

Description: s019 incoming call response

QLLC.047

Level: C-INFO

Short Syntax: QLLC.047 s029 rst req: nt *cep* lcn state modifier

Long Syntax: QLLC.047 s029 reset request: network *cep* lcn state modifier

Description: s029 reset request

QLLC.048

Level: C-INFO

Short Syntax: QLLC.048 s022 cont req: nt *cep* lcn state

Long Syntax: QLLC.048 s022 contact request: network *cep* lcn state

Description: s022 contact request

QLLC.049

Level: C-INFO

Short Syntax: QLLC.049 s023 cont resp: nt *cep* lcn state modifier

Long Syntax: QLLC.049 s023 contact response: network *cep* lcn state modifier

Description: s023 contact response.

QLLC.050

Level: C-INFO

Short Syntax: QLLC.050 s024 disc req prim: nt *cep* lcn state

Long Syntax: QLLC.050 s024 discontact request prim: nt *cep* lcn state

Description: s024 discontact request prim.

QLLC.051

Level: C-INFO

Short Syntax: QLLC.051 s025 disc resp prim: nt *cep* lcn state

Long Syntax: QLLC.051 s025 discontact response prim: nt *cep* lcn state

Description: s025 discontact response prim.

QLLC.052

Level: C-INFO

Short Syntax: QLLC.052 s027 strt q_rsp tmr: nt *cep* lcn state

Long Syntax: QLLC.052 start q_rsp timer: network *cep* lcn state

Description: start q_rsp timer.

QLLC.053

Level: C-INFO

Short Syntax: QLLC.053 s028 chg hnd prim: nt *cep* lcn state hndl

Long Syntax: QLLC.053 s028 change handle prim: network *cep* lcn state h_handle

Description: s028 change handle prim.

QLLC.054

Level: C-INFO

Short Syntax: QLLC.054 s029 abort call: nt *cep* lcn state

Long Syntax: QLLC.054 s029 abort call: network *cep* lcn state

Description: s029 abort call.

Chapter 96. Q.931 Signalling Layer 3 for ISDN (Q931)

This chapter describes Q.931 Signalling Layer 3 for ISDN (Q931) messages. For information on message content and how to use the message, refer to the Introduction.

Q931.001

Level: U-INFO

Short Syntax: Q931.001 SETUP recvd CRV (0x *crv*) from (*cgn*) to (*cdn*) Channel (B *chan*) bw *bw* kbps on isdn/ *intf* nt *net*

Long Syntax: Q931.001 Set up msg received from the switch with Call reference value (0x *crv*) from originator *cgn* to destination *cdn* on Channel number *chan* speed *bw* kbps on isdn *intf* network *net*

Description: Setup message received, getting ready for a data connection

Action: None

Q931.002

Level: U-INFO

Short Syntax: Q931.002 SETUP sent CRV (0x *crv*) from (*cgn*) to (*cdn*) Channel (B *chan*) bw *bw* kpbs on isdn/ *intf* nt *net*

Long Syntax: Q931.002 Set up msg sent to the switch with Call reference value (0x *crv*) from originator *cgn* to destination *cdn* on Channel number *chan* bw *bw* on isdn *intf* network *net*

Description: Setup message sent, getting ready for a data connection

Action: None

Q931.003

Level: U-INFO

Short Syntax: Q931.003 ALERT sent CRV (0x *crv*) on isdn/ *intf* nt *net*

Long Syntax: Q931.003 Alert msg sent to the switch with Call reference value (0x *crv*) on ISDN/ *intf* network *net*

Description: Alert message sent, check config to see if we can accept the call

Action: None

Q931.004

Level: U-INFO

Short Syntax: Q931.004 CONNECT sent CRV (0x *crv*) on isdn/ *intf* nt *net*

Long Syntax: Q931.004 Connect msg sent to the switch with Call reference value (0x *crv*) on isdn/ *intf* network *net*

Description: Connect message sent, start B-Channel Communication

Action: None

Q931.008

Level: U-INFO

Short Syntax: Q931.008 State change CRV (0x *crv*) connid *connid* from *oldstate* to *newstate* on isdn/ *intf* nt *net*

Long Syntax: Q931.008 Q931 state changed for Call reference value (0x *crv*), connid *connid* from *oldstate* to *newstate* on isdn/ *intf* network *net*

Description: State changed for a channel

Action: None

Q931.009

Level: U-INFO

Short Syntax: Q931.009 RESTART rcv CRV (0x *crv*) Ind[0]=0x *ind* channel= *chan* on isdn/ *intf* nt *net*

Long Syntax: Q931.009 Restart msg rcv from the switch with Call reference value (0x *crv*) ind *ind* Chan *chan* on isdn *intf* network *net*

Description: Restart message rcvd, terminate B-Channel communication

Action: None

Q931.018

Level: U-INFO

Short Syntax: Q931.018 Incoming SETUP rejected DN0 mismatch CDN (*cgn*) isdn/ *intf* nt *net*

Long Syntax: Q931.018 Set up msg received from the

switch did not have the right CDN (0x *cgn*) on isdn *intf* network *net*

Description: Setup message received, and incompatible DN0

Action: None

Q931.019

Level: U-INFO

Short Syntax: Q931.019 SETUP recvd CRV (0x *crv*) from (*cgn*)and rejected - incompatible BC (*bc1 bc2 bc3 bc4*) on nt isdn/ *intf* nt *net*

Long Syntax: Q931.019 Set up msg received from the switch with Call reference value (0x *crv*) from *cgn* with incompatible bearer capability *bc1 bc2 bc3 bc4* on isdn/ *intf* network *net*

Description: Setup message received, and rejected due to incompatible bearer caps.

Action: None

Q931.020

Level: U-INFO

Short Syntax: Q931.020 Clear Channel B *crv* send msg (0x *cgn*) crv (0x *bc*) cause (0x *intf*) on isdn/ nt

Long Syntax: Q931.020 Send a DISC/REL/REL COMP *crv* to call on chan *cgn* up msg to the switch with Call reference value (0x *bc*) cause *intf* on isdn/ network

Description: Setup message received, and rejected due to incompatible bearer caps.

Action: None

Panic q931ym

Short Syntax: YDC ISDN: mem alloc fld

Description: The YDC ISDN network handler failed to allocate sufficient memory during the initialization phase.

Action: Contact customer service.

Chapter 97. RIP for IPv6 (RIP6)

This chapter describes RIP for IPv6 (RIP6) messages. For information on message content and how to use the message, refer to the Introduction.

RIP6.001

Level: UE-ERROR

Short Syntax: RIP6.001 bd ver *version_number* frm hst *source_IPV6_address*

Long Syntax: RIP6.001 bad version *version_number* received from host *source_IPV6_address*

Description: The version field in the RIP6 header did not match the current version.

Cause: This is probably caused by an error in the source host.

Action: Contact the manufacturer of the source host and report the problem.

RIP6.002

Level: U-TRACE

Short Syntax: RIP6.002 Received RIP6 Request *source_ipv6_address -> dest_ipv6_address* on nt *Network ID*

Long Syntax: RIP6.002 Request received from host *source_ipv6_address* to *dest_ipv6_address* network *Network ID*

Description: A RIP6 routing table request was received from another host. A routing table update will be sent to it.

RIP6.003

Level: C-TRACE

Short Syntax: RIP6.003 Received RIP6 Resp *source_ipv6_address -> dest_ipv6_address* on nt *Network ID*

Long Syntax: RIP6.003 response received from host *source_ipv6_address* to *dest_ipv6_address* network *Network ID*

Description: A RIP6 routing table update was received. Note that it may take more than one response packet to transmit the entire routing table, especially if the routing table is large.

RIP6.004

Level: UE_ERROR

Short Syntax: RIP6.004 rip6_pkt *source_ipv6_address -> dest_ipv6_address* on nt *Network ID*

Long Syntax: RIP6.004 Bad RIP6 received from host *source_ipv6_address* to *dest_ipv6_address* network *Network ID*

Description: RIP6 packet, either request or reponse must have at least one RTE. It will be discarded.

RIP6.005

Level: UE_ERROR

Short Syntax: RIP6.005 rcv RIP6 update net *Network ID* from *src_addr* bad port *srcport*

Long Syntax: RIP6.005 receiving RIP6 update on net *Network ID* from *src_addr* with bad UDP6 source port *srcport*

Description: The router is receiving a RIP6 update/response with a bad UDP6 source port Source port must be 521.

RIP6.006

Level: UE-ERROR

Short Syntax: RIP6.006 bd cmd *command_code* frm *source_IP_address*

Long Syntax: RIP6.006 bad command code *command_code* received from host *source_IP_address*

Description: A RIP6 message was received with an unrecognized command code.

Cause: This is probably caused by an error or out of date software in the source host.

Action: Contact the manufacturer of the source host and report the problem.

RIP6.007

Level: UE_ERROR

Short Syntax: RIP6.007 rcv RIP6 update net *Network ID* from *src_addr* bad_dest_address *dst_addr*

Long Syntax: RIP6.007 receiving RIP6 update on net *Network ID* from *src_addr* with bad dest. address *dst_addr*

Description: The router is receiving a RIP6 update/response with bad destination address Destination address must be either a link local address of this intercae or a multicast address: FF02::9.

RIP6.008

Level: UE_ERROR

Short Syntax: RIP6.008 rcv RIP6 update net *Network ID* from *src_addr* bad rte's prefix *prefix*

Long Syntax: RIP6.008 receiving RIP6 update on net *Network ID* from *src_addr* with bad rte's prefix6 *prefix*

Description: The router is receiving a RIP6 update/response with a bad rte's prefix. Rte's prefix should not be a multicast address or a link local address.

RIP6.009

Level: UE_ERROR

Short Syntax: RIP6.009 rcv RIP6 update net *Network ID* from *src_addr* bad rte's metric *metric*

Long Syntax: RIP6.009 receiving RIP6 update on net *Network ID* from *src_addr* with bad rte's metric *metric*

Description: The router is receiving a RIP6 update/response with a bad rte's metric. Rte's metric must in the range of 1 and 16, inclusively.

RIP6.010

Level: U-INFO

Short Syntax: RIP6.010 Net *destination_IP_address* unrch via *next_hop_IP_address*, del

Long Syntax: RIP6.010 network *destination_IP_address* now unreachable via router *next_hop_IP_address*, deleted

Description: An incoming RIP6 update from the router that was previously listed as the next hop to the destination network has announced that the destination is unreachable (i.e. at metric 'infinity'). The RIP6 route to that destination is being deleted.

RIP6.011

Level: U-INFO

Short Syntax: RIP6.011 Update net *destination_IP_network* hps *metric* via *next_hop_IP_address*

Long Syntax: RIP6.011 update route to net *destination_IP_network* at metric *metric* hops via router *next_hop_IP_address*

Description: A new (better) route to the given destination has been learned via RIP6 and has been installed.

RIP6.012

Level: C-TRACE

Short Syntax: RIP6.012 Sending RIP6 Request *source_ipv6_address* -> *dest_ipv6_address* on nt *Network ID*

Long Syntax: RIP6.012 Sending RIP6 multicast Request from *source_ipv6_address* to *dest_ipv6_address* network *Network ID*

Description: The router is sending a RIP6 request from each interface which has just come up.

RIP6.013

Level: C-TRACE

Short Syntax: RIP6.013 Sending RIP6: *source_IP_address* -> *destination_IP_address* on net *Network ID packet_count* pkts *number_of_routes* rtes

Long Syntax: RIP6.013 Sending RIP6 from *source_IP_address* to *destination_IP_address* on net *Network ID* in *packet_count* packets with *number_of_routes* route enties

Description: The router is mutlicasting a RIP6 update (triggered either by a timer or a change in the routing table) to the specified address.

RIP6.014

Level: C-TRACE

Short Syntax: RIP6.014 Sending RIP6: *source_IP_address* -> *destination_IP_address* on net *Network ID packet_count* pkts *number_of_routes* rtes

Long Syntax: RIP6.014 Sending RIP6 from *source_IP_address* to *destination_IP_address* on net *Network ID* in *packet_count* packets with *number_of_routes* route enties

Description: The router is sending a RIP6 update (triggered by a request from another host) to the specified address.

RIP6.015

Level: CI-ERROR

Short Syntax: RIP6.015 cnt all pkt

Long Syntax: RIP6.015 cannot allocate packet for transmission

Description: When RIP6 went to allocate a packet for transmission (either for a request or reply), none was available.

RIP6.016

Level: C-TRACE

Short Syntax: RIP6.016 Sending RIP6: *source_IP_address* -> *destination_IP_address* on net *Network ID*

Long Syntax: RIP6.016 Sending RIP6 from *source_IP_address* to *destination_IP_address* on net *Network ID*

Description: A RIP6 packet (either a routing table

update, or when an interface first comes up, a request) was sent.

RIP6.017

Level: UI-ERROR

Short Syntax: RIP6.017 err *output_error_code* sending pkt nt *network*

Long Syntax: RIP6.017 error code *output_error_code* when sending packet out net *network*

Description: An outgoing reply packet was dropped as the result of some problem in the router.

Cause: There are many potential causes of this problem, such as an overloaded output queue, a down network, etc.

Action: Consult logging output from the relevant network subsystem for more information.

RIP6.018

Level: U-INFO

Short Syntax: RIP6.018 nt rt to *destination_IP_address* tmd out

Long Syntax: RIP6.018 network route to *destination_IP_address* timed out

Description: A route to a destination via some other router in the routing database has not been heard from for a while and is now being marked as unreachable.

RIP6.019

Level: U-INFO

Short Syntax: RIP6.019 nt rt to *destination_IP_address* del

Long Syntax: RIP6.019 network route to *destination_IP_address* deleted

Description: A route to a destination via some other router in the routing database has not been heard from for a while, has been marked unreachable, and is now being deleted.

Panic rip6udperr

Short Syntax: RIP6 udp port not avail

Description: Another application registered previously with rip's UDP port.

Action: Contact customer service.

RIP6.020

Level: UE_ERROR

Short Syntax: RIP6.020 rcv RIP6 update net *Network ID* from *src_addr* bad rte's prefix_len *prefix_len*

Long Syntax: RIP6.020 receiving RIP6 update on net *Network ID* from *src_addr* with bad rte prefixlen *prefix_len*

Description: The router is receiving a RIP6 update/response with a bad rte's prefix len. Rte's prefix len must in the range of 0 and 128, inclusively.

RIP6.021

Level: UE_ERROR

Short Syntax: RIP6.021 rcv RIP6 update net *Network ID* from *src_addr* bad next hop rte prefix *prefix*

Long Syntax: RIP6.021 receiving RIP6 update on net *Network ID* from *src_addr* with bad next hop rte prefix6 *prefix*

Description: The router is receiving a RIP6 update/response with a bad next hop rte prefix. Rte's prefix should be zero or a link local address.

RIP6.022

Level: UE_ERROR

Short Syntax: RIP6.022 rcv RIP6 update net *Network ID* from *src_addr* bad next hop rte rtag *rtag*

Long Syntax: RIP6.022 receiving RIP6 update on net *Network ID* from *src_addr* with bad next hop rte route tag *rtag*

Description: The router is receiving a RIP6 update/response with a bad next hop rte route tag Next Hop Rte route tag MUST be zero.

RIP6.023

Level: UE_ERROR

Short Syntax: RIP6.023 rcv RIP6 update net *Network ID* from *src_addr* bad next hop rte prefix_len *plen*

Long Syntax: RIP6.023 receiving RIP6 update on net *Network ID* from *src_addr* with bad next hop rte prefix len *plen*

Description: The router is receiving a RIP6 update/response with a bad next hop rte prefix len Next Hop Rte prefix len MUST be zero.

RIP6.024

Level: UE_ERROR

Short Syntax: RIP6.024 rcv RIP6 update on net *Network ID* bad srcaddr *src_addr*

Long Syntax: RIP6.024 receiving RIP6 update on net *Network ID* with bad source address *src_addr*

Description: The router is receiving a RIP6 update/response with bad source address. Source address must be a link local address.

RIP6.025

Level: UE_ERROR

Short Syntax: RIP6.025 rcv RIP6 update net *Network ID* from *src_addr* bad hopcount *hopcount*

Long Syntax: RIP6.025 receiving RIP6 update on net *Network ID* from *src_addr* with bad hopcount *hopcount*

Description: The router is receiving a RIP6 update/response with a bad hop count. Hop count must be 255.

Chapter 98. Router FDDL (RTRF)

This chapter describes Router FDDL (RTRF) messages. For information on message content and how to use the message, refer to the Introduction.

RTRF.001

Level: UE_ERROR

Short Syntax: RTRF.001 *message network_interface*

Long Syntax: RTRF.001 *message network_interface*

Description: The message describes details of the error, and the error is associated with the network_interface

RTRF.002

Level: UE_ERROR

Short Syntax: RTRF.002 *message*

Long Syntax: RTRF.002 *message*

Description: The message describes details of the error

RTRF.003

Level: C_INFO

Short Syntax: RTRF.003 *message IP_Address/ IP_Mask* for inbound intf *network_interface*

Long Syntax: RTRF.003 *message IP_Address/ IP_Mask* for inbound network interface *network_interface*

Description: The message describes information about an IP_Address and IP_Mask associated with inbound traffic on a network_interface

RTRF.004

Level: C_INFO

Short Syntax: RTRF.004 *message IPX_NetAddress/ IPX_NodeAddress* for inbound intf *network_interface*

Long Syntax: RTRF.004 *message IPX_NetAddress/ IPX_NodeAddress* for inbound network interface *network_interface*

Description: The message describes information about an IPX_NetAddress and IPX_NodeAddress associated with inbound traffic on a network_interface

RTRF.005

Level: C_INFO

Short Syntax: RTRF.005 *message IPX_NetAddress* for inbound intf *network_interface*

Long Syntax: RTRF.005 *message IPX_NetAddress* for inbound network interface *network_interface*

Description: The message describes information about an IPX_NetAddress associated with inbound traffic on a network_interface

RTRF.006

Level: UI_ERROR

Short Syntax: RTRF.006 *message*

Long Syntax: RTRF.006 *message*

Description: An unusual, internal error has occurred. Details are explained in the message.

RTRF.007

Level: C_INFO

Short Syntax: RTRF.007 *message IP_Address/ IP_Mask*

Long Syntax: RTRF.007 *message IP_Address/ IP_Mask*

Description: The message describes information about an IP_Address and IP_Mask.

RTRF.008

Level: C_INFO

Short Syntax: RTRF.008 *message IPX_NetAddress*

Long Syntax: RTRF.008 *message IPX_NetAddress*

Description: The message describes information about an IPX_NetAddress.

RTRF.009

Level: C_INFO

Short Syntax: RTRF.009 *message*

Long Syntax: RTRF.009 *message*

Description: The message describes details of the error

RTRF.010

Level: U_INFO

Short Syntax: RTRF.010 *message network_interface*

Long Syntax: RTRF.010 *message network_interface*

Description: The message describes details of the event, and the event is associated with the network_interface

---

RTRF.011

Level: UI_ERROR

Short Syntax: RTRF.011 *routine_name*: FDDL msg error, *message_name*, net *network_interface*

Long Syntax: RTRF.011 *routine_name*: FDDL message error, message *message_name*, for net *network_interface*

Description: An error condition has been returned to FDDL when sending the specified message to the lower level API. The routine name given is place of occurance that the error was encountered.

Action: There is no action required. Errors of this type may occur due to temporary problems caused by congestion and other events which may cause message requests to be canceled.

RTRF.012

Level: U_INFO

Short Syntax: RTRF.012 *routine_name*: Route add canceled, *message*, net *network_interface*

Long Syntax: RTRF.012 *routine_name*: Route add was canceled due to *message* on net *network_interface*

Description: The route add request was canceled due to the reason described in the message.

RTRF.013

Level: C_INFO

Short Syntax: RTRF.013 *routine_name*: Route *message*, Grp *IP_Group*, Src *IP_Source*, net *network_interface*, outputs *oif_number*

Long Syntax: RTRF.013 *routine_name*: Route *message*, Group *IP_Group*, Source *IP_Source*, net *network_interface*, number of output ifc *oif_number*

Description: A forwarding route has been either added or deleted to the lower layer databases. The message displayed signifies whether the action was an add or delete.

RTRF.014

Level: C_INFO

Short Syntax: RTRF.014 *routine_name*: MC Addr *IP_Address message*, net *network_interface*

Long Syntax: RTRF.014 *routine_name*: Multicast Address *IP_Address* has been *message*, net *network_interface*

Description: A registered multicast address has either been added or deleted to the lower layer databases. The message displayed signifies whether the action was an add or delete.

RTRF.015

Level: C_INFO

Short Syntax: RTRF.015 *routine_name*: MC Fwding *message*, net *network_interface*

Long Syntax: RTRF.015 *routine_name*: Multicast forwarding has been *message* for net *network_interface*

Description: FDDL has changed the status of multicast forwarding for the lower layer databases at the designated net interface, either enabling or disabling IP multicast forwarding.

RTRF.016

Level: C_INFO

Short Syntax: RTRF.016 *routine_name*: *indication_type* Indication received, net *network_interface*

Long Syntax: RTRF.016 *routine_name*: *indication_type* Indication received for net *network_interface*

Description: The given indication was received from the lower layer API by FDDL for the specified network interface.

RTRF.017

Level: U_INFO

Short Syntax: RTRF.017 *routine_name*: net *network_interface* is now *interface_status*

Long Syntax: RTRF.017 *routine_name*: network interface *network_interface* is now *interface_status*

Description: FDDL has received notification of an interface status change. The new interface status is described in the message.

RTRF.018

Level: UI_ERROR

Short Syntax: RTRF.018 *routine_name*: FDDL *function_name* registration failure

Long Syntax: RTRF.018 *routine_name*: FDDL *function_name* registration failure

Description: A failure was reported when FDDL attempted to register with lower layer databases. This will not allow the given function to operate normally and will not be available for further use.

Action: An error here signifies that initialization did not proceed correctly. Lower layer hardware is not allowing the upper layer functions access to its databases. Investigate for hardware problems.

RTRF.019

Level: C_INFO

Short Syntax: RTRF.019 *message IPX_NetAddress/ IP_NodeAddress*

Long Syntax: RTRF.019 *message IPX_NetAddress/ IP_NodeAddress*

Description: The message describes information about an IPX_NetAddress/IPX_NodeAddress pair

---

RTRF.020

Level: U_INFO

Short Syntax: RTRF.020 *function_name*: placeholder

Long Syntax: RTRF.020 *function_name*: placeholder

Description: none

RTRF.021

Level: U_INFO

Short Syntax: RTRF.021 *function_name*: placeholder

Long Syntax: RTRF.021 *function_name*: placeholder

Description: none

RTRF.022

Level: U_INFO

Short Syntax: RTRF.022 *function_name*: placeholder

Long Syntax: RTRF.022 *function_name*: placeholder

Description: none

RTRF.023

Level: U_INFO

Short Syntax: RTRF.023 *function_name*: placeholder

Long Syntax: RTRF.023 *function_name*: placeholder

Description: none

RTRF.024

Level: U_INFO

Short Syntax: RTRF.024 *function_name*: placeholder

Long Syntax: RTRF.024 *function_name*: placeholder

Description: none

RTRF.025

Level: U_INFO

Short Syntax: RTRF.025 *function_name*: placeholder

Long Syntax: RTRF.025 *function_name*: placeholder

Description: none

RTRF.026

Level: U_INFO

Short Syntax: RTRF.026 *function_name*: placeholder

Long Syntax: RTRF.026 *function_name*: placeholder

Description: none

RTRF.027

Level: U_INFO

Short Syntax: RTRF.027 *function_name*: placeholder

Long Syntax: RTRF.027 *function_name*: placeholder

Description: none

RTRF.028

Level: U_INFO

Short Syntax: RTRF.028 *function_name*: placeholder

Long Syntax: RTRF.028 *function_name*: placeholder

Description: none

RTRF.029

Level: U_INFO

Short Syntax: RTRF.029 *function_name*: placeholder

Long Syntax: RTRF.029 *function_name*: placeholder

Description: none

RTRF.030

Level: U_INFO

Short Syntax: RTRF.030 *function_name*: placeholder

Long Syntax: RTRF.030 *function_name*: placeholder

Description: none

RTRF.031

Level: U_INFO

Short Syntax: RTRF.031 *function_name*: placeholder

Long Syntax: RTRF.031 *function_name*: placeholder

Description: none

RTRF.032

Level: U_INFO

Short Syntax: RTRF.032 *function_name*: placeholder

Long Syntax: RTRF.032 *function_name*: placeholder

Description: none

RTRF.033

Level: U_INFO

Short Syntax: RTRF.033 *function_name*: placeholder

Long Syntax: RTRF.033 *function_name*: placeholder

Description: none

RTRF.034

Level: U_INFO

Short Syntax: RTRF.034 *function_name*: placeholder

Long Syntax: RTRF.034 *function_name*: placeholder

Description: none

RTRF.035

Level: U_INFO

Short Syntax: RTRF.035 *function_name*: placeholder

Long Syntax: RTRF.035 *function_name*: placeholder

Description: none

RTRF.036

Level: U_INFO

Short Syntax: RTRF.036 *function_name*: placeholder

Long Syntax: RTRF.036 *function_name*: placeholder

Description: none

RTRF.037

Level: U_INFO

Short Syntax: RTRF.037 *function_name*: placeholder

Long Syntax: RTRF.037 *function_name*: placeholder

Description: none

RTRF.038

Level: U_INFO

Short Syntax: RTRF.038 *function_name*: placeholder

Long Syntax: RTRF.038 *function_name*: placeholder

Description: none

RTRF.039

Level: U_INFO

Short Syntax: RTRF.039 *function_name*: placeholder

Long Syntax: RTRF.039 *function_name*: placeholder

Description: none

RTRF.040

Level: U_INFO

Short Syntax: RTRF.040 *function_name*: placeholder

Long Syntax: RTRF.040 *function_name*: placeholder

Description: none

RTRF.041

Level: U_INFO

Short Syntax: RTRF.041 *function_name*: placeholder

Long Syntax: RTRF.041 *function_name*: placeholder

Description: none

RTRF.042

Level: U_INFO

Short Syntax: RTRF.042 *function_name*: placeholder

Long Syntax: RTRF.042 *function_name*: placeholder

Description: none

RTRF.043

Level: U_INFO

Short Syntax: RTRF.043 *function_name*: placeholder

Long Syntax: RTRF.043 *function_name*: placeholder

Description: none

RTRF.044

Level: U_INFO

Short Syntax: RTRF.044 *function_name*: placeholder

Long Syntax: RTRF.044 *function_name*: placeholder

Description: none

RTRF.045

Level: U_INFO

Short Syntax: RTRF.045 *function_name*: placeholder

Long Syntax: RTRF.045 *function_name*: placeholder

Description: none

RTRF.046

Level: U_INFO

Short Syntax: RTRF.046 *function_name*: placeholder

Long Syntax: RTRF.046 *function_name*: placeholder

Description: none

RTRF.047

Level: U_INFO

Short Syntax: RTRF.047 *function_name*: placeholder

Long Syntax: RTRF.047 *function_name*: placeholder

Description: none

RTRF.048

Level: U_INFO

Short Syntax: RTRF.048 *function_name*: placeholder

Long Syntax: RTRF.048 *function_name*: placeholder

Description: none

RTRF.049

Level: U_INFO

Short Syntax: RTRF.049 *function_name*: placeholder

Long Syntax: RTRF.049 *function_name*: placeholder

Description: none

RTRF.050

Level: U_INFO

Short Syntax: RTRF.050 *function_name*: placeholder

Long Syntax: RTRF.050 *function_name*: placeholder

Description: none

Chapter 99. Routing Information Protocol (RIP)

This chapter describes Routing Information Protocol (RIP) messages. For information on message content and how to use the message, refer to the Introduction.

RIP.001

Level: UE-ERROR

Short Syntax: RIP.001 bd ver *version_number* frm hst *source_IP_address*

Long Syntax: RIP.001 bad version *version_number* received from host *source_IP_address*

Description: The version field in the RIP header did not match the current version.

Cause: This is probably caused by an error in the source host.

Action: Contact the manufacturer of the source host and report the problem.

RIP.002

Level: U-TRACE

Short Syntax: RIP.002 rq frm *source_IP_address*

Long Syntax: RIP.002 request received from host *source_IP_address*

Description: A RIP routing table request was received from another host. A routing table update will be sent to it.

RIP.003

Level: U-INFO

Short Syntax: RIP.003 trc on *tracing_file* frm *source_IP_address*

Long Syntax: RIP.003 trace on to *tracing_file* received from host *source_IP_address*

Description: A request from a host to turn RIP tracing on to a given log file was received. The router ignores this request.

RIP.004

Level: U-INFO

Short Syntax: RIP.004 trc off frm *source_IP_address*

Long Syntax: RIP.004 trace off received from host *source_IP_address*

Description: A request from a host to turn RIP tracing off was received. The router ignores this request.

RIP.005

Level: C-TRACE

Short Syntax: RIP.005 rsp frm *source_IP_address*

Long Syntax: RIP.005 response received from host *source_IP_address*

Description: A RIP routing table update was received. Note that it may take more than one response packet to transmit the entire routing table, especially if the routing table is large.

RIP.006

Level: UE-ERROR

Short Syntax: RIP.006 bd cmd *command_code* frm *source_IP_address*

Long Syntax: RIP.006 bad command code *command_code* received from host *source_IP_address*

Description: A RIP message was received with an unrecognized command code.

Cause: This is probably caused by an error or out of date software in the source host.

Action: Contact the manufacturer of the source host and report the problem.

RIP.007

Level: UE-ERROR

Short Syntax: RIP.007 rsp frm off nt *source_IP_address*

Long Syntax: RIP.007 response received from off network host *source_IP_address*

Description: A RIP routing update response was received from a machine which was not directly attached to the network the response came in on. The packet is discarded.

Cause: Since normal RIP software is generally written to send data only to connected nets, this is probably indicative of a hostile event.

Action: Examine audit trails and other information to determine the original source host.

RIP.009

Level: U-TRACE

Short Syntax: RIP.009 dyn rt to *destination_IP_network* frm *next_hop_IP_address* dis

Long Syntax: RIP.009 dynamic route to *destination_IP_network* from *next_hop_IP_address* disallowed

Description: A dynamic route was received but is being ignored because the configuration of RIP receive policy for the interface disallows acceptance of this route.

RIP.010

Level: U-INFO

Short Syntax: RIP.010 nt *destination_IP_address* unrch via *next_hop_IP_address*, del

Long Syntax: RIP.010 network *destination_IP_address* now unreachable via router *next_hop_IP_address*, deleted

Description: An incoming RIP update from the router that was previously listed as the next hop to the destination network has announced that the destination is unreachable (i.e. at metric 'infinity'). The RIP route to that destination is being deleted.

RIP.011

Level: U-INFO

Short Syntax: RIP.011 updt nt *destination_IP_network* hps *metric* via *next_hop_IP_address*

Long Syntax: RIP.011 update route to net *destination_IP_network* at metric *metric* hops via router *next_hop_IP_address*

Description: A new (better) route to the given destination has been learned via RIP and has been installed.

RIP.012

Level: C-TRACE

Short Syntax: RIP.012 snd rqst *source_IP_address*

Long Syntax: RIP.012 send request from address *source_IP_address*

Description: The router is sending a RIP request from each of the addresses associated with an interface which has just come up.

RIP.013

Level: C-TRACE

Short Syntax: RIP.013 snd brd to *destination_IP_address* *packet_count* pkts *number_of_routes* rtes

Long Syntax: RIP.013 sending broadcast response to address *destination_IP_address* in *packet_count* packets with *number_of_routes* routes

Description: The router is sending a normal RIP broadcast update (triggered either by a timer or a change in the routing table) to the specified address.

RIP.014

Level: C-INFO

Short Syntax: RIP.014 snd to *destination_IP_address* *packet_count* pkts *number_of_routes* rtes

Long Syntax: RIP.014 sending response to address *destination_IP_address* in *packet_count* packets with *number_of_routes* routes

Description: The router is sending a RIP update (triggered by a request from another host) to the specified address.

RIP.015

Level: CI-ERROR

Short Syntax: RIP.015 cnt all pkt

Long Syntax: RIP.015 cannot allocate packet for transmission

Description: When RIP went to allocate a packet for transmission (either for a request or reply), none was available.

RIP.016

Level: C-TRACE

Short Syntax: RIP.016 snd pkt *destination_IP_address*

Long Syntax: RIP.016 sending packet to *destination_IP_address*

Description: A RIP packet (either a routing table update, or when an interface first comes up, a request) was sent.

RIP.017

Level: UI-ERROR

Short Syntax: RIP.017 err *output_error_code* sndng pkt nt *network*

Long Syntax: RIP.017 error code *output_error_code* when sending packet out net *network*

Description: An outgoing reply packet was dropped as the result of some problem in the router.

Cause: There are many potential causes of this problem, such as an overloaded output queue, a down network, etc.

Action: Consult logging output from the relevant network subsystem for more information.

RIP.018

Level: U-INFO

Short Syntax: RIP.018 nt rt to *destination_IP_address* tmd out

Long Syntax: RIP.018 network route to *destination_IP_address* timed out

Description: A route to a destination via some other router in the routing database has not been heard from for a while and is now being marked as unreachable.

RIP.019

Level: U-INFO

Short Syntax: RIP.019 nt rt to *destination_IP_address* del

Long Syntax: RIP.019 network route to *destination_IP_address* deleted

Description: A route to a destination via some other router in the routing database has not been heard from for a while, has been marked unreachable, and is now being deleted.

Panic ripudperr

Short Syntax: rip udp port not avail

Description: Another application registered previously with rip's UDP port.

Action: Contact customer service.

RIP.020

Level: U-INFO

Short Syntax: RIP.020 ver *version_number* frm hst *source_IP_address* intf *source_IP_interface*

Long Syntax: RIP.020 Mismatch version *version_number* received from host *source_IP_address* on interface *source_IP_interface*

Description: The version field in the RIP header did not match the current version on the receive interface..

Cause: This is probably caused by a configuration error in the source host.

Action: Correct the configuration in the source host.

RIP.021

Level: UE-ERROR

Short Syntax: RIP.021 bd auth frm hst *source_IP_address* intf *source_IP_interface*

Long Syntax: RIP.021 Authentication error received from host *source_IP_address* on interface *source_IP_interface*

Description: The packet is reject due to authentication err caused either by invalid authentication info or authentication is not enable.

Cause: This is probably caused by a misconfiguration.

Action: Correct the configuration.

RIP.022

Level: C-TRACE

Short Syntax: RIP.022 snd RIP2 to *destination_IP_address* from *source_IP_address* *packet_count* pkts *number_of_routes* rtes

Long Syntax: RIP.022 sending RIP2 response to address *destination_IP_address* from *source_IP_address* in *packet_count* packets with *number_of_routes* routes

Description: The router is sending a normal RIP2 update (triggered either by a timer or a change in the routing table) to the specified address.

Chapter 100. RSVP

This chapter describes RSVP messages. For information on message content and how to use the message, refer to the Introduction.

RSVP.001

Level: C-INFO

Short Syntax: RSVP.001 Initializing RSVP; status= *init_status*.

Long Syntax: RSVP.001 Initializing RSVP function; result status is *init_status*.

Description: The RSVP init routine is called to perform initialization functions. The result status is displayed (NOT CONFIGURED, STARTED, CONFIG ERROR).

RSVP.002

Level: C-INFO

Short Syntax: RSVP.002 RSVP stopped at clock *time_now*.

Long Syntax: RSVP.002 RSVP function is stopped by operator at system clock *time_now*.

Description: The operator stopped RSVP function using op-console command.

RSVP.004

Level: U-INFO

Short Syntax: RSVP.004 Rcvd net *n_net updown_state* msg; net-type= *net_type* b/w=

Long Syntax: RSVP.004 RSVP received network *n_net updown_state* message; network type= *net_type* bandwith=

Description: RSVP received an upcall regarding a network interface UP/DOWN status

RSVP.007

Level: U-INFO

Short Syntax: RSVP.007 RSVP not enabled on i/f *Interface* due to *Reason_string*

Long Syntax: RSVP.007 Enable RSVP on interface *Interface* failed due to *Reason_string*

Description: Enable-RSVP on an interface failed because of link config conflicts such as BRS already configured on the link or other reasons.

RSVP.010

Level: UE-ERROR

Short Syntax: RSVP.010 Not RSVP V1 vrsn *RSVPVersion* fm *SourceIPAddress*

Long Syntax: RSVP.010 Not RSVP V1 version *RSVPVersion* in packet from *SourceIPAddress*

Description: An RSVP packet with a bad version number was received. (rsvp_rx_process)

RSVP.011

Level: UE-ERROR

Short Syntax: RSVP.011 Bad RSVP checksum *RSVPCheckSum* fm *SourceIPAddress"*

Long Syntax: RSVP.011 Invalid RSVP checksum *RSVPCheckSum* in packet from *SourceIPAddress"*

Description: An RSVP packet with a bad checksum was received. (rsvp_rx_process)

RSVP.012

Level: UE-ERROR

Short Syntax: RSVP.012 Bad RSVP pkt lngth *RSVPPacketLength* fm *SourceIPAddress*

Long Syntax: RSVP.012 Bad RSVP packet length *RSVPPacketLength* in packet from *SourceIPAddress*

Description: An RSVP packet with a bad length was received. (rsvp_rx_process)

RSVP.015

Level: UE-ERROR

Short Syntax: RSVP.015 Bad RSVP objct lngth *RSVPObjectLength* in objct cla *RSVPObjectClass* from *SourceIPAddress*

Long Syntax: RSVP.015 A bad RSVP object length *RSVPObjectLength* in obj cla *RSVPObjectClass* in packet from *SourceIPAddress*

Description: The RSVP object length was not a multiple of 4, was less than 4, or it's next object pointer was bad. (rsvp_map_pkt)

RSVP.016

Level: UE-ERROR

Short Syntax: RSVP.016 Bad RSVP objct class *RSVPObjectClass* from *SourceIPAddress*

Long Syntax: RSVP.016 RSVP object class *RSVPObjectClass* not defined in packet from *SourceIPAddress*

Description: The RSVP object class was not defined. (rsvp_map_pkt)

RSVP.017

Level: UE-ERROR

Short Syntax: RSVP.017 Fltr without flw from *SourceIPAddress*

Long Syntax: RSVP.017 RSVP filter spec received before flow spec in packet from *SourceIPAddress*

Description: An filterspec was received with a flowspec. (rsvp_map_pkt)

RSVP.018

Level: UE-ERROR

Short Syntax: RSVP.018 RSVP objct lngth err *SourceIPAddress*

Long Syntax: RSVP.018 RSVP object length was bad in packet from *SourceIPAddress*

Description: An RSVP object's length was bad. (rsvp_map_pkt)

RSVP.021

Level: UE-ERROR

Short Syntax: RSVP.021 RSVP src *SenderTemplate*/dstn *Session* cnflct fm *SourceIPAddress*

Long Syntax: RSVP.021 An RSVP source *SenderTemplate*/destination *Session* conflict from *SourceIPAddress*

Description: Either the RSVP sender template or the filter spec does not match the session. (rsvp_check_srcport)

RSVP.022

Level: UE-ERROR

Short Syntax: RSVP.022 RSVP Pth mssng tmplt *SenderTemplate* or tspc *SenderTSpec* fm *SourceIPAddress*

Long Syntax: RSVP.022 RSVP Path message has no template *SenderTemplate* or tspec *SenderTSpec* from *SourceIPAddress*

Description: An RSVP Path message is missing either a sender_template or a sender_tspec. (rsvp_check_sender)

RSVP.026

Level: UE-ERROR

Short Syntax: RSVP.026 RSVP WF Resv mssng flow fm *SourceIPAddress*

Long Syntax: RSVP.026 RSVP WF RESV message missing flowspec from *SourceIPAddress*

Description: An RSVP Resv message in wildcard filter (WF) style is missing flowspec information. (rsvp_check_flow)

RSVP.027

Level: UE-ERROR

Short Syntax: RSVP.027 RSVP *RSVPStyle* Resv mssng fltr or flow fm *SourceIPAddress*

Long Syntax: RSVP.027 RSVP *RSVPStyle* RESV message missing filterspec or flowspec from *SourceIPAddress*

Description: An RSVP Resv message in fixed filter (FF) style or shared explicit (SE) style is missing filterspec or flowspec information. (rsvp_check_flow)

RSVP.028

Level: UE-ERROR

Short Syntax: RSVP.028 RSVP Resv unkn styl *RSVPStyle* fm *SourceIPAddress*

Long Syntax: RSVP.028 RSVP RESV message contains unknown style *RSVPStyle* from *SourceIPAddress*

Description: An RSVP Resv message has an unknown or unsupported style (rsvp_check_flow)

RSVP.031

Level: UE-ERROR

Short Syntax: RSVP.031 RSVP msg type *MsgType* mssng sssn fm *SourceIPAddress*

Long Syntax: RSVP.031 RSVP message type *MsgType* rcvd with missing session from *SourceIPAddress*

Description: An RSVP message is missing a session object (rsvp_msg_integrity)

RSVP.032

Level: UE-ERROR

Short Syntax: RSVP.032 RSVP Path mssng hp or tm fm *SourceIPAddress*

Long Syntax: RSVP.032 RSVP Path message missing hop or time from *SourceIPAddress*

Description: An RSVP Path message is missing a hop or time value (rsvp_msg_integrity)

RSVP.033

Level: UE-ERROR

Short Syntax: RSVP.033 RSVP Resv mssng hp, tm or styl fm *SourceIPAddress*

Long Syntax: RSVP.033 RSVP Resv message missing hop, time or style from *SourceIPAddress*

Description: An RSVP Resv message is missing rsvp_shop, time or style information. (rsvp_msg_integrity)

RSVP.034

Level: UE-ERROR

Short Syntax: RSVP.034 RSVP Path rrr mssng rrr fm *SourceIPAddress*

Long Syntax: RSVP.034 RSVP Path error message missing error spec from *SourceIPAddress*

Description: An RSVP Path error message is missing error_spec information. (rsvp_msg_integrity)

RSVP.035

Level: UE-ERROR

Short Syntax: RSVP.035 RSVP Resv rrr mssng rrr or styl fm *SourceIPAddress*

Long Syntax: RSVP.035 RSVP Resv error message missing error or style from *SourceIPAddress*

Description: An RSVP Resv error message is missing error_spec or style_spec information. (rsvp_msg_integrity)

RSVP.036

Level: UE-ERROR

Short Syntax: RSVP.036 RSVP Path tr mssng hop *RsvpHop* fm *SourceIPAddress*

Long Syntax: RSVP.036 RSVP Path tear message missing rsvp_hop *RsvpHop* from *SourceIPAddress*

Description: An RSVP Path tear message is missing rsvp_hop information. (rsvp_msg_integrity)

RSVP.037

Level: UE-ERROR

Short Syntax: RSVP.037 RSVP Resv tr mssng hop, scp, or styl from *SourceIPAddress*

Long Syntax: RSVP.037 RSVP Resv tear message missing hop, scope or style from *SourceIPAddress*

Description: An RSVP Path tear message is missing rsvp_hop, scope, or style information. (rsvp_msg_integrity)

RSVP.038

Level: UE-ERROR

Short Syntax: RSVP.038 RSVP Resv cnf mssng rrr spc, cnfrm or styl fm *SourceIPAddress*

Long Syntax: RSVP.038 RSVP Resv confirm message missing error spec, confirm or style from *SourceIPAddress*

Description: An RSVP Resv confirm message is missing error_spec, resv_confirm, or style information. (rsvp_msg_integrity)

RSVP.039

Level: UE-ERROR

Short Syntax: RSVP.039 Unknwn RSVP msg *msgtype* rcvd fm *SourceIPAddress*

Long Syntax: RSVP.039 Unknown RSVP message *msgtype* received from *SourceIPAddress*

Description: An RSVP Path tear message is missing rsvp_hop, scope, or style. (rsvp_msg_integrity)

RSVP.041

Level: C-TRACE

Short Syntax: RSVP.041 Sess *sess_id*: *port* rt chg: new i_i/f= *i_if*; new out-mask= *out_mask*

Long Syntax: RSVP.041 Rte chg on Sessn *sess_id* port *port* detected; new input port= *i_if*; new out mask= *out_mask*

Description: A route change on Session %I port %d has been detected; the new input interface is now %n; the new output interface mask is now %x

---

RSVP.046

Level: UE-ERROR

Short Syntax: RSVP.046 prt ncnsstncy

Long Syntax: RSVP.046 Port inconsistancy (tbd)

Description: rsvp_chk_port complained (rsvp_proc_path)

RSVP.047

Level: UI-ERROR

Short Syntax: RSVP.047 Cant add RSVP pth state fm nt *network ID*

Long Syntax: RSVP.047 Cannot add new RSVP Path state information from net *network ID*

Description: An RSVP Path message with new state cannot be added to either the full RSVP table, ie, RSVP_TABLE_NEW failed (rsvp_proc_path)

RSVP.048

Level: UI-ERROR

Short Syntax: RSVP.048 RSVP pth has bad tm int *time_values* nt *network ID*

Long Syntax: RSVP.048 RVSP path message has bad time interval *time_values* net *network ID*

Description: An RSVP Path message's time values are either longer than the maximum allowed or shorter than the minimum allowed (rsvp_proc_path)

RSVP.049

Level: UI-ERROR

Short Syntax: RSVP.049 NULL input if *network ID*

Long Syntax: RSVP.049 Empty input *network ID*

Description: Local session with a null input interface (later). (rsvp_proc_path)

RSVP.050

Level: UI-ERROR

Short Syntax: RSVP.050 RSVP Rte failure to dest *dst_addr*

Long Syntax: RSVP.050 RSVP route query to dest *dst_addr* failed

Description: RSVP query to IP routing database results in no route to the destination

---

RSVP.051

Level: UI-ERROR

Short Syntax: RSVP.051 Cant updt TC fltr nt *network ID*

Long Syntax: RSVP.051 Cannot update Traffic Control Filter net *network ID*

Description: An RSVP path message caused a change in traffic control, but the traffic control filter could not be updated. (rsvp_proc_path)

RSVP.052

Level: UI-ERROR

Short Syntax: RSVP.052 Can't get *entry_type* entry for snder *SenderAddress* prot *Protocol* port *SenderPort* to session *SessionAddress* port *SessPort*

Long Syntax: RSVP.052 Can't get *entry_type* entry space for sender *SenderAddress* prot *Protocol* port *SenderPort* to session *SessionAddress* port *SessPort*

Description: The specified entry table is running out of space while processing the specified Path request

---

RSVP.062

Level: CE-ERROR

Short Syntax: RSVP.062 RSVP *RSVPStyle* resv not *Action* from *SourceIPAddress* code *reasoncode*

Long Syntax: RSVP.062 An *RSVPStyle* style RSVP resv could not be *Action*from *SourceIPAddress* vcrm code *reasoncode*

Description: Either a new reservation could not be established or an old reservation could not be increased as a result of an RSVP resv message. (rsvp_proc_FF, SE, or WF)

RSVP.064

Level: UI-ERROR

Short Syntax: RSVP.064 PktClas update error for Session *SessionAddress* Prot *Protocol* Port *SessionPort* out-i/f *Outport*

Long Syntax: RSVP.064 Packet Classifier update error for Session *SessionAddress* Prot *Protocol* Port *SessionPort* out-i/f *Outport*

Description: The call to update packet classifer failed during processing of Resv message for the session

---

RSVP.065

Level: UE-ERROR

Short Syntax: RSVP.065 *MsgT* for sess *SessionAddress*: *SessionProt* at i/f *NetNum* discarded (*Reason*)

Long Syntax: RSVP.065 msg *MsgT* for sess *SessionAddress*: *SessionProt* at i/f *NetNum* discarded due to *Reason*

Description: Received an RSVP message that's out of state; discarded.

RSVP.066

Level: P-TRACE

Short Syntax: RSVP.066 Rsv Failed *reason value policydb value*

Long Syntax: RSVP.066 Reservation Failed due to *reason value policydb value*

Description: Reservation failed, prohibited by policy

RSVP.067

Level: C-TRACE

Short Syntax: RSVP.067 RSVP Plcy Qry *PlcyQryResults* Cd *Code*

Long Syntax: RSVP.067 RSVP Policy Query *PlcyQryResults* code *Code*

Description: RSVP queried policy with these results

RSVP.068

Level: C-TRACE

Short Syntax: RSVP.068 IP rtr *ChangeReason* to subnet *DestSubnet* mask *DestMask*

Long Syntax: RSVP.068 IP route change notification (route *ChangeReason*) to dest subnet *DestSubnet* msk *DestMask*

Description: Received IP route-change notification (changed|deleted) to destination subnet. This trace is displayed only if the route is used by RSVP.

RSVP.069

Level: C-TRACE

Short Syntax: RSVP.069 Local *upcallType* upcall event

Long Syntax: RSVP.069 *upcallType* upcall event

Description: Received RSVP control message for a local (i.e. to the router) application. The message text list the type of upcall event.

RSVP.070

Level: P-TRACE

Short Syntax: RSVP.070 Rcvd *RSVPStyle* msg for sess *SessionAddress*: *Protocol* prot *SessionPort* from *OrigAddress*

Long Syntax: RSVP.070 Valid RSVP msg type *RSVPStyle* rcv'd for sess *SessionAddress* port *Protocol* prot *SessionPort* from *OrigAddress*

Description: A RSVP message (Path, Resv FF/SE/WF) was received for a particular session (SessionAddress, Protocol, SessionPort) from an IP node with OrigAddress.

RSVP.071

Level: C-TRACE

Short Syntax: RSVP.071 *type* RESV from *SourceAddress*: *SourePort* to *DestAddress*: *DestPort* Protocol *Protocol* on out-i/f *OPort*

Long Syntax: RSVP.071 *type* RSVP RESV entry in pkt classifier for flow from *SourceAddress* Port *SourePort* to *DestAddress* Port *DestPort* Protocol *Protocol* on out-i/f *OPort*

Description: An entry has been added/deleted in the RSVP packet classifier for the specified session/flow on the specific outbound interface

---

RSVP.072

Level: C-TRACE

Short Syntax: RSVP.072 *type* PATH state from *SourceAddress*: *SourePort* prot *Protocol* to *DestAddress*: *DestPort*

Long Syntax: RSVP.072 *type* a Path state for flow from *SourceAddress* port *SourePort* protocol *Protocol* to *DestAddress* port *DestPort*

Description: Added/deleted a Path state for flow from a source port to a destination port (session)

RSVP.073

Level: P-TRACE

Short Syntax: RSVP.073 --RSVP send IP pkt to *Dest_Address* on net *Netp*, return code= *retcode*

Long Syntax: RSVP.073 RSVP sends an IP packet out to *Dest_Address* on net *Netp*, with return code *retcode*

Description: As a result of RSVP internal state and event, sent a RSVP message to a next-hop RSVP router or host

---

RSVP.074

Level: P-TRACE

Short Syntax: RSVP.074 Send *msg_type* for session *SessionAddress*: *SessionPort*

Long Syntax: RSVP.074 Send a RSVP message type *msg_type* for session *SessionAddress* port *SessionPort*

Description: Send a specified type of refresh or tear message to the specified session

---

RSVP.075

Level: U-TRACE

Short Syntax: RSVP.075 *type* state timeout from *SourceAddress*: *SourePort* prot *Protocol* to *DestAddress*: *DestPort*

Long Syntax: RSVP.075 A *type* state for flow from *SourceAddress* port *SourePort* protocol *Protocol* to *DestAddress* port *DestPort* timed out

Description: A PATH or RESV state for flow from a source port to a destination port (session) has just timed out and been removed.

RSVP.076

Level: P-TRACE

Short Syntax: RSVP.076 Forward QoS pkt from *Src_Address* to *Dest_Address* prot *Protocol* rt-code= *retcode*

Long Syntax: RSVP.076 Packet Classifier forwards a

QoS pkt from *Src_Address* to *Dest_Address* protocol *Protocol*; return code= *retcode*

Description: Packet Classifier identifies a packet in a QoS flow, and forwards this packet to the appropriate queue.

RSVP.077

Level: U-TRACE

Short Syntax: RSVP.077 Pkt Classifier table flushed.

Long Syntax: RSVP.077 Packet Classifier table is flushed!

Description: The Packet Classifer table is flushed at init time or due to Operator Console command.

RSVP.078

Level: P-TRACE

Short Syntax: RSVP.078 Send pos RESV-Confirm Pkt back to *recever_addr*

Long Syntax: RSVP.078 A RESV Confirm packet is sent back to *recever_addr* as requested.

Description: A RESV Confirm message is sent to the receiver (the RESV originator) who requested a reservation confirmation on the RESV message.

RSVP.079

Level: U-TRACE

Short Syntax: RSVP.079 Ntwk DISC rcvd frm *next_hop* on sess *sess_addr*: *port_num*

Long Syntax: RSVP.079 A network DISC received from next-hop *next_hop* on session *sess_addr* port *port_num*

Description: A network or remote host generated DISC is received from a next hop router regarding a session.

RSVP.080

Level: U-TRACE

Short Syntax: RSVP.080 *var1= v1*; *var2= v2*; *var3= v3*; *var4= v4*

Long Syntax: RSVP.080 component var1 *var1= v1*; var2 *var2= v2*; var3 *var3= v3*; var4 *var4= v4*

Description: A generic trace for unusual events; var1 shows module name and first trace variable name, v1 shows first trace variable value; var2 shows second trace variable name, v2 shows second trace variable value, and so on.

Chapter 101. SDLC Relay (SRLY)

This chapter describes SDLC Relay (SRLY) messages. For information on message content and how to use the message, refer to the Introduction.

SRLY.001

Level: C-INFO

Short Syntax: SRLY.001 SRLY interface initialization starting network *networkID*

Long Syntax: SRLY.001 SRLY initialization started on network *networkID*

Description: SRLY forwarder has started initialization on the relay interface.

SRLY.002

Level: C-INFO

Short Syntax: SRLY.002 SRLY interface initialization complete network *networkID*

Long Syntax: SRLY.002 SRLY initialization completed on network *networkID*

Description: SRLY forwarder has completed initialization on the relay interface.

SRLY.003

Level: C-TRACE

Short Syntax: SRLY.003 SRLY frame received on network *networkID*

Long Syntax: SRLY.003 SRLY frame received on network *networkID*

Description: SDLC Relay frame received.

SRLY.004

Level: C-INFO

Short Syntax: SRLY.004 SRLY frame sent on network *networkID*

Long Syntax: SRLY.004 SRLY frame sent on network *networkID*

Description: SDLC Relay frame transmitted.

SRLY.005

Level: UI-ERROR

Short Syntax: SRLY.005 SRLY frame discarded for group *group_number* on network *networkID* - *discard_reason*

Long Syntax: SRLY.005 SRLY frame discarded for group *group_number* on network *networkID* - *discard_reason*

Description: SRLY frame discarded.

SRLY.006

Level: CI-ERROR

Short Syntax: SRLY.006 SRLY memory allocation failed

Long Syntax: SRLY.006 A SRLY memory allocation request failed

Description: A SRLY memory allocation request for resources has failed.

SRLY.007

Level: CI-ERROR

Short Syntax: SRLY.007 SRLY port defined for non-SDLC net or invalid hdw - net *networkID*

Long Syntax: SRLY.007 SRLY port defined for non-SDLC net or invalid hdw - network *networkID*

Description: SRLY port defined on non-SDLC net or using invalid hardware - disabled.

SRLY.008

Level: CI-ERROR

Short Syntax: SRLY.008 Configuration error for group *group_number* - *configuration_error* - group internally disabled

Long Syntax: SRLY.008 Configuration error for group *group_number* - *configuration_error* - group internally disabled

Description: Group configuration error.

SRLY.009

Level: CI-ERROR

Short Syntax: SRLY.009 SRLY frame discarded on network *networkID* - *discard_reason*

Long Syntax: SRLY.009 SRLY frame discarded on network *networkID* - *discard_reason*

Description: SRLY frame discarded.

SRLY.010

Level: CI-ERROR

Short Syntax: SRLY.010 SRLY frame discarded - *discard_reason*

Long Syntax: SRLY.010 SRLY frame discarded - *discard_reason*

Description: SRLY frame discarded.

Panic srlyudperr

Short Syntax: SDLC Relay UDP port not available\r\n

Description: Another application registered previously with SDlC Relay's UDP port.

Action: Contact customer service.

Chapter 102. Security Protocol (SEC)

This chapter describes Security Protocol (SEC) messages. For information on message content and how to use the message, refer to the Introduction.

SEC.001

Level: C-INFO

Short Syntax: SEC.001 Tacacs+: *A message*

Long Syntax: SEC.001 TacacsPlus Message: *A message*

Description: Generic Message for Tacacs Plus

SEC.002

Level: C-INFO

Short Syntax: SEC.002 Tacx+StartPacket: *A message*

Long Syntax: SEC.002 TacacsPlus StartPacket Message: *A message*

Description: Generic Message for Tacacs Plus Start Packet

SEC.004

Level: U-INFO

Short Syntax: SEC.004 Tacacs conn to *neighbor* open on sprt *sourceport* dprt *destinationport*

Long Syntax: SEC.004 Tacacs connection to neighbor *neighbor* open on soure port *sourceport* destination port *destinationport*

Description: An OPEN message has been received on this connection for this neighbor.

Cause: The connection to the neighbor has completed successfully.

Action: None. This is an informational message.

SEC.005

Level: C-INFO

Short Syntax: SEC.005 Tacx+ContinuePacket: *A message*

Long Syntax: SEC.005 TacacsPlus ContinuePacket Message: *A message*

Description: Generic Message for Tacacs Plus continue packet

SEC.006

Level: C-INFO

Short Syntax: SEC.006 Tacx+ReplyPacket: *A message*

Long Syntax: SEC.006 TacacsPlus ReplyPacket Message: *A message*

Description: Generic Message for Tacacs Plus reply packet

SEC.007

Level: C-INFO

Short Syntax: SEC.007 TacPlus: [*id,*] *A message*

Long Syntax: SEC.007 TacacsPlus Message: *id, A message*

Description: Generic Message for Tacacs Plus provides request id.

SEC.008

Level: U-INFO

Short Syntax: SEC.008 TacPlus: [*id,*] *A message*

Long Syntax: SEC.008 TacacsPlus Message: [*id,*] *A message*

Description: Generic Message for Tacacs Plus clean path messages

SEC.009

Level: ERROR

Short Syntax: SEC.009 TacPlus: [*id,*] *A message*

Long Syntax: SEC.009 Tacacs-Plus Message: [*id,*] *A message*

Description: Generic Message for Tacacs Plus

SEC.010

Level: C-INFO

Short Syntax: SEC.010 TacPlus: rq[*id,*] tcp[*id2,*] *A message*

Long Syntax: SEC.010 TacacsPlus Message: *id, id2, A message*

Description: Generic Message for Tacacs Plus shows req id and tcp id

SEC.011

Level: C-INFO

Short Syntax: SEC.011 *A message*

Long Syntax: SEC.011 Message: *A message*

Description: Generic Message for Security Protocol

SEC.016

Level: C-INFO

Short Syntax: SEC.016 UDP port *port* not hooked

Long Syntax: SEC.016 UDP port *port* not hooked

Description: AuthenticationProtocol could not hook the UDP port to receive packets on

SEC.017

Level: C-INFO

Short Syntax: SEC.017 Rcvd Resp for unknown id *id*

Long Syntax: SEC.017 Received Response for unknown id *id*

Description: Authentication Protocol received a response that did not match any outstanding requests

SEC.018

Level: C-INFO

Short Syntax: SEC.018 Rcvd Invalid Authenticator

Long Syntax: SEC.018 Received Invalid Authenticator

Description: Radius received a packet with an invalid authenticator and discarded it

SEC.019

Level: C-INFO

Short Syntax: SEC.019 *direction packetType*

Long Syntax: SEC.019 *direction packetType* packet

Description: Authentication protocol UDP packet type received or sent

SEC.020

Level: C-INFO

Short Syntax: SEC.020 No Srvr Cfd

Long Syntax: SEC.020 No Server Addresses Configured packet

Description: No Server addresses were configured for authentication protocol

SEC.021

Level: C-INFO

Short Syntax: SEC.021 Radius hooked UDP port *port*

Long Syntax: SEC.021 Radius hooked UDP port *port*\

Description: Radius hooked the UDP port to receive radius packets on

SEC.022

Level: C-INFO

Short Syntax: SEC.022 *direction packetType* to *address* via *src* port *port*

Long Syntax: SEC.022 *direction packetType* packet to *address* source *src* port *port*

Description: Radius packet type sent to address and port specified

SEC.023

Level: C-INFO

Short Syntax: SEC.023 Auth *result* user= *user*

Long Syntax: SEC.023 Authentication *result* user= *user*

Description: Authentication passed or failed

SEC.024

Level: C-INFO

Short Syntax: SEC.024 Auth Req Outstanding for *compId*

Long Syntax: SEC.024 Auth Req Outstanding for *compId*

Description: Authentication request is already outstanding on this net, so disregarding new request

SEC.025

Level: C-INFO

Short Syntax: SEC.025 Request List at Max = *maxSize*

Long Syntax: SEC.025 Request List at Max = *maxSize*

Description: request list has reached the maximum size and a request had to be discarded

SEC.026

Level: C-INFO

Short Syntax: SEC.026 *action* Request id= *id* compID= *size* list size=

Long Syntax: SEC.026 *action* Request id= *id* compID= *size* list size=

Description: adding/removing a request to the security list

SEC.027

Level: C-INFO

Short Syntax: SEC.027 *action* compID= *id*

Long Syntax: SEC.027 *action* compID= *id*

Description: a security action taking place

SEC.028

Level: C-INFO

Short Syntax: SEC.028 Tacacs hooked UDP port *port*

Long Syntax: SEC.028 Tacacs hooked UDP port *port*\

Description: Tacacs hooked the UDP port to receive tacacs packets on

---

SEC.029

Level: C-INFO

Short Syntax: SEC.029 *direction packetType* for id *rqid* to *address* via *src* port *port*

Long Syntax: SEC.029 *direction packetType* packet for request id *rqid* to *address* source *src* port *port*

Description: Radius packet type sent to address and port specified

---

SEC.030

Level: C-INFO

Short Syntax: SEC.030 *action* compID= *id* net: *net*

Long Syntax: SEC.030 *action* completionID= *id* network number: *net*

Description: a security action taking place on net

Chapter 103. Serial Line Network Interface (SL)

This chapter describes Serial Line Network Interface (SL) messages. For information on message content and how to use the message, refer to the Introduction.

SL.001

Level: CI-ERROR

Short Syntax: SL.001 no bfr available for slftst on nt *network ID*

Long Syntax: SL.001 no buffer available for selftest on network *network ID*

Description: A packet buffer was not available when the interface self-test needed one.

SL.007

Level: U-TRACE

Short Syntax: SL.007 slftst started on nt *network ID*

Long Syntax: SL.007 selftest started on network *network ID*

Description: Self-test is being started on the serial line.

SL.019

Level: UE-ERROR

Short Syntax: SL.019 cbl typ *cable_type* nt compt wth lvl cnvt typ *level_converter_type*, nt *network ID*

Long Syntax: SL.019 Cable of type *cable_type* is not compatible with level converter of type *level_converter_type*, network *network ID*

Description: The cable and the level converter on the interface are not compatible with each other. The self-test will fail.

Cause: Wrong cable type for level converter.

Action: Use correct cable type.

Cause: If cable_type is "none", no cable.

Action: Connect adapter cable.

Cause: Cable broken so that it does not indicate cable type correctly (very unlikely).

Action: Replace cable.

SL.020

Level: UI-ERROR

Short Syntax: SL.020 *cable_type* can't be used with *internal_external* clk, nt *network ID*

Long Syntax: SL.020 *cable_type* cable cannot be used with *internal_external* clocking enabled, network *network ID*

Description: There is an incompatibility between the mode of the cable (DCE or DTE) and the type of clocking used. The interface will not be brought up.

Cause: DTE cable with internal clocking.

Action: Use DCE cable or external clocking.

Cause: DCE cable with external clocking.

Action: Use DTE cable or internal clocking.

SL.021

Level: CE-ERROR

Short Syntax: SL.021 slf tst failed, mdm sts: CTS = *cts*, DSR = *dsr*, DCD = *dcd*, nt *network ID*

Long Syntax: SL.021 Self test failed because of modem status: CTS = *cts*, DSR = *dsr*, DCD = *dcd*, network *network ID*

Description: The interface failed self test because at least one of the modem signals was off. The present state of the modem signals is shown in the ELS message. The normal state of the modem signals is CTS=ON, DSR=ON, and DCD=ON for RS-232, V.35, and V.36. For X.21, the normal state of the Indication signal is ON. In the ELS message, DCD represents the X.21 Indication signal. For HSSI, the normal state of the CA signal is ON. In the ELS message, DCD represents the HSSI CA signal.

Cause: Cable not connected to modem.

Action: Connect cable.

Cause: Modem not powered up.

Action: Power up modem.

Cause: Modem does not have good connection to other end of line (especially DCD OFF).

Action: Solve modem problem.

SL.022

Level: C-INFO

Short Syntax: SL.022 Modem status change CTS = *cts*, DSR = *dsr*, DCD = *dcd*, nt *network ID*

Long Syntax: SL.022 Modem status change CTS = *cts*, DSR = *dsr*, DCD = *dcd*, on network *network ID*

Description: A modem status change has occurred. The present state of the modem signals is shown in the ELS message. The normal state of the modem signals is CTS=ON, DSR=ON, and DCD=ON for RS-232, V.35, and V.36. For X.21, the normal state of the Indication signal is ON. In the ELS message, DCD represents the X.21 Indication signal. For HSSI, the normal state of the CA signal is ON. In the ELS message, DCD represents the HSSI CA signal.

SL.023

Level: CE-ERROR

Short Syntax: SL.023 int dwn due to mdm sts: CTS = *cts*, DSR = *dsr*, DCD = *dcd*, nt *network ID*

Long Syntax: SL.023 Interface down because of modem status: CTS = *cts*, DSR = *dsr*, DCD = *dcd*, network *network ID*

Description: The interface was brought down because one of the modem signals was off. The normal state of the modem signals is CTS=ON, DSR=ON, and DCD=ON for RS-232, V.35, and V.36. For X.21, the normal state of the Indication signal is ON. In the ELS message, DCD represents the X.21 Indication signal. For HSSI, the normal state of the CA signal is ON. In the ELS message, DCD represents the HSSI CA signal.

SL.024

Level: UI-ERROR

Short Syntax: SL.024 conf frame sz *configured_size* too large, reducing to *maximum_size*, nt *network ID*

Long Syntax: SL.024 Configured frame size of *configured_size* bytes too large, reducing to *maximum_size* bytes, network *network ID*

Description: The user-configured frame size for this interface is larger than the maximum that is allowed for the particular serial line device. The size is reduced to the largest allowable one.

Cause: Configuration in excess of allowable size.

Action: Reconfigure size and restart.

SL.027

Level: UI-ERROR

Short Syntax: SL.027 No level conv, disabling nt *network ID*

Long Syntax: SL.027 No level converter, disabling network *network ID*

Description: There is no level converter on this port of the serial adapter. The self-test will fail, and future self-tests will be cancelled.

Cause: No level converter.

Action: Add level converter to port.

Cause: Defective level converter which reads as not installed.

Action: Replace level converter.

SL.028

Level: UI-ERROR

Short Syntax: SL.028 Unk level conv *converter_type*, disabling nt *network ID*

Long Syntax: SL.028 Unknown level converter type *converter_type*, disabling network *network ID*

Description: There is a level converter of an unknown type on this port of the serial adapter. The self-test will fail, and future self-tests will be cancelled.

Cause: Unknown type of level converter.

Action: Upgrade to newer software that supports this type of level converter.

Cause: Defective level converter which reads as unknown type.

Action: Replace level converter.

SL.034

Level: UE-ERROR

Short Syntax: SL.034 no cable installed, nt *network ID*

Long Syntax: SL.034 No cable installed or installed cable broken or non-compatible, network *network ID*

Description: The system does not detect an adapter cable for the network interface. Self-test will fail.

Cause: No cable installed.

Action: Connect the correct adapter cable.

Cause: Cable broken so that it does not indicate cable type correctly (very unlikely).

Action: Replace cable.

Chapter 104. Server Cache Synchronization Protocol (SCSP)

This chapter describes Server Cache Synchronization Protocol (SCSP) messages. For information on message content and how to use the message, refer to the Introduction.

SCSP.002

Level: U-TRACE

Short Syntax: SCSP.002 Add SG nt *network* sg *server_group* rc *return_code*

Long Syntax: SCSP.002 Add Server Group. Network *network* SGID *server_group* rc *return_code*

Description: A server group was added. The return code of 0 indicates immediate success. The 1483 client may become active later. Look for a ELS indicating SG UP.

SCSP.003

Level: U-TRACE

Short Syntax: SCSP.003 Del SG nt *network* sg *server_group* rc *return_code*

Long Syntax: SCSP.003 Delete Server Group. Network *network* SGID *server_group* rc *return_code*

Description: A server group was deleted. The return code of 0 indicates success.

SCSP.004

Level: UE-ERROR

Short Syntax: SCSP.004 Mult DCS nt *network* sg *server_group* dcs *dcs_id*

Long Syntax: SCSP.004 Multiple DCS IDs at a ATM address. Network *network* SGID *server_group* DCSID *dcs_id*

Description: A SCSP message was received from a DCS and the DCS ID does not match the previous DCS ID that we had from that ATM address. The message is discarded. The DCS_ID is the ID of the DCS that we already have at that ATM address.

Cause: configuration error

SCSP.005

Level: UI-ERROR

Short Syntax: SCSP.005 out of memory

Long Syntax: SCSP.005 An error occured when attempting to allocate memory

Description: An error occured when attempting to allocate memory. Memory is depleted.

Cause: overload

SCSP.006

Level: U-INFO

Short Syntax: SCSP.006 DCS not config. nt *network* sg *server_group* dcs *DCS_ID*

Long Syntax: SCSP.006 Message received from unconfigured DCS. nt *network* sg *server_group* dcs *DCS_ID*

Description: A message was received from a DCS that was not configured under this server group. The configuration indicates secure mode, so no DCS is automatically brought up.

SCSP.007

Level: U-TRACE

Short Syntax: SCSP.007 SCSP up on nt *network*

Long Syntax: SCSP.007 SCSP initialized on network *network*

Description: The indicated network has come up and SCSP has been initialized for this network.

SCSP.008

Level: U-TRACE

Short Syntax: SCSP.008 Add DCS nt *network* sg *server_group* atm *partial_atm_addr*

Long Syntax: SCSP.008 A DCS is added to network *network*, server group *server_group*, atm addr (esi,sel) *partial_atm_addr*

Description: A DCS was added to the given server group. The channel is not yet up, nor is there a DCSID yet.

SCSP.009

Level: U_INFO

Short Syntax: SCSP.009 Hello on down DCS, nt *network* sg *server_group*

Long Syntax: SCSP.009 Hello msg received on down DCS, network *network*, server_group *server_group*

Description: The only way this could happen is if we get an hello, but we have not yet received a channel_up for this channel from the API...really shouldn't happen either

---

SCSP.010

Level: U_TRACE

Short Syntax: SCSP.010 DCS Hello state chg to *dcs_hello_state*, nt *network* sg *server_group* dcs *dcsid*

Long Syntax: SCSP.010 DCS Hello FSM state change to *dcs_hello_state* on network *network*, server group *server_group*, DCSID *dcsid*

Description: The DCS Hello Finite State Mache has changed state. The states are: DOWN - channel is not up yet DOWN_INOP - channel has not yet been opened WAITING - waiting for hello msg from DCS UNIDIRECTIONAL - a hello message has been received, but it did not contain our LSID BIDIRECTIONAL - final state, exchanging hello

---

SCSP.011

Level: U_INFO

Short Syntax: SCSP.011 RID doesn't match LSID, nt *network* sg *server_group* dcs *dcsid*

Long Syntax: SCSP.011 RID in received msg doesn't match LSID of this DCS. network *network*, server group *server_group*, DCSID *dcsid*

Description: The Receiver ID in the message does not match the configured LSID of the given DCS.

SCSP.012

Level: P_TRACE

Short Syntax: SCSP.012 Hello rcvd nt *network* sg *server_group* dcs *dcsid*

Long Syntax: SCSP.012 Hello message received on network *network*, server group *server_group*, DCSID *dcsid*

Description: Normal hello message

SCSP.013

Level: U_TRACE

Short Syntax: SCSP.013 DCS CA state chg to *dcs_ca_state/ dcs_master_state*, nt *network* sg *server_group* dcs *dcsid*

Long Syntax: SCSP.013 DCS CA FSM state change to *dcs_ca_state/ dcs_master_state* on network *network*, server group *server_group*, DCSID *dcsid*

Description: The DCS Cache Alignment Finite State Machine has changed state. DOWN - hellos have not yet been exchanged. MS_NEG - negotiating master/slave SUMMARIZE - exchanging cache summarization records UPDATE - exchanging database records ALIGNED - is the final state.

SCSP.014

Level: UE_ERROR

Short Syntax: SCSP.014 CA msg rejected nt *network* sg *server_group* dcs *dcsid*

Long Syntax: SCSP.014 CA msg rejected nt *network* sg *server_group* dcs *dcsid*

Description: The received CA message was rejected for one of these reasons: In MS_NEG and we don't accept the M/S claim. In SUMMARIZE, UPDATE or ALIGNED and I flag is set or M flag is incorrect. In SUMMARIZE, UPDATE or ALIGNED and MASTER and the seq no is less than ours In SUMMARIZE/SLAVE and seq no is not one more than our last one. In UPDATE or ALIGNED and it's not a duplicate In DOWN state

---

SCSP.015

Level: P_TRACE

Short Syntax: SCSP.015 CA rcvd nt *network* sg *server_group* dcs *dcsid*

Long Syntax: SCSP.015 CA message received on network *network*, server group *server_group*, DCSID *dcsid*

Description: Cache Alignment message was received.

SCSP.016

Level: UE_ERROR

Short Syntax: SCSP.016 CSUS msg rejected nt *network* sg *server_group* dcs *dcsid*

Long Syntax: SCSP.016 CSUS msg rejected nt *network* sg *server_group* dcs *dcsid*

Description: The received CSUS message was rejected for one of these reasons: Not in UPDATE or ALIGNED state.

SCSP.017

Level: UE_ERROR

Short Syntax: SCSP.017 Bad message nt *network* sg *server_group* dcs *dcsid*

Long Syntax: SCSP.017 Bad message nt *network* sg *server_group* dcs *dcsid*

Description: The received message was rejected for one of these reasons: The packet was too short for the indicated length. The packet contained a CSA or CSAS record that was too short.

SCSP.018

Level: P_TRACE

Short Syntax: SCSP.018 CSA rcvd nt *network* sg *server_group* dcs *dcsid* cpa *protocol_addr csa_state*

Long Syntax: SCSP.018 Cache Update received nt *network* sg *server_group* dcs *dcsid* cpa *protocol_addr csa_state*

Description: A cache update was received from the given DCS. cpa is the protocol address.

SCSP.019

Level: U_TRACE

Short Syntax: SCSP.019 Hello missed, state chg to *dcs_hello_state*, nt *network* sg *server_group* dcs *dcsid*

Long Syntax: SCSP.019 Hello message missed, FSM state change to *dcs_hello_state* on network *network*, server group *server_group*, DCSID *dcsid*

Description: A hello message was not received from the DCS within the Hello Interval times the Dead Factor.

SCSP.020

Level: U_TRACE

Short Syntax: SCSP.020 CA missed, retransmitting, nt *network* sg *server_group* dcs *dcsid*

Long Syntax: SCSP.020 CA message missed, retransmitting. network *network*, server group *server_group*, DCSID *dcsid*

Description: A CA message was not received from the DCS within the expected period when in SUMMARIZE state.

SCSP.021

Level: U_TRACE

Short Syntax: SCSP.021 CSUS missed, retransmitting, nt *network* sg *server_group* dcs *dcsid*

Long Syntax: SCSP.021 CSUS message missed, retransmitting. network *network*, server group *server_group*, DCSID *dcsid*

Description: A CSUS message was not received from the DCS within the expected period when in the UPDATE state.

SCSP.022

Level: U_TRACE

Short Syntax: SCSP.022 retransmitting CSAs, nt *network* sg *server_group* dcs *dcsid*

Long Syntax: SCSP.022 retransmitting CSAs. network *network*, server group *server_group*, DCSID *dcsid*

Description: CSAs sent in a CSU_REQ message were not acknowledged. They are being retransmitted.

SCSP.024

Level: UE_ERROR

Short Syntax: SCSP.024 sg (*server_group*) bad in msg nt *network*

Long Syntax: SCSP.024 message contains a sg (*server_group*) that is not configured. network *network*

Description: A message or CSA was received for a server group that is not configured on this network.

SCSP.025

Level: UE_ERROR

Short Syntax: SCSP.025 bad msg type (*message_type*) nt *network*

Long Syntax: SCSP.025 bad message type (*message_type*) received on network *network*

Description: A message with an unrecognized message type was received.

SCSP.026

Level: UE_ERROR

Short Syntax: SCSP.026 bad msg hdr nt *network*

Long Syntax: SCSP.026 bad message header received on network *network*

Description: A bad message was received. Could be on of the following reasons: Bad message version. Bad checksum.

SCSP.027

Level: U_TRACE

Short Syntax: SCSP.027 EP state chg (*ep_state*) nt *network* SG *server_group*

Long Syntax: SCSP.027 Endpoint state change to *ep_state*, network *network*, SGID *server_group*

Description: The endpoint or ATM address used by this server group has changed state. 0 is inactive, 1 is active

---

SCSP.028

Level: P_TRACE

Short Syntax: SCSP.028 *message_type* msg sent nt *network* sg *server_group* sid *sender_id* rid *receiver_id*

Long Syntax: SCSP.028 *message_type* message sent on

network *network* server group *server_group*. sid *sender_id* rid *receiver_id*

Description: A message of the given type is being sent. sid is the Sender ID (LSID). rid is the Receiver ID (DCS ID)

SCSP.029

Level: P_TRACE

Short Syntax: SCSP.029 CSUS rcvd nt *network* sg *server_group* dcs *dcsid*

Long Syntax: SCSP.029 CSUS message received on network *network*, server group *server_group*, DCSID *dcsid*

Description: Cache State Update Solicit message was received.

SCSP.031

Level: P_TRACE

Short Syntax: SCSP.031 *message_type* msg rcvd nt *network* sg *server_group* sid *sender_id* rid *receiver_id*

Long Syntax: SCSP.031 *message_type* message received on network *network* server group *server_group*. sid *sender_id* rid *receiver_id*

Description: A message of the given type was received. sid is the Sender ID (LSID). rid is the Receiver ID (DCS ID)

SCSP.032

Level: U_TRACE

Short Syntax: SCSP.032 CSA retry exceeded, state chg to *dcs_hello_state*, nt *network* sg *server_group* dcs *dcsid*

Long Syntax: SCSP.032 CSA retry count exceeded, HFSM state change to *dcs_hello_state* on network *network*, server group *server_group*, DCSID *dcsid*

Description: A DCS did not acknowledge receipt of a CSA after several retries. This is considered an abnormal event. The DCS is reset to WAITING state.

SCSP.033

Level: C_TRACE

Short Syntax: SCSP.033 cache entry *cache_action*, paddr *protocol_address*, oid *origin_id*,, key *cache_key*

Long Syntax: SCSP.033 cache entry *cache_action*, protocol address *protocol_address*, origin ID *origin_id*,, cache key *cache_key*

Description: A SCSP cache element is created, updated, unlinked or aged. cache_action is one of: created - cache element is created updated - sequence number updated relinked - updated and relinked unlinked - server is removing cache entry ignored - update is ignored because sequence number is less aged - aged out

---

SCSP.034

Level: UE_ERROR

Short Syntax: SCSP.034 duplicate SG registration (*server_group*) nt *network*

Long Syntax: SCSP.034 duplicate server group registration (*server_group*) on network *network*

Description: A server has attempted to start a server group with a server group id that has already been used. Check the configuration for duplicate server group ids.

Chapter 105. Simple Network Management Protocol (SNMP)

This chapter describes Simple Network Management Protocol (SNMP) messages. For information on message content and how to use the message, refer to the Introduction.

SNMP.001

Level: P-TRACE

Short Syntax: SNMP.001 rcvd pkt frm hst *source_address*

Long Syntax: SNMP.001 received packet from host *source_address*

Description: This message is generated for each SNMP packet received from a remote host.

SNMP.002

Level: P-TRACE

Short Syntax: SNMP.002 snt pkt to hst *dest_address*

Long Syntax: SNMP.002 sent packet to host *dest_address*

Description: This message is generated for each SNMP packet sent to a remote host.

SNMP.003

Level: UE-ERROR

Short Syntax: SNMP.003 rcvd non-SNMP pkt frm hst *source_address* (err= *value*)

Long Syntax: SNMP.003 received non-SNMP packet from host *source_address* (error code = *value*)

Description: This message is generated by a first-level reasonableness check of an incoming SNMP packet. The error codes have the following meanings: 1 - packet does not begin with SEQUENCE (0x30) 2 - packet sequence length too small 3 - packet sequence length improperly encoded (in one byte) 4 - first packet field not an ASN.1 INTEGER 5 - packet sequence length improperly encoded (in two bytes) 6 - first packet field not an ASN.1 INTEGER 7 - some other error was detected

Cause: Another node on the network sent an improperly formed SNMP packet to the router.

Action: Examine the remote node, specified in the error message, for errors.

SNMP.004

Level: UE-ERROR

Short Syntax: SNMP.004 bad ver *version* frm hst *source_address*

Long Syntax: SNMP.004 bad version number *version* from host *source_address*

Description: This message indicates that an SNMP packet contained an incorrect version number.

Cause: Either the router or the Network Manager is running an incompatible version.

Action: Update (or back out) one version of SNMP.

Cause: A bad packet slipped through the first-level error checks.

Action: Check the network for wild packets.

SNMP.005

Level: U-TRACE

Short Syntax: SNMP.005 no access: comm *community*, hst *source_address*

Long Syntax: SNMP.005 no access to community *community* from host *source_address*

Description: This message indicates that an SNMP request from a remote host specified a community which does not exist or a community which did not list that host's IP address as acceptable.

Cause: The remote host is using the wrong community name.

Action: Update the remote hosts Network Manager.

Cause: The defined community in the router is incorrect.

Action: Correct the community name or add the remote host's IP address to the community's list.

SNMP.006

Level: UE-ERROR

Short Syntax: SNMP.006 bad appl type *appl_type* frm hst *source_address*

Long Syntax: SNMP.006 bad application type *appl_type* from host *source_address*

Description: This message indicates that an SNMP packet had a bad request type. That is, it was not a GET, GETNEXT or SET request.

Cause: The remote host is in error (perhaps sending response packets).

Action: Check the remote host.

Cause: A bad packet slipped through the first-level error checks.

Action: Check the network for wild packets.

SNMP.007

Level: UI-ERROR

Level: OOM

Short Syntax: SNMP.007 no free pkr bfr

Long Syntax: SNMP.007 no packet buffer available

Description: This message is generated when SNMP cannot allocate a packet in which to construct an SNMP response.

Cause: All available free memory is currently in use on the system.

Action: Retry query at a later time. If unsuccesful, a memory upgrade may be required. Monitor memory statistics to determine usage.

SNMP.008

Level: U-TRACE

Short Syntax: SNMP.008 R/O access for SET: hst *source_address*, comm *community*

Long Syntax: SNMP.008 SET request from host *source_address* has read-only access on community *community*

Description: This message indicates that a SET request came in on a community which only provides read-only access to the MIB.

Action: Provide a community which allows sets or get the remote host to stop sending SET requests.

SNMP.009

Level: UI-ERROR

Level: OOM

Short Syntax: SNMP.009 TRAP: no free pkt bfr

Long Syntax: SNMP.009 TRAP: no free packer buffer available

Description: This message is generated whenever SNMP cannot send a trap because it cannot allocate storage.

Cause: All available free memory is currently in use on the system.

Action: A memory upgrade may be required. Monitor memory statistics to determine usage.

SNMP.012

Level: C-TRACE

Short Syntax: SNMP.012 comm *name* added

Long Syntax: SNMP.012 community *name* added

Description: This message is generated by the SNMP configuration routine when it reads a new community in from SRAM.

SNMP.013

Level: UE-ERROR

Short Syntax: SNMP.013 rcvd non-SNMP pkt frm hst *source_address*

Long Syntax: SNMP.013 received non-SNMP packet from host *source_address*

Description: This message is generated by a reasonableness check of an incoming SNMP packet. This check is done just before processing the PDU.

Cause: Another node on the network sent an improperly formed SNMP packet to the router, and the packet slipped through the first level error checks.

Action: Examine the remote node, specified in the error message, for errors.

SNMP.014

Level: UE-ERROR

Short Syntax: SNMP.014 bad ovarlen *source_address* frm hst *ovarlen*

Long Syntax: SNMP.014 length of variable to be sent out exceeds max length *source_address* from host *ovarlen*

Description: This message is generated by a length check on the variable to be sent out.

Cause: Another node on the network sent an improperly formed SNMP packet to the router, and the packet slipped through the first level error checks.

Action: Examine the remote node, specified in the error message, for errors.

SNMP.015

Level: P-TRACE

Short Syntax: SNMP.015 rcvd get-req pkt frm hst *source_address*

Long Syntax: SNMP.015 received a get-request packet from host *source_address*

Description: This message is generated for each SNMP packet received from a remote host of the type get-request.

SNMP.016

Level: P-TRACE

Short Syntax: SNMP.016 rcvd get-nxt pkt frm hst *source_address*

Long Syntax: SNMP.016 received a get-next packet from host *source_address*

Description: This message is generated for each SNMP packet received from a remote host of the type get-next.

SNMP.017

Level: P-TRACE

Short Syntax: SNMP.017 rcvd set-req pkt frm hst *source_address*

Long Syntax: SNMP.017 received a set-request packet from host *source_address*

Description: This message is generated for each SNMP packet received from a remote host of the type set-request.

SNMP.018

Level: U-TRACE

Short Syntax: SNMP.018 pkt frm hst : *source_address* caused err typ toobig

Long Syntax: SNMP.018 packet from host *source_address* resulted in a pkt with error status: toobig

Description: This message indicates that a packet was sent out with the error status as too big as a result of the SNMP variable in question not fitting the packet size.

Action: Increase the packet-size.

SNMP.019

Level: U-TRACE

Short Syntax: SNMP.019 pkt frm hst : *source_address* caused err typ nosuchnam

Long Syntax: SNMP.019 packet from host *source_address* resulted in a pkt with error status: nosuchname

Description: This message indicates that a packet was sent out with the error status as noSuchName as a result of the SNMP variable in question not existing in the system or not in the view associated with the specified community or the operation is a set on a read-only variable.

Action: Ensure that the requested variable exists in the system (also possibly the particular instance of the variable), it is in the requested view, the community has the correct access type and the requested variable is writable if it is a set operation.

SNMP.020

Level: U-TRACE

Short Syntax: SNMP.020 pkt frm hst : *source_address* caused err typ badvalue

Long Syntax: SNMP.020 packet from host *source_address* resulted in a pkt with error status: badvalue

Description: This message indicates that a packet was sent out with the error status as badvalue as a result of trying to set a variable with a wrong value specified in the SET request.

Action: Ensure that the SET request from the remote host specifies a value consistent with the ASN1 type of the value that it is attempting to set.

SNMP.021

Level: UE-ERROR

Short Syntax: SNMP.021 Pkt discd, inp buffs low, net *Network ID*

Long Syntax: SNMP.021 Packet Discarded, input buffers are low, network *Network ID*

Description: The input buffer pool of the incoming SNMP packet, fell below the low watermark. The router dropped the SNMP packet to try and free up buffer space for other traffic.

Cause: A burst of traffic has overflowed the input buffers on an interface.

Action: If this occurs regularly then the input buffers of the interface indicated in the message may have to be increased.

SNMP.022

Level: UE-ERROR

Short Syntax: SNMP.022 ext err (*tag*) at *file*(*line*) : *message*

Long Syntax: SNMP.022 code encountered external error (*tag*) at *file*(*line*) : *message*

Description: SNMP code encountered error situation caused by an external event.

Action: Take proper action according to the error message.

SNMP.023

Level: UI-ERROR

Short Syntax: SNMP.023 int err (*tag*) at *file*(*line*) : *message*

Long Syntax: SNMP.023 code encountered internal error (*tag*) at *file*(*line*) : *message*

Description: SNMP code encountered error situation caused by an internal event.

Action: Take proper action according to the error message.

SNMP.024

Level: C-TRACE

Short Syntax: SNMP.024 generic trc (*tag*) at *file*(*line*) : *message*

Long Syntax: SNMP.024 generic trace message (*tag*) at *file*(*line*) : *message*

Description: SNMP code generates the message. Trace messages are categoried into P1 (the most significant), P2, P3, and P4.

Action: Take proper action according to the trace message.

SNMP.025

Level: C-TRACE

Short Syntax: SNMP.025 trc sgmt: *trace_segment*

Long Syntax: SNMP.025 trace segment is generated *trace_segment*

Description: SNMP code generates the trace segment.

Action: Take proper action according to the trace segments.

SNMP.026

Level: C-TRACE

Short Syntax: SNMP.026 pkt trc (*tag*) at *file*(*line*) : *message*

Long Syntax: SNMP.026 snmp packet trace (*tag*) at *file*(*line*) : *message*

Description: Indicates that snmp_packet is traced. Need to turn on SNMP_25 to get the rest of packet information.

SNMP.027

Level: C-TRACE

Short Syntax: SNMP.027 snmp container (*tag*) at *file*(*line*)

Long Syntax: SNMP.027 snmp main data structure trace (*tag*) at *file*(*line*)

Description: Indicates that snmp major data structure is traced. Need to turn on SNMP_25 to get the rest of packet information.

SNMP.028

Level: CI-ERROR

Short Syntax: SNMP.028 err (*tag*) at *file*(*line*) : *message*

Long Syntax: SNMP.028 code encountered error (*tag*) at *file*(*line*) : *message*

Description: SNMP code encountered error situation. Messages are categoried into E1 (the most severe), E2 and E3 levels.

Action: Take proper action according to the error message.

Panic nmnostor

Short Syntax: SNMP: no storage for MIB

Description: No storage was available to add an entry to the MIB.

Panic nmitype

Short Syntax: SNMP: interface type not defined for net

Description: The structure that defines an interface does not define a value for the MIB-II ifType variable.

Action: Contact customer service for a new load. Do not try and enable SNMP with this load.

Panic snmpudperr

Short Syntax: snmp udp port not avail

Description: Another application registered previously with snmp's UDP port.

Action: Contact customer service.

Chapter 106. Source Routing Transparent (SRT) Bridge

This chapter describes Source Routing Transparent (SRT) Bridge messages. For information on message content and how to use the message, refer to the Introduction.

SRT.001

Level: UI-ERROR

Short Syntax: SRT.001 No buf to dup broadcast frame *source_mac-> dest_mac* to port *port*, nt *network*

Long Syntax: SRT.001 No buffer available to duplicate frame from *source_mac* to *dest_mac* on to port *port*, network *network*

Description: No buffer available to copy a frame in order to send a bridged frame on multiple interfaces. Bridged packets are sent on multiple interfaces either for multicast destination addresses, or in the case of certain static entries. No copy of this frame will be sent on the specified port and network.

Cause: Severe packet buffer shortage.

Action: Check memory statistics in GWCON to verify packet buffer level.

Cause: Traffic peak using all available buffers.

Action: This is the problem if this message occurs infrequently.

SRT.002

Level: UI-ERROR

Short Syntax: SRT.002 Err *error_code* setting promsic mode on nt *network*

Long Syntax: SRT.002 Error code *error_code* trying to set promiscuous mode on network *network*

Description: The Spanning Tree Protocol requested setting this network into Learning state, but the command to the device failed. The error_code is a device-specific error code that may indicate what the error is.

Cause: Hardware failure or software bug.

Action: Contact customer service.

SRT.003

Level: UI-ERROR

Short Syntax: SRT.003 Hw cache full on port *port* nt *network*

Long Syntax: SRT.003 Hardware cache full on port *port* network *network*

Description: A hardware cache used for internal filtering and learning detected a full condition while attempting to age entries.

Cause: Too many entries in hardware cache.

Action: Reduce resolution time period.

SRT.004

Level: UI-ERROR

Short Syntax: SRT.004 No buf for *command_name* cmd to nt *network*

Long Syntax: SRT.004 No buffer available for *command_name* command to network *network*

Description: No buffer was available to send a command to the device. The possible command names are ″D_CNFGSRB″ (configure source-routing bridging), ″SRT_ON″ (promiscuous on), ″SRT_INFORM″ (learn capabilities of device), ″SRT_SET_AGE″ (set age for filtering database in device), ″SRT_DECR_AGE″ (do aging pass on filtering database in device), ″SRT_ADD_ENTRY″ (add static entry), ″SRT_DEL_ENTRY″ (delete entry, from console), ″SRT_SEARCH_ENTRY″ (search for particular entry, from console), and ″SRT_LIST_ENTRY″ (list contents of learning database in card). For commands ″D_CNFGSRB″ and ″SRT_ON″ the result will be that the interface may remain in the wrong state. A failure on ″SRT_INFORM″ could cause serious problems. For other commands the results will be less serious.

Cause: Severe packet buffer shortage.

Action: Check memory statistics in GWCON to verify packet buffer level.

Cause: Traffic peak using all available buffers.

Action: This is the problem if this message occurs infrequently.

SRT.005

Level: UI-ERROR

Short Syntax: SRT.005 *source_mac-> dest_mac* send fld, rsn *reason_code*, port *port* nt *network*

Long Syntax: SRT.005 Sending Frame from *source_mac* to *dest_mac* failed, reason *reason_code*, on port *port* network *network*

Description: The sending of a packet being forwarded

failed. The reason is the internal error code for the failure.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

SRT.006

Level: CI-ERROR

Short Syntax: SRT.006 Input q ovf *source_mac*-> *dest_mac*, dropped, nt *network*

Long Syntax: SRT.006 Input queue overflow on frame from *source_mac* to *dest_mac*, packet dropped from network *network*

Description: The input queue for frames to be forwarded is too long, and this frame has been dropped to attempt to alleviate the congestion.

Cause: Bursty traffic may be causing congestion.

Action: Wait for burst to subside.

Cause: Too much traffic for forwarder to forward.

Action: Reconfigure network. Increase speed of router.

Cause: Inadequate buffer resources.

Action: Examine memory statistics in GWCON.

SRT.008

Level: CE-ERROR

Short Syntax: SRT.008 *source_mac*-> *dest_mac* too big (*reformatted_length* > *output_maximum*) for port *port* nt *network*, dropped

Long Syntax: SRT.008 Frame from *source_mac* to *dest_mac* is too big (reformatted length *reformatted_length* bytes > output maximum size *output_maximum* bytes) for port *port* network *network*, dropped

Description: The specified frame is too large to send on this outgoing port and network. The reformatted_length is the size of the frame including MAC headers after any mapping of data link headers.

Cause: Host on network with large maximum frame size sending to host on network with smaller maximum frame size.

Action: Reconfigure sending host to not send such large frames. If frame is of a routable protocol supporting fragmentation (such as IP or ISO) or maximum frame size determination (DNA or XNS), convert to using routing instead of bridging.

Cause: Host on network with large maximum frame size sending to host via an intervening network with smaller maximum frame size.

Action: Reconfigure network to use networks with large maximum frame size (such as FDDI or 802.5) as the backbone networks. Reconfigure port costs in Spanning Tree Protocol to favor spanning trees via networks with large maximum frame sizes.

SRT.009

Level: UE-ERROR

Short Syntax: SRT.009 *source_mac*-> *dest_mac* drp, nt *network* down

Long Syntax: SRT.009 Frame from *source_mac* to *dest_mac* dropped, input network *network* is down

Description: A frame has been received for bridging on a network that is down. It will be ignored.

Cause: A BPDU has been sent to the unicast address of the router on this interface.

Action: Correct action of sending node.

Cause: Internal state inconsistency.

SRT.010

Level: P-TRACE

Short Syntax: SRT.010 *source_mac*-> *dest_mac* drp, src add flt, port *port* nt *network*

Long Syntax: SRT.010 Frame from *source_mac* to *dest_mac* dropped, source address filtered, port *port* network *network*

Description: A MAC frame has been received by the hardware, but is being dropped because the source MAC address is being administratively filtered by the bridge. The frame will be dropped.

Cause: Receipt of frame whose source MAC address matches the source filter.

SRT.011

Level: U-TRACE

Short Syntax: SRT.011 *source_mac*-> *dest_mac* dropped, input port *port* nt *network* not forwarding

Long Syntax: SRT.011 Frame from *source_mac* to

dest_mac dropped, input port *port* network *network* not in forwarding state

Description: A MAC frame was received on a port that is still only in ″learning″ state. Frames are only bridged when the input port is in ″forwarding″ state. While the port is still in ″learning″ state, they are only processed to learn the source addresses for the filtering database. The frame will not be bridged.

Cause: Normal part of transition to ″forwarding″ state.

SRT.012

Level: U-INFO

Short Syntax: SRT.012 *source_mac*-> *dest_mac* dropped, output port *port* nt *network* not forwarding

Long Syntax: SRT.012 Frame from *source_mac* to *dest_mac* dropped, output port *port* network *network* not in forwarding state

Description: A MAC frame was being bridged, but the destination port was not in ″forwarding″ state. It will not be sent on that port.

Cause: Output port still in ″learning″ state.

Action: None needed, port will transition to ″forwarding″.

Cause: Static entry in filtering database points to port that is not in ″forwarding″ state.

SRT.013

Level: P-TRACE

Short Syntax: SRT.013 *source_mac*-> *dest_mac* drp, dst same LAN, port *port* nt *network*

Long Syntax: SRT.013 Frame from *source_mac* to *dest_mac* dropped, destination on same LAN, port *port* network *network*

Description: A MAC frame has been received whose destination address is known to be on the same side of the bridge as the packet came from. It is dropped by the filtering logic since it does not need to be bridged.

Cause: Normal local traffic on network.

SRT.014

Level: CI-ERROR

Short Syntax: SRT.014 *source_mac*-> *dest_mac* drp, dst port *port* not enabled, nt *network*

Long Syntax: SRT.014 Frame from *source_mac* to *dest_mac* dropped, destination port *port* not enabled, network *network*

Description: A frame being bridged was destined for a port which is not running transparent bridging, or not in ″forwarding″ state for transparent bridging.

Cause: Static entry in filtering database points to port that is not in ″forwarding″ state.

SRT.015

Level: P-TRACE

Short Syntax: SRT.015 *source_mac*-> *dest_mac* brdg port *port* nt *network* to port *port* nt *network*

Long Syntax: SRT.015 Frame from *source_mac* to *dest_mac* bridged from port number *port* network *network* to port number *port* network *network*

Description: A frame is being bridged between these two interfaces. The destination address was known, so it was sent only to the correct destination network.

SRT.016

Level: P-TRACE

Short Syntax: SRT.016 *source_mac*-> *dest_mac* brdg-all port *port* nt *network* to port *port* nt *network*

Long Syntax: SRT.016 Frame from *source_mac* to *dest_mac* bridged to all ports from port number *port* network *network* to port number *port* network *network*

Description: A frame is being transparently bridged to all active transparent bridging ports. This happens when the frame destination is a multicast, when the frame destination is not in the learning database, or when required by static entries in the learning database. There will be one message for each port the frame is sent on.

SRT.017

Level: U-INFO

Short Syntax: SRT.017 Enabling SRT on port *port* nt *network*

Long Syntax: SRT.017 Enabling SRT on port *port* network *network*

Description: The SRT forwarder is starting the process of enabling bridging on the specified interface. This starts when the interface comes up from a self-test.

SRT.018

Level: C-INFO

Short Syntax: SRT.018 SRT startup complete on port *port* nt *network*

Long Syntax: SRT.018 SRT startup complete on port *port* network *network*

Description: The SRT forwarder has completed the process of enabling bridging on the specified interface. It will now enter ″blocking″ state.

SRT.019

Level: UI-ERROR

Short Syntax: SRT.019 Unsupp ifc typ *type_name*, nt *network*

Long Syntax: SRT.019 Unsupported interface type *type_name*, network *network*

Description: The SRT forwarder had been enabled on a type of interface it does not support.

Cause: Enabling SRT on an interface which does not support SRT, such as ProNET-10.

SRT.020

Level: UI-ERROR

Short Syntax: SRT.020 Can't autocfg brdg addr, lowest port *port* nt *network* no MAC addr

Long Syntax: SRT.020 Cannot autoconfigure the bridge address, the lowest numbered port *port* network *network* has no MAC address

Description: The user has configured the bridge to autoconfigure the bridge address based on the MAC address of the lowest number port. However, the lowest numbered port is of a type that does not have a MAC address, such as a serial line.

Action: Assign address to bridge by using SRT config> command "SET BRIDGE".

SRT.022

Level: UI-ERROR

Short Syntax: SRT.022 Bridge config with no valid ports, disabling

Long Syntax: SRT.022 Bridge configured with no valid ports, disabling the bridge

Description: The bridge has been enabled, but there are no ports configured on that bridge or there was a mismatch between the interface and bridge port configuration records. The bridge will be left disabled.

Action: Resolve configuration conflicts between bridge ports and devices.

SRT.023

Level: UI-ERROR

Short Syntax: SRT.023 Port *port* config on nonexist network number *network_number*

Long Syntax: SRT.023 Port *port* configured on nonexistent network number *network_number*

Description: The port has been configured to use a network that has not been configured with the Config> ADD DEVICE command. This port of the bridge will be disabled.

Cause: Inconsistency between router device configuration and bridge configuration.

Action: Correct the network number in the bridge configuration, or add the network in the device configuration.

SRT.024

Level: UI-ERROR

Short Syntax: SRT.024 *existent_port_count* ports is < 2, disabling

Long Syntax: SRT.024 *existent_port_count* existent ports is less than 2, disabling bridge

Description: Less than two (valid) ports have been configured on the bridge. There must be at least two ports.

Cause: Less than two ports configured.

Action: Add more ports, or don't try and use bridging.

Cause: Too many ports on non-configured devices.

Action: Resolve configuration conflicts between bridging ports and devices.

SRT.025

Level: UI-ERROR

Short Syntax: SRT.025 No mem for filt db (req *requested_size*, min *minimum_size*), disabl

Long Syntax: SRT.025 No memory for filtering databse (desired size *requested_size* bytes, absolute minimum size *minimum_size* bytes), disabling bridge

Description: There is not enough free memory to allocate even a minimal size filtering database. The bridge will be disabled. The bridge starts by trying to allocate requestd_size bytes, and then tries with progressively smaller sizes down to minimum_size. The minimum size is enough only for the registered and static entries.

Cause: Severe shortage of memory.

Action: Reduce routing table sizes in other protocols, use system with less protocols, expand memory in router.

SRT.026

Level: C-INFO

Short Syntax: SRT.026 *source_mac*== *dest_mac*, drop, port *port* nt *network*

Long Syntax: SRT.026 Frame from *source_mac* to *dest_mac*, source same as destination, dropping, from port *port* network *network*

Description: Frames to and from the same address are not bridged by this bridge.

SRT.027

Level: P-TRACE

Short Syntax: SRT.027 Chg state *old_state* to *new_state*, port *port* nt *network*

Long Syntax: SRT.027 Changing port state from *old_state* to *new_state* for port *port*, network *network*

Description: The Spanning Tree Protocol has requested this state change for this port in the SRT bridge. The old_state and new_state are one of: FORWARDING (Spanning Tree Protocol Forwarding state), LEARNING (Spanning Tree Protocol Learning state), LISTENING (Spanning Tree Protocol Listening state), BLOCKED (Spanning Tree Protocol Blocking state), CONFIGURING (configuration of port device pending), POSTCONFIGURING (configuration of port device done), PRECONFIGURING (port enabled, configuration of port device to start), and DISABLED (port disabled).

SRT.028

Level: UI_ERROR

Short Syntax: SRT.028 No room for PERM *mac_address* in filt database, disabling

Long Syntax: SRT.028 No room for permanent address *mac_address* in filtering database, disabling bridge

Description: There is no room for the permanent entry in the filtering database. The bridge will be disabled.

Cause: Filtering database size too small.

Action: Make filtering database larger.

Cause: Too many permanent entries.

Action: Configure less permanent entries.

SRT.029

Level: UI_ERROR

Short Syntax: SRT.029 No mem for PERM *mac_address*, disabling

Long Syntax: SRT.029 No memory for permanent address *mac_address*, disabling bridge

Description: There is no room for the permanent entry in an auxiliary database. The bridge will be disabled.

Cause: Too little free memory.

Action: Make routing databases smaller.

Action: Increase memory size.

Cause: Too many permanent entries.

Action: Configure less permanent entries.

SRT.030

Level: UI-ERROR

Short Syntax: SRT.030 *command* Cmd fld to net *network*

Long Syntax: SRT.030 *command* command failed to network *network*

Description: A command to a network device failed. The possible command names are ″SRT_ON″ (promiscuous on), ″SRT_OFF″ (promiscuous off), ″SRT_INFORM″ (learn capabilities of device), ″SRT_ADD_ENTRY″ (add static entry in device), ″SRT_SET_AGE″ (set age for filtering database in device), and ″SRT_DECR_AGE″ (do aging pass on filtering database in device). For commands ″SRT_ON″ and ″SRT_OFF″ the result will be that the interface may remain in the wrong state. A failure on ″SRT_INFORM″ could cause serious problems. For other commands the results will be less serious.

Cause: Hardware failure or software bug.

Action: Contact customer service.

SRT.031

Level: UI-ERROR

Short Syntax: SRT.031 No buf to dup *routing_type* frame *source_mac-> dest_mac* to port *port*, nt *network*

Long Syntax: SRT.031 No buffer available to duplicate *routing_type* frame from *source_mac* to *dest_mac* on to port *port*, network *network*

Description: No buffer available to copy a frame in order to send an All Routes Explorer (ARE) or Spanning Tree Explorer (STE) routing_type frame on multiple interfaces. ARE frames are sent on all interfaces which are part of the SRT spanning tree, STE frames are sent on all interfaces running source-routing. No copy of this frame will be sent on the specified port and network.

Cause: Severe packet buffer shortage.

Action: Check memory statistics in GWCON to verify packet buffer level.

Cause: Traffic peak using all available buffers.

Action: This is the problem if this message occurs infrequently.

SRT.032

Level: UI-ERROR

Short Syntax: SRT.032 SR *source_mac-> dest_mac* send fld, rsn *reason_code*, port *port* nt *network*

Long Syntax: SRT.032 Sending source routed frame from *source_mac* to *dest_mac* failed, reason *reason_code*,

on port *port* network *network*

Description: The sending of a source routed frame being forwarded failed. The reason_code is the internal error code for the failure.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

SRT.033

Level: C-TRACE

Short Syntax: SRT.033 *routing_type* dup RD drop *source_mac*-> *dest_mac* from port *port*, nt *network*

Long Syntax: SRT.033 *routing_type* with duplicate Route Descriptor from *source_mac* to *dest_mac* from port *port*, network *network*

Description: A source-routed frame having a All Routes Descriptor (ARE) or Spanning Tree Explorer (STE) routing_type in the RIF has a duplicate Routing Descriptor in the RIF. The frame will be dropped. This is a normal occurrence for ARE frames when there are any duplicate paths in the source routing domain. For STE frames, this indicates that there is an interface that is part of the source-routing spanning tree that should not be.

Cause: Receiving an ARE/STE from a segment is has already been on.

Action: None needed for ARE, this is normal. For STE, one may want to correct it's "spanning tree," but this is not essential.

SRT.034

Level: UE-ERROR

Short Syntax: SRT.034 SRF dup LOUT (RIF *RIF*) drop *source_mac*-> *dest_mac* from port *port*, nt *network*

Long Syntax: SRT.034 SRF with duplicate LOUT (RIF *RIF*) from *source_mac* to *dest_mac* from port *port*, network *network*

Description: A source-routed frame of Specifically-routed frame (SRF) type has a duplicate LOUT (outgoing LAN ID). This is illegal, and the frame will be dropped.

Cause: Station sending frame with invalid RIF that would go through the same bridge more than once, thus looping forever.

Action: Find out why station is using this RIF. Either it is using a hand-configured one that is wrong, or there is a bug in the discovery algorithm.

SRT.036

Level: UE-ERROR

Short Syntax: SRT.036 STE max RD (RIF *RIF*) drop *source_mac*-> *dest_mac* from port *port*, nt *network*

Long Syntax: SRT.036 Spanning Tree Explorer exceeds maximum Route Descriptors (RIF *RIF*) from *source_mac* to *dest_mac* from port *port*, network *network*

Description: A Spanning Tree Explorer (STE) source-routed frame has more Route Descriptors than this bridge is configured to allow for STE frames. The frame will be dropped.

Cause: Upstream bridge has an STE RD limit inconsistent with this bridge.

Action: Reconfigure all bridges in source-routing domain to have consistent STE RD limit.

Cause: Network has too many hops for configured STE RD limit.

Action: Reconfigure all bridges in source-routing domain to have STE RD limit consistent with the diameter of the domain.

SRT.037

Level: CE-ERROR

Short Syntax: SRT.037 SRF unk LOUT (RIF *RIF*) drop *source_mac*-> *dest_mac* from port *port*, nt *network*

Long Syntax: SRT.037 SRF with unknown LOUT (RIF *RIF*) from *source_mac* to *dest_mac* from port *port*, network *network*

Description: A source-routed frame of Specifically-routed frame (SRF) type has an outgoing LOUT (LAN ID Out) that does not match that of any active source-routing interface in the router. It will be dropped.

Cause: End station using RIF that was discovered before an interface went down in the router.

Action: None should be needed, the session on the station will fail, and it will re-initiate route discovery.

Cause: More than one bridge on the incoming segment with the same bridge number, and this LOUT matches in it.

Action: Reconfigure for legal configuration. All Bridge

Numbers must be unique on a given segment.

Cause: End station using completely invalid RIF.

Action: Find out why station is using this RIF.

SRT.038

Level: P-TRACE

Short Syntax: SRT.038 ARE rcv (RIF *RIF*) *source_mac*-> *dest_mac* from port *port*, nt *network*

Long Syntax: SRT.038 All Routes Explorer received (RIF *RIF*) from *source_mac* to *dest_mac* from port *port*, network *network*

Description: An All Routes Explorer frame has been received on the specified port.

SRT.039

Level: P-TRACE

Short Syntax: SRT.039 ARE sent (RIF *RIF*) *source_mac*-> *dest_mac* to port *port*, nt *network*

Long Syntax: SRT.039 All Routes Explorer sent (RIF *RIF*) from *source_mac* to *dest_mac* to port *port*, network *network*

Description: An All Routes Explorer frame has been sent on the specified port.

SRT.040

Level: P-TRACE

Short Syntax: SRT.040 STE rcv (RIF *RIF*) *source_mac*-> *dest_mac* from port *port*, nt *network*

Long Syntax: SRT.040 Spanning Tree Explorer received (RIF *RIF*) from *source_mac* to *dest_mac* from port *port*, network *network*

Description: A Spanning Tree Explorer frame has been received on the specified port.

SRT.041

Level: P-TRACE

Short Syntax: SRT.041 STE sent (RIF *RIF*) *source_mac*-> *dest_mac* to port *port*, nt *network*

Long Syntax: SRT.041 Spanning Tree Explorer sent (RIF *RIF*) from *source_mac* to *dest_mac* to port *port*, network *network*

Description: A Spanning Tree Explorer frame has been sent on the specified port.

SRT.042

Level: U-INFO

Short Syntax: SRT.042 *routing_type* LF lowered (*old_LF* to *new_LF*) *source_mac*-> *dest_mac* from port *port*, nt *network*

Long Syntax: SRT.042 *routing_type* Largest Frame size lowered (from *old_LF* bytes to *new_LF* bytes) from *source_mac* to *dest_mac* from port *port*, network *network*

Description: A source-routing explorer (ARE or STE in routing_type) has had the Largest Frame (LF) field lowered in its RIF. This happens whenever a frame is received from a segment with a smaller maximum frame size than the one presently encoded in the LF bits. This is a normal part of the spanning tree protocol to determine the maximum frame size on all routes.

Cause: It is somewhat abnormal to see this happen on received frames, and indicates that the endnodes or other bridges on this segment have different frame sizes configured. However, it is a perfectly legal configuration.

Action: Make frame size configurations consistent on a given segment.

SRT.043

Level: C-INFO

Short Syntax: SRT.043 *routing_type* LF lowered (*old_LF* to *new_LF*) *source_mac*-> *dest_mac* to port *port*, nt *network*

Long Syntax: SRT.043 *routing_type* Largest Frame size lowered (from *old_LF* bytes to *new_LF* bytes) from *source_mac* to *dest_mac* to port *port*, network *network*

Description: A source-routing explorer (ARE or STE in routing_type) has had the Largest Frame (LF) field lowered in its RIF. This happens whenever a frame is sent to a segment with a smaller maximum frame size than the one presently encoded in the LF bits. This is a normal part of the spanning tree protocol to determine the maximum frame size on all routes.

SRT.044

Level: P-TRACE

Short Syntax: SRT.044 SRF rcv (RIF *RIF*) *source_mac*-> *dest_mac* from port *port*, nt *network*

Long Syntax: SRT.044 Specifically-routed frame received (RIF *RIF*) from *source_mac* to *dest_mac* from port *port*, network *network*

Description: A Specifically-routed frame has been received on the specified port.

SRT.045

Level: P-TRACE

Short Syntax: SRT.045 Send SRF (RIF *RIF*) *source_mac*-> *dest_mac* to port *port*, nt *network*

Long Syntax: SRT.045 Sending Specifically-routed frame (RIF *RIF*) from *source_mac* to *dest_mac* to port *port*, network *network*

Description: A Specifically-routed frame is being sent on the specified port.

SRT.046

Level: UI-ERROR

Short Syntax: SRT.046 *routing_type* rcv *source_mac*-> *dest_mac* from disabl port *port*, nt *network*, disc

Long Syntax: SRT.046 *routing_type* frame received from *source_mac* to *dest_mac* on disabled port *port*, network *network*, discarded

Description: A source-routed frame has been received on the specified port, but that port is not configured for bridging. The routing_type is one of SRF (Specifically-routed frame), STE (Spanning Tree Explorer), or ARE (All Routes Explorer). This really should not happen on more than a transient basis, because ports that are not enabled for bridging should not be queueing packets to the source-routing forwarder.

SRT.047

Level: UI-ERROR

Short Syntax: SRT.047 *routing_type* rcv *source_mac*-> *dest_mac* from non-SR port *port*, nt *network*, disc

Long Syntax: SRT.047 *routing_type* frame received from *source_mac* to *dest_mac* on non-source-routing port *port*, network *network*, discarded

Description: A source-routed frame has been received on the specified port, but that port is not configured for source-routing bridging. The routing_type is one of SRF (Specifically-routed frame), STE (Spanning Tree Explorer), or ARE (All Routes Explorer). This really should not happen on more than a transient basis, because ports that are not enabled for bridging should not be queueing packets to the source-routing forwarder.

SRT.048

Level: P-TRACE

Short Syntax: SRT.048 STE dropped (RIF *RIF*) *source_mac*-> *dest_mac* from blk port *port*, nt *network*

Long Syntax: SRT.048 Spanning Tree Explorer dropped (RIF *RIF*) from *source_mac* to *dest_mac* from blocked port *port*, network *network*

Description: A Spanning Tree Explorer (STE) frame was dropped, and not forwarded, because the incoming port is not part of the spanning tree, or has been configured not to forward STE frames.

Cause: Normal for STE frames, this is the difference between them and ARE frames.

SRT.050

Level: UI-ERROR

Short Syntax: SRT.050 err *error_string* ena SR on nt *network*

Long Syntax: SRT.050 Got *error_string* error trying to enable source-routing on network *network*

Description: The bridge tried to enable source-routing bridging on this interface, but the interface refused the configuration command. Source-routing will be left disabled on this interface.

Cause: Either bad commands were passed to the interface, or there is a bug in the interface firmware.

Action: Contact customer service.

SRT.051

Level: UE-ERROR

Short Syntax: SRT.051 SRF *source_mac*-> *dest_mac* too big (*reformatted_length* > *output_maximum*) for port *port* nt *network*, dropped

Long Syntax: SRT.051 Specifically-routed frame from *source_mac* to *dest_mac* is too big (reformatted length *reformatted_length* > output maximum size *output_maximum*) for port *port* network *network*, dropped

Description: The specified Specifically-routed (source-routed) frame is too large to send on this outgoing port and network. The reformatted_length is the size of the frame including MAC headers after any mapping of data link headers.

Cause: Host not honoring LF bit values from its returned explorer frames.

Action: Fix host.

SRT.052

Level: UE-ERROR

Short Syntax: SRT.052 *routing_type source_mac*-> *dest_mac* too big (*reformatted_length* > *output_maximum*) for port *port* nt *network*, dropped

Long Syntax: SRT.052 *routing_type* frame from *source_mac* to *dest_mac* is too big (reformatted length *reformatted_length* > output maximum size *output_maximum*) for port *port* network *network*, dropped

Description: The source-routed explorer (ARE or STE routing_type) frame is too large to send on this outgoing port and network. The reformatted_length is the size of the frame including MAC headers after any mapping of data link headers.

Cause: The sending host is putting too much data in its explorer frames. These should normally be short, since it should not be making any assumptions about the maximum frame size available.

Action: Correct behavior of sending host.

SRT.053

Level: UI-ERROR

Short Syntax: SRT.053 *routing_type* inv RIF len *RIF_length*, *source_mac*-> *dest_mac* port *port*, nt *network*, disc

Long Syntax: SRT.053 *routing_type* with invalid RIF lenth *RIF_length* from *source_mac* to *dest_mac* from port *port*, network *network*, discarded

Description: A source-routing frame was received with an invalid RIF length encoded in the Length bits of the RIF. The routing_type is one of SRF (Specifically-routed frame), STE (Spanning Tree Explorer), or ARE (All Routes Explorer).

Cause: Received frame with RIF length less than 2 or not a multiple of 2 in length.

Action: Correct software in sending node.

SRT.054

Level: UI-ERROR

Short Syntax: SRT.054 No mem for hash tab (req *requested_size*), disabl

Long Syntax: SRT.054 No memory for hash table (desired size *requested_size* bytes), disabling bridge

Description: There is not enough free memory to allocate the hash table for the filtering database. The bridge will be disabled.

Cause: Severe shortage of memory.

Action: Reduce routing table sizes in other protocols, use system with less protocols, expand memory in router.

SRT.055

Level: UI-ERROR

Short Syntax: SRT.055 No mem for conv hash tab (req *requested_size*), disabl

Long Syntax: SRT.055 No memory for conversion hash table (desired size *requested_size* bytes), disabling bridge

Description: There is not enough free memory to allocate the hash table for the conversion database. The bridge will be disabled.

Cause: Severe shortage of memory.

Action: Reduce routing table sizes in other protocols, use system with less protocols, expand memory in router.

SRT.056

Level: CI-ERROR

Short Syntax: SRT.056 Input SR q ovf *source_mac*-> *dest_mac*, dropped, nt *network*

Long Syntax: SRT.056 Input source-routing queue overflow on frame from *source_mac* to *dest_mac*, packet dropped from network *network*

Description: The input queue for source-routed frames to be forwarded is too long, and this frame has been dropped to attempt to alleviate the congestion.

Cause: Bursty traffic may be causing congestion.

Action: Wait for burst to subside.

Cause: Too much traffic for forwarder to forward.

Action: Reconfigure network. Increase speed of router.

Cause: Inadequate buffer resources.

Action: Examine memory statistics in GWCON.

SRT.057

Level: P-TRACE

Short Syntax: SRT.057 *source_mac*-> *dest_mac* brdg port *port* nt *network* to port *port* nt *network*

Long Syntax: SRT.057 Frame from *source_mac* to *dest_mac* bridged from port number *port* network *network* to port number *port* network *network*

Description: A frame is being bridged between these two interfaces. The destination address was known, so it was sent only to the correct destination network.

SRT.058

Level: CE-ERROR

Short Syntax: SRT.058 TB->SR *source_mac*-> *dest_mac* too big (*reformatted_length* > *output_maximum*) for port *port* nt *network*, drop

Long Syntax: SRT.058 Transparent frame converted to source-routed frame from *source_mac* to *dest_mac* is too big (reformatted length *reformatted_length* bytes > output maximum size *output_maximum* bytes) for port *port* network *network*, dropped

Description: The specified transparent bridge frame is too large to send as a source-routed frame on this outgoing port and network. The reformatted_length is the size of the frame including MAC headers and RIF

after any mapping of data link headers.

Cause: Host on network with large maximum frame size sending to host on network with smaller maximum frame size.

Action: Reconfigure sending host to not send such large frames. If frame is of a routable protocol supporting fragmentation (such as IP or ISO) or maximum frame size determination (DNA or XNS), convert to using routing instead of bridging.

Cause: Host on network with large maximum frame size sending to host via an intervening network with smaller maximum frame size.

Action: Reconfigure network to use networks with large maximum frame size (such as FDDI or 802.5) as the backbone networks. Reconfigure port costs in Spanning Tree Protocol to favor spanning trees via networks with large maximum frame sizes.

SRT.059

Level: P-TRACE

Short Syntax: SRT.059 TB->SR *source_mac*-> *dest_mac* (RIF *RIF*) brdg port *port* nt *network* to port *port* nt *network*

Long Syntax: SRT.059 Transparent frame converted to source-routed frame from *source_mac* to *dest_mac* (RIF *RIF*) bridged from port number *port* network *network* to port number *port* network *network*

Description: A frame is being conversion bridged between these two interfaces. The destination address and RIF were known, so it was sent only to the correct destination network.

SRT.060

Level: P-TRACE

Short Syntax: SRT.060 TB->SR *source_mac*-> *dest_mac* (RIF *RIF*) brdg-all port *port* nt *network* to port *port* nt *network*

Long Syntax: SRT.060 Transparent frame converted to source-routed frame from *source_mac* to *dest_mac* (RIF *RIF*) bridged to all ports from port number *port* network *network* to port number *port* network *network*

Description: A frame is being conversion bridged to all active source-routing ports. This happens when the frame destination is a multicast or when the frame destination is not in the source-routing learning database. There will be one message for each port the frame is sent on.

SRT.061

Level: UE-ERROR

Short Syntax: SRT.061 SRF rcv *source_mac*-> *dest_mac* (RIF *RIF*) to disabl port *port*, nt *network*, disc

Long Syntax: SRT.061 Specifically routed frame frame received from *source_mac* to *dest_mac* (RIF *RIF*) to disabled port *port*, network *network*, discarded

Description: A Specifically Routed frame has been received whose RIF would send it on the specified port, but that port is not configured for bridging.

Cause: End station using invalid RIF. This can happen when the end station acquires a RIF, and caches it, but in the interim the bridge has been reconfigured and restarted.

SRT.062

Level: CE-ERROR

Short Syntax: SRT.062 Warning:SR->TB *source_mac*-> *dest_mac* too big (*reformatted_length* > *output_maximum*) from port *port* nt *network*

Long Syntax: SRT.062 Source-routed frame converted to transparent frame from *source_mac* to *dest_mac* is too big (reformatted length *reformatted_length* bytes > output maximum size *output_maximum* bytes) from port *port* network *network*, may get dropped.

Description: The specified source-routed frame is larger than that is allowed by LF-BIT configuration for the transparent bridge domain. After mapping to the MAC headers of the outgoing port, the packet may get dropped if it exceeds the MSDU limit of the port.

Cause: Source-routing host not honoring maximum frame size that was determined in source-routing threading process.

Action: Correct behavior of host.

Cause: Host on network with large maximum frame size sending to host on network with smaller maximum frame size.

Action: Reconfigure sending host to not send such large frames. If frame is of a routable protocol supporting fragmentation (such as IP or ISO) or maximum frame size determination (DNA or XNS), convert to using routing instead of bridging.

Cause: Host on network with large maximum frame size sending to host via an intervening network with smaller maximum frame size.

Action: Reconfigure network to use networks with large maximum frame size (such as FDDI or 802.5) as the backbone networks. Reconfigure port costs in Spanning Tree Protocol to favor spanning trees via networks with large maximum frame sizes.

SRT.063

Level: UI-ERROR

Short Syntax: SRT.063 No buf to dup *routing_type* frame *source_mac*-> *dest_mac* for SR->TB from port *port* nt *network*

Long Syntax: SRT.063 No buffer available to duplicate *routing_type* frame from *source_mac* to *dest_mac* for source-routing to transparent bridging conversion from port *port* network *network*

Description: No buffer available to copy a frame in order to send Routes Explorer (ARE) or Spanning Tree Explorer (STE) routing_type frame out as a transparent bridged frame in the transparent bridging domain. No copy of this frame will be sent into the transparent bridge domain.

Cause: Severe packet buffer shortage.

Action: Check memory statistics in GWCON to verify packet buffer level.

Cause: Traffic peak using all available buffers.

Action: This is the problem if this message occurs infrequently.

SRT.064

Level: UI-ERROR

Short Syntax: SRT.064 No mem for conv db (req *requested_size*), disabl

Long Syntax: SRT.064 No memory for conversion database (desired size *requested_size* bytes), disabling bridge

Description: There is not enough free memory to allocate the conversion database. The bridge will be disabled.

Cause: Severe shortage of memory.

Action: Reduce routing table sizes in other protocols, use system with less protocols, expand memory in router.

SRT.065

Level: UI-ERROR

Short Syntax: SRT.065 Can't add stat ent *MAC_address* on nt *network*

Long Syntax: SRT.065 Can not add static entry for address *MAC_address* on network *network*

Description: An attempt to add a particular static entry to the internal database of a bridging interface having internal filtering failed.

Cause: Hardware failure or software bug.

Action: Contact customer service.

SRT.066

Level: UI-ERROR

Short Syntax: SRT.066 Can't ena TB on nt *network*

Long Syntax: SRT.066 Can not enable transparent bridging on network *network*

Description: The bridge has been configured to enable transparent bridging on an IEEE 802.5 Token-Ring network that does not have the hardware to support transparent bridging. Transparent bridging will not be enabled on this interface.

Cause: Misconfiguration.

Action: Correct configuration.

SRT.067

Level: UI-ERROR

Short Syntax: SRT.067 SRF *source_mac*-> *dest_mac* (RIF *RIF*) fwd to disabl port *port*, nt *network*, disc

Long Syntax: SRT.067 Specifically routed frame frame from *source_mac* to *dest_mac* (RIF *RIF*) forwarded to disabled port *port*, network *network*, discarded

Description: A Specifically Routed frame has been sent on a port, but that port is not configured for bridging. This should never happen, since prior checks should prevent calling this code if the port is not configured for bridging.

SRT.068

Level: UI-ERROR

Short Syntax: SRT.068 Eth type table full for *ethernet_type*

Long Syntax: SRT.068 Ethernet type table full for Ethernet type *ethernet_type*

Description: There is no space in the Ethernet type registration table for the specified ethernet_type. This happens when there are too many hash collisions, and there are not enough overflow buckets.

Cause: Too many added Ethernet type filters.

Action: Do not use as many Ethernet type filters.

SRT.069

Level: UI-ERROR

Short Syntax: SRT.069 SNAP type table full for PID *protocol*

Long Syntax: SRT.069 Subnetwork Access Protocol table full for Protocol Identifier type *protocol*

Description: There is no space in the SNAP PID registration table for the specified protocol. This happens when there are too many hash collisions, and

there are not enough overflow buckets.

Cause: Too many added SNAP PID filters.

Action: Do not use as many SNAP PID filters.

SRT.070

Level: P-TRACE

Short Syntax: SRT.070 *source_mac*-> *dest_mac* drp, dst add flt, port *port* nt *network*

Long Syntax: SRT.070 Frame from *source_mac* to *dest_mac* dropped, destination address filtered, port *port* network *network*

Description: A MAC frame has been received by the hardware, but is being dropped because the destination MAC address is being administratively filtered by the bridge. The frame will be dropped.

Cause: Receipt of frame whose destination MAC address matches the exclusive filter.

SRT.071

Level: UI-ERROR

Short Syntax: SRT.071 SR not supp on port *port*, nt *network*

Long Syntax: SRT.071 Source Routing not supported on port *port*, network *network*

Description: Source Routing is configured on the port which is attached to an underlying network which inherently does not support source routing type of functionality. Such networks are Ethernet and FDDI. Bridge disables source routing on the port.

Cause: User misconfiguration.

SRT.081

Level: P-TRACE

Short Syntax: SRT.081 NB STE converted to SRF (RIF *RIF*) *source_mac*-> *dest_mac* from port *port*

Long Syntax: SRT.081 NETBIOS STE converted to SRF (RIF *RIF*) *source_mac*-> *dest_mac* from port *port*

Description: A NETBIOS STE converted to SRF by NETBIOS Name Caching

---

SRT.082

Level: P-TRACE

Short Syntax: SRT.082 NB STE not converted, RIF too long

Long Syntax: SRT.082 NETBIO STE not converted, RIF too long

Description: NETBIO STE not converted, RIF too long

SRT.083

Level: P-TRACE

Short Syntax: SRT.083 NB find-name STE filtered (RIF *RIF*) *source_mac*-> *dest_mac* from port *port*

Long Syntax: SRT.083 NETBIOS find-name STE filtered (RIF *RIF*) *source_mac*-> *dest_mac* from port *port*

Description: A NETBIOS find-name STE has been filtered

SRT.084

Level: P-TRACE

Short Syntax: SRT.084 Hello BPDU dropped, STP disabled on prt *port*, nt *network*

Long Syntax: SRT.084 Hello BPDU dropped because STP disabled on port *port*, network *network*

Description: A spanning tree Hello BPDU frame was received on a port that has been disabled for spanning tree participation by the "disable tree port#" command.

SRT.085

Level: UI-ERROR

Short Syntax: SRT.085 Frame relay Port *port* config on non-Frame Relay intf *network_number*

Long Syntax: SRT.085 Frame relay Port *port* configured on non-Frame relay interface *network_number*

Description: This port uses a Frame Relay network. However, subsequent to bridge configuration, the interface configuration changed such that the interface is no longer configured to be Frame Relay, or re-ordered the device records.

Cause: Inconsistency between router interface configuration and bridge configuration.

Action: Correct the data link support on the interface to be of type Frame Relay and/or correct the interface number in the bridge configuration.

SRT.086

Level: UI-ERROR

Short Syntax: SRT.086 Port *port*, cir= *circuit_name* reg with Frly nt *network_number* failed, rsn= *reason*

Long Syntax: SRT.086 Bridge port *port* with circuit= *circuit_name* registration with Frame relay network *network_number* failed, reason= *reason*

Description: During bridge initialization, Frame Relay bridge ports attempt to register with their associated Frame Relay interfaces. This message indicates a failure in this process.

Cause: The reasons for failure are: (1) Insufficient

memory. (2) Another bridge port is using this circuit. (3) The circuit is unknown.

Action: (1) Reevaluate the memory requirements. (2) Eliminate or reconfigure the conflicting bridge port which uses the same circuit (3) Configure the circuit in the frame relay configuration

---

SRT.087

Level: UE-ERROR

Short Syntax: SRT.087 ARE max RD drop *source_mac*-> *dest_mac* from port *port*, nt *network*

Long Syntax: SRT.087 All Routes Explorer exceeds maximum Route Descriptors from *source_mac* to *dest_mac* from port *port*, network *network*

Description: An All Routes Explorer (ARE) source-routed frame has more Route Descriptors than this bridge is configured to allow for ARE frames. The frame will be dropped.

Cause: Upstream bridge has an ARE RD limit inconsistent with this bridge.

Action: Reconfigure all bridges in source-routing domain to have consistent ARE RD limit.

Cause: Network has too many hops for configured ARE RD limit.

Action: Reconfigure all bridges in source-routing domain to have ARE RD limit consistent with the diameter of the domain.

SRT.088

Level: CE-ERROR

Short Syntax: SRT.088 *routing_type* inv LIN (RIF *RIF*) drop *source_mac*-> *dest_mac* from port *port*, nt *network*

Long Syntax: SRT.088 *routing_type* with invalid LIN (RIF *RIF*) from *source_mac* to *dest_mac* from port *port*, network *network*

Description: A source-routed frame of broadcast-routed frame (ARE or STE) type has an incoming LIN (LAN ID In) that does not match the configured segment number of the bridge port on which it was received. It will be dropped.

Cause: Configuration mismatch among bridges attached to the segment in question.

Action: Reconfigure for legal configuration. All bridges must be configured with the same LAN ID for each segment.

Cause: End station using completely invalid RIF.

Action: Find out why station is using this RIF.

SRT.090

Level: UI-ERROR

Short Syntax: SRT.090 ATM Port *port* config on non-ATM intf *network_number*

Long Syntax: SRT.090 ATM Port *port* configured on non-ATM interface *network_number*

Description: This port uses an ATM network. However, subsequent to bridge configuration, the interface configuration changed such that the interface is no longer configured to be ATM, or re-ordered the device records.

Cause: Inconsistency between router interface configuration and bridge configuration.

Action: Correct the data link support on the interface to be of type ATM and/or correct the network number in the bridge configuration.

SRT.091

Level: UI-ERROR

Short Syntax: SRT.091 Port *port*, vpi= *vpi* vci= *vci* reg with ATM nt *network_number* failed, rsn= *reason*

Long Syntax: SRT.091 Bridge port *port* with vpi= *vpi* vci= *vci* registration with ATM network *network_number* failed, reason= *reason*

Description: During bridge initialization, ATM bridge ports attempt to register with their associated ATM interfaces. This message indicates a failure in this process.

Cause: The reasons for failure are: (1) Insufficient memory. (2) Another bridge port is using this circuit. (3) The circuit is unknown.

Action: (1) Reevaluate the memory requirements. (2) Eliminate or reconfigure the conflicting bridge port which uses the same circuit (3) Configure the circuit in the ATM configuration.

SRT.092

Level: U-INFO

Short Syntax: SRT.092 DMAC addr. MAX limit. Not adding into dbase

Long Syntax: SRT.092 DMAC addr maximum limit reached. This addr won't be added to SR database

Description: The bridge database already has 7 duplicate MAC addresses. Bridge is detecting another duplicate MAC address.

Action: This is common in an environment where more than 7 duplicate MAC addresses exist.

SRT.093

Level: U-INFO

Short Syntax: SRT.093 DMAC seg. mismatch. (RIF *RIF*) SA- *source_mac* from port *port*, nt *network*

Long Syntax: SRT.093 DMAC last segment mismatch. (RIF *RIF*) from *source_mac* from port *port*, network *network*

Description: SRF frame was received with originating segment no. that didn't match the PRIMARY or SECONDARY RIF.

Action: This is common in an environment where duplicate MAC address exists on more than 2 different segments.

SRT.094

Level: U-INFO

Short Syntax: SRT.094 DMAC RIF not updated.(RIF *RIF*) SA- *source_mac* from port *port*, nt *network*

Long Syntax: SRT.094 DMAC RIF not updated.(RIF *RIF*) from *source_mac* from port *port*, network *network*

Description: Another RIF was received within resolution time period and it won't be used to refresh database entry. This is common when All Route Broadcast (ARE) frame is sent by station.

SRT.095

Level: UE-ERROR

Short Syntax: SRT.095 Cannot dynamically add/reset bridge port, nt *network_number*, reason = *reason*

Long Syntax: SRT.095 Bridge port associated with network *network_number* cannot be added/reset, reason = *reason*

Description: Cannot configure the bridge port associated with the interface being added or reset. Some changes to the bridge configuration cannot be made without a restart.

Cause: The reasons for failure are: (1) NetBIOS filters are configured for the port being added/reset. (2) LNM is configured for the port being added/reset. (3) 1:1 SRB configuration may have changed. (4) Addition/reset of the associated bridge port would cause a change in the bridge personality (type) (i.e., STB, SRB, STB & SRB, SRT, SR-TB, ASRT). (5) Bridge options have been changed (i.e., Change to internal virtual segment).

Action: Restart the system.

SRT.096

Level: UI-ERROR

Short Syntax: SRT.096 No mem for multiaccess hash tab (req *requested_size*), disabl

Long Syntax: SRT.096 No memory for multiaccess hash table (desired size *requested_size* bytes), disabling multiaccess bridge ports

Description: There is not enough free memory to allocate the hash table for the multiaccess database. Multiaccess bridge ports will be disabled.

Cause: Severe shortage of memory.

Action: Reduce routing table sizes in other protocols, use system with less protocols, expand memory in router.

SRT.097

Level: UI-ERROR

Short Syntax: SRT.097 No mem for multiaccess db (req *requested_size*), disabl

Long Syntax: SRT.097 No memory for multiaccess database (desired size *requested_size* bytes), disabling multiaccess bridge ports

Description: There is not enough free memory to allocate the multiaccess database. Multiaccess bridge ports will be disabled.

Cause: Severe shortage of memory.

Action: Reduce routing table sizes in other protocols, use system with less protocols, expand memory in router.

SRT.098

Level: UI-ERROR

Short Syntax: SRT.098 FR/ATM Port *port* config on non-FR/ATM intf *network_number*

Long Syntax: SRT.098 FR/ATM Port *port* configured on non-FR/ATM interface *network_number*

Description: This port uses a Frame Relay or ATM network. However, subsequent to bridge configuration, the interface configuration changed such that the interface is no longer configured to be Frame Relay or ATM, or re-ordered the device records.

Cause: Inconsistency between router interface configuration and bridge configuration.

Action: Correct the data link support on the interface to be of type Frame Relay or ATM and/or correct the interface number in the bridge configuration.

SRT.099

Level: U-INFO

Short Syntax: SRT.099 *source_mac*-> *dest_mac* dropped, output circuit on port *port* nt *network* not active

Long Syntax: SRT.099 Frame from *source_mac* to *dest_mac* dropped, output circuit on port *port* network *network* is not active

Description: A MAC frame was being bridged on a multiaccess port, but the circuit on the destination port was not in ″active″ state. It will not be sent.

Cause: Circuit on output port is not in use.

Action: None needed, port will become ″active″ when data is received on the circuit.

Panic SRTimem

Short Syntax: SRT: memory allocation failed

Description: The SRT forwarder failed to allocate sufficient memory to hold its most fundamental tables.

Cause: Insufficient free memory.

Action: Making databases for other protocols smaller.

Action: Increase memory size.

Fatal srtiisrt

Short Syntax: SRT: Invalid i_srt on input

Description: The i_srt flag passed from the handler to forwarder has an invalid value.

Cause: Software bug.

Action: Take a crash dump and contact customer service.

Fatal srtuimed

Short Syntax: SRT: unknown input media

Description: The input net type is not one of the ones understood by the SRT bridge (802.3/Ethernet, FDDI, or 802.5).

Cause: Software bug.

Action: Take a crash dump and contact customer service.

Chapter 107. Spanning Tree Protocol (STP)

This chapter describes Spanning Tree Protocol (STP) messages. For information on message content and how to use the message, refer to the Introduction.

STP.001

Level: C-TRACE

Short Syntax: STP.001 Cfg BPDU rcv frm *source_address bridge_type- bridge_instance* port *bridge_port*, nt *network*

Long Syntax: STP.001 Configuration BPDU received frm *source_address* on *bridge_type- bridge_instance* port *bridge_port*, network *network*

Description: A configuration BPDU has been received from the specified MAC address.

Cause: Another bridge on the same network as this bridge on this port.

STP.002

Level: C-TRACE

Short Syntax: STP.002 Tcn BPDU rcv frm *source_address bridge_type- bridge_instance* port *bridge_port*, nt *network*

Long Syntax: STP.002 Topology change notification BPDU received frm *source_address* on *bridge_type- bridge_instance* port *bridge_port*, network *network*

Description: A topology change notification BPDU has been received from the specified MAC address.

Cause: Topology change has been detected at or downstream of the sending bridge.

Action: None needed, the message should stop when the topology change is acknowledged by the root bridge.

STP.003

Level: UE-ERROR

Short Syntax: STP.003 Ukn BPDU type *BDPU_type* rcv frm *source_address bridge_type- bridge_instance* port *bridge_port*, nt *network*

Long Syntax: STP.003 Unkown BPDU type *BDPU_type* received frm *source_address* on *bridge_type- bridge_instance* port *bridge_port*, network *network*

Description: A BPDU with an undefined value in the BPDU Type field was received from the specified host. It will be ignored.

Cause: Programming error at remote bridge.

Action: Correct remote node.

Cause: Data corruption in received packet.

Action: Eliminate source of data corruption.

STP.005

Level: UE-ERROR

Short Syntax: STP.005 BPDU bd ver *Protocol_Version_Identifier* frm *source_address bridge_type- bridge_instance* port *bridge_port*, nt *network*

Long Syntax: STP.005 BPDU bad Version *Protocol_Version_Identifier* frm *source_address* on *bridge_type- bridge_instance* port *bridge_port*, network *network*

Description: A configuration BPDU has been received with a Protocol Version Identifier that is not 00. It will be ignored.

Cause: Programming error at remote bridge.

Action: Correct remote node.

Cause: Data corruption in received packet.

Action: Eliminate source of data corruption.

STP.006

Level: UE-ERROR

Short Syntax: STP.006 Cfg BPDU trunc (*length* byt) frm *source_address bridge_type- bridge_instance* port *bridge_port*, nt *network*

Long Syntax: STP.006 Configuration BPDU tuncated (*length* bytes) frm *source_address* on *bridge_type- bridge_instance* port *bridge_port*, network *network*

Description: A configuration BPDU has been received which is less than 35 bytes in length. It will be ignored.

Cause: Programming error at remote bridge.

Action: Correct remote node.

Cause: Data corruption in received packet.

Action: Eliminate source of data corruption.

STP.007

Level: UE-ERROR

Short Syntax: STP.007 Cfg BPDU unk flg *flags* frm *source_address bridge_type- bridge_instance* port *bridge_port*, nt *network*

Long Syntax: STP.007 Configuration BPDU unknown flags *flags* frm *source_address* on *bridge_type- bridge_instance* port *bridge_port*, network *network*

Description: A configuration BPDU has been received which has undefined bits set in the flags field. It will be ignored.

Cause: Programming error at remote bridge.

Action: Correct remote node.

Cause: Data corruption in received packet.

Action: Eliminate source of data corruption.

STP.008

Level: UE-ERROR

Short Syntax: STP.008 Tcn BPDU trunc (*length* byt) frm *source_address bridge_type- bridge_instance* port *bridge_port*, nt *network*

Long Syntax: STP.008 Topology change notification BPDU tuncated (*length* bytes) frm *source_address* on *bridge_type- bridge_instance* port *bridge_port*, network *network*

Description: A topology change notificaton BPDU has been received that is less than 4 bytes in length. It will be ignored.

Cause: Programming error at remote bridge.

Action: Correct remote node.

Cause: Data corruption in received packet.

Action: Eliminate source of data corruption.

STP.009

Level: UI-ERROR

Short Syntax: STP.009 No buf for BPDU *bridge_type- bridge_instance* port *bridge_port*, nt *network*

Long Syntax: STP.009 No buffer to send BDPU on *bridge_type- bridge_instance* port *bridge_port*, network *network*

Description: No packet buffer was available to construct and send a BDPU on the specified port.

Cause: Severe packet buffer shortage.

Action: Check memory statistics in GWCON to verify packet buffer level.

Cause: Traffic peak using all available buffers.

Action: This is the problem if this message occurs infrequently.

STP.010

Level: P-TRACE

Short Syntax: STP.010 Sndg cfg BPDU *bridge_type- bridge_instance* port *bridge_port*, nt *network*

Long Syntax: STP.010 Sending Configuration BPDU on *bridge_type- bridge_instance* port *bridge_port* network *network*

Description: A Configuration BPDU will be sent on the specified port. This is done normally on a periodic basis as part of the spanning tree protocol. The flags field in this BPDU is zero, e.g., neither the Topology Change or the Topology Change Acknowledgement bits are set.

STP.011

Level: P-TRACE

Short Syntax: STP.011 Sndg Cfg BPDU flgs *TC TCA bridge_type- bridge_instance* port *bridge_port*, nt *network*

Long Syntax: STP.011 Sending Configuration BPDU with flags *TC TCA* on *bridge_type- bridge_instance* port *bridge_port*, network *network*

Description: A Configuration BPDU will be sent on the specified port. This is done normally on a periodic basis as part of the spanning tree protocol. TC will be displayed if the Topology Change bit is set in the Flags byte of the BPDU, TCA will be displayed if the Topology Change Acknowledge bit is set in the flags byte.

Cause: The Topology Change flag is set if this bridge is the root and it knows that there is a topology change in process. Also, non-root bridges propogate this bit received in incoming Configuration BPDUs.

Action: None needed, this flag will be set only for the sum of the current maximum age and current forward delay parameters (as propagated by the root bridge).

Cause: The Topology Change Acknowledge flag is set if this bridge has received a Topology Change Notification BPDU, and this port is the Designated Bridge on its LAN.

Action: None needed, this flag will only be sent on one BDPU.

STP.012

Level: P-TRACE

Short Syntax: STP.012 Sndg tcn BPDU *bridge_type- bridge_instance* port *bridge_port*, nt *network*

Long Syntax: STP.012 Sending Topology Change Notification BPDU on *bridge_type- bridge_instance* port *bridge_port* network *network*

Description: A Topology Change Notification BPDU will be sent on the specified port. These are sent on the

root port of non-root bridges when they detect a topology change in the spanning tree.

Cause: A bridge, or an interface on a bridge, has gone up or down in this spanning tree.

Action: None needed. This state persists only until a topology change acknowledgement is received, or a timeout that indicates that the old root bridge is no longer reachable.

STP.013

Level: UI-ERROR

Short Syntax: STP.013 BPDU snd fld, rsn *reason_code*, *bridge_type- bridge_instance* port *bridge_port*, nt *network*

Long Syntax: STP.013 BPDU send failed for reason code *reason_code* on *bridge_type- bridge_instance* port *bridge_port* network *network*

Description: The attempt to queue a BPDU for transmission on the specified port failed.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

STP.014

Level: U-INFO

Short Syntax: STP.014 Blocking *bridge_type-bridge_instance* port *bridge_port*, nt *network*, det topol chg

Long Syntax: STP.014 Blocking *bridge_type-bridge_instance* port *bridge_port*, network *network*, detecting topology change

Description: This port has just been placed in Blocking state. This is a change in the topology, so this bridge detects a topology change. This will in turn cause topology change notifications to be sent.

Cause: A bridge, or an interface on a bridge, has gone up or down in this spanning tree.

Action: None needed. This is normal when there are changes.

STP.015

Level: U-INFO

Short Syntax: STP.015 Topol chg detected *bridge_type-bridge_instance* port *bridge_port*, nt *network*

Long Syntax: STP.015 Topology change detected on *bridge_type- bridge_instance* port *bridge_port*, network *network*

Description: A topology change notification has been received on this port, and this port is the designated port on its LAN. This causes the protocol to enter topology change notification state. The topology change will be acknowledged towards the sender, and propagated towards the root.

Cause: A bridge, or an interface on a bridge, has gone up or down in this Spanning Tree.

Action: None needed. This is normal when there are changes.

STP.016

Level: U-INFO

Short Syntax: STP.016 Select as root *bridge_type-bridge_instance*, det topol chg

Long Syntax: STP.016 Selected as root on *bridge_type-bridge_instance*, detecting topology change

Description: This bridge has just selected itself as the root of the spanning tree when it previously had not been. This causes the bridge to enter topology change notification state.

Cause: A bridge, or an interface on a bridge, has gone up or down in this spanning tree.

Action: None needed. This is normal when there are changes.

Cause: This is the first bridge up, thus it is the root of the tree.

STP.017

Level: C-INFO

Short Syntax: STP.017 Tply chg ackd *bridge_type-bridge_instance* port *bridge_port*, nt *network*

Long Syntax: STP.017 Topology change acknowledged on *bridge_type- bridge_instance* port *bridge_port*, network *network*

Description: A topology change acknowledgement has been detected on the specified port. This port is the root port of the bridge.

Cause: Bridge on same LAN as our root port has set topology change acknowledgement flag in outgoing Configuration BDPU. This was in response to a topology change notification that this bridge originated or propagated.

Action: None needed. This is the normal conclusion of topology change notification.

STP.018

Level: C-INFO

Short Syntax: STP.018 Acking tply chg *bridge_type- bridge_instance* port *bridge_port*, nt *network*

Long Syntax: STP.018 Acknowledging topology change on *bridge_type- bridge_instance* port *bridge_port*, network *network*

Description: A topology change notification is being acknowledged on the specified port. This is done when a topology change notification is received on a port that is the designated port for that LAN.

Cause: Change on bridge topology downstream of this bridge.

Action: None needed. This is a normal port of reconfiguration of the spanning tree.

STP.019

Level: C-TRACE

Short Syntax: STP.019 Tplgy chg notif timer expired *bridge_type- bridge_instance*

Long Syntax: STP.019 Topology Change Notification timer expired on *bridge_type- bridge_instance*

Description: The Topology Change timer expired. This bridge will cease sending topology change notification BPDU's on its root port.

Cause: This timer expires when the bridge has been in Topology Change Notification state for the bridge hello timer period.

Action: None needed, this is the normal conclusion of this state.

STP.020

Level: C-TRACE

Short Syntax: STP.020 Tplgy chg timer expired *bridge_type- bridge_instance*

Long Syntax: STP.020 Topology Change timer expired on *bridge_type- bridge_instance*

Description: The Topology Change timer expired. This bridge, which is the root, will cease sending the Topology Change in its Configuration BPDUs.

Cause: This happens when this root bridge has been in Topology Change state for the sum of current maximum age and current forward delay parameters.

Action: None needed, this is the normal conclusion of this state.

STP.021

Level: U-INFO

Short Syntax: STP.021 Msg age tmr exp *bridge_type- bridge_instance* port *bridge_port*, nt *network*, try Root

Long Syntax: STP.021 Message age timer expired on *bridge_type- bridge_instance* port *bridge_port*, network *network*, will try and become root

Description: The message age timer has expired on this port. The bridge will attempt to become the root. It will become the designated port on that LAN.

Cause: No Configuration BPDU's being received on this interface. Either there are no bridges on this LAN, or they are down.

STP.022

Level: C-TRACE

Short Syntax: STP.022 Hello tmr exp *bridge_type- bridge_instance*

Long Syntax: STP.022 Hello timer expired on *bridge_type- bridge_instance*

Description: The hello timer has expired on this port. Configuration BPDUs will be sent on all ports.

STP.023

Level: C-TRACE

Short Syntax: STP.023 Stop msg age tmr *bridge_type- bridge_instance* port *bridge_port*, nt *network*

Long Syntax: STP.023 Stopping message age timer for *bridge_type- bridge_instance* port *bridge_port*, network *network*

Description: Stopping the message age timer on this port because is it the designated port on its LAN.

STP.024

Level: U-INFO

Short Syntax: STP.024 Not root *bridge_type- bridge_instance*, stop hello tmr

Long Syntax: STP.024 Not root anymore on *bridge_type- bridge_instance*, stopping hello timer

Description: This bridge has just decided that it is no longer the root bridge of the spanning tree. The hello timer will also be cancelled.

STP.025

Level: C-INFO

Short Syntax: STP.025 Stop tplgy chg age tmr *bridge_type- bridge_instance*

Long Syntax: STP.025 Stopping topology change timer

for *bridge_type- bridge_instance*

Description: Stopping the topology change timer because this bridge is no longer the root.

STP.026

Level: U-INFO

Short Syntax: STP.026 Root *bridge_type- bridge_instance*, strt hello tmr

Long Syntax: STP.026 Selected as root on *bridge_type- bridge_instance*, starting hello timer

Description: This bridge has just decided that it is the root bridge of the spanning tree. The hello timer will be started.

STP.027

Level: C-TRACE

Short Syntax: STP.027 Strt msg age tmr *bridge_type- bridge_instance* port *bridge_port*, nt *network*

Long Syntax: STP.027 Starting message age timer for *bridge_type- bridge_instance* port *bridge_port*, network *network*

Description: Starting the message age timer on this port.

STP.028

Level: C-TRACE

Short Syntax: STP.028 Attmpt root *bridge_type- bridge_instance*, strt hello tmr

Long Syntax: STP.028 Attempting to become root on *bridge_type- bridge_instance*, starting hello timer

Description: This bridge is attempting to become the root bridge of the spanning tree. The hello timer will be started.

STP.032

Level: UI-ERROR

Short Syntax: STP.032 DROP: *bpdu_type* BPDU frm recvd on non-parti port *bridge_port*, nt *network*

Long Syntax: STP.032 DROP: *bpdu_type* BPDU frame received on non-participating port *bridge_port*, network *network*

Description: A Source Route Bridge BPDU or IEEE802.1D BPDU has been received, but the port is not participating in the SRB or IEEE802.1D Spanning Tree Protocol.

Fatal stpubpdu

Short Syntax: Attempt to send unknown BPDU type

Description: The code attempted to send an unknown type of BPDU.

Cause: Possible software bug.

Action: Get crash dump, contact customer service.

Chapter 108. SRT FDDL-Related ELS messages

This chapter describes SRT FDDL-Related ELS messages messages. For information on message content and how to use the message, refer to the Introduction.

SRTF.001

Level: U-INFO

Short Syntax: SRTF.001 fddl: duplicate registration

Long Syntax: SRTF.001 fddl: duplicate registration

Description: The SRT protocol is already registered with the fddl interface.

SRTF.002

Level: UI-ERROR

Short Syntax: SRTF.002 fddl: registration failed

Long Syntax: SRTF.002 fddl: registration failed

Description: The SRT protocol registeration with the fddl interface failed.

SRTF.003

Level: UI-ERROR

Short Syntax: SRTF.003 fddl: *function_name* call failed

Long Syntax: SRTF.003 fddl: *function_name* call failed

Description: A fddl_api call by SRT was unsuccessful.

SRTF.004

Level: U-INFO

Short Syntax: SRTF.004 fddl send interfaceConfigSet for ifNum *interfaceNum* ,state= *new_state*

Long Syntax: SRTF.004 fddl send interfaceConfigSet for ifNum *interfaceNum* ,state= *new_state*

Description: A port state change event occured. The new state is sent to the respective interface.

SRTF.005

Level: U-INFO

Short Syntax: SRTF.005 fddl send macAddrDeleteRequest for addr= *mac_addr*

Long Syntax: SRTF.005 fddl send macAddrDeleteRequest for addr= *mac_addr*

Description: A mac address delete message is sent to the fddl interface.

SRTF.006

Level: U-INFO

Short Syntax: SRTF.006 fddl send macAddrPurgeRequest

Long Syntax: SRTF.006 fddl send macAddrDeleteRequest

Description: A database purge request message is sent to the fddl interface.

SRTF.007

Level: U-INFO

Short Syntax: SRTF.007 fddl send macAddrQueryRequest for addr= *mac_addr*

Long Syntax: SRTF.007 fddl send macAddrQueryRequest for addr= *mac_addr*

Description: A mac address query request message is sent to the fddl interface.

SRTF.008

Level: U-INFO

Short Syntax: SRTF.008 fddl recv macAddrResolutionRequest for addr= *mac_addr*, from ifNum= *interfaceNum*

Long Syntax: SRTF.008 fddl recv macAddrResolutionRequest for addr= *mac_addr*, from ifNum= *interfaceNum*

Description: A mac address resolution request message is received.

SRTF.009

Level: U-INFO

Short Syntax: SRTF.009 fddl send macAddrResolutionReply for addr= *mac_addr*,ifNum= *interfaceNum*,vpiVci= *vpiVci*

Long Syntax: SRTF.009 fddl send macAddrResolutionReply for addr= *mac_addr*,ifNum= *interfaceNum*,vpiVci= *vpiVci*

Description: A mac address resolution reply message is sent to the fddl interface.

SRTF.010

Level: U-INFO

Short Syntax: SRTF.010 fddl recv macAddrQueryReply for addr= *mac_addr*,ifNum= *interfaceNum*,age= *age* secs

Long Syntax: SRTF.010 fddl recv macAddrQueryReply for addr= *mac_addr*,ifNum= *interfaceNum*,age= *age* seconds

Description: A mac address query reply message is received.

SRTF.011

Level: UI-ERROR

Short Syntax: SRTF.011 fddl macAddrQueryReply, entry not in database: *mac_addr*

Long Syntax: SRTF.011 fddl macAddrQueryReply, entry not in database: *mac_addr*

Description: A fddl query reply received for an address not in SRT database.

SRTF.012

Level: U-INFO

Short Syntax: SRTF.012 fddl recv macAddrLearnReqest for addr= *mac_addr*,ifNum= *interfaceNum*

Long Syntax: SRTF.012 fddl recv macAddrLearnReqest for addr= *mac_addr*,ifNum= *interfaceNum*

Description: A mac address learn request message is received.

SRTF.013

Level: UI-ERROR

Short Syntax: SRTF.013 fddl macAddrLearnRequest, entry not in database: *mac_addr*

Long Syntax: SRTF.013 fddl macAddrLearnRequest, entry not in database: *mac_addr*

Description: A fddl learn request error; an address not in SRT database even after learning it.

SRTF.014

Level: U-INFO

Short Syntax: SRTF.014 fddl send macAddrLearnReply for addr= *mac_addr*,ifNum= *interfaceNum*

Long Syntax: SRTF.014 fddl send macAddrLearnReply for addr= *mac_addr*,ifNum= *interfaceNum*

Description: A mac address learn reply message is sent with the addr and interface info.

SRTF.015

Level: U-INFO

Short Syntax: SRTF.015 fddl recv statisticsUpdateIndication for ifNum= *interfaceNum*

Long Syntax: SRTF.015 fddl recv statisticsUpdateIndication for ifNum= *interfaceNum*

Description: A statistics update indication message is received from an interface.

SRTF.016

Level: UI-ERROR

Short Syntax: SRTF.016 fddl invalid ifNum= *interfaceNum* in msg = *message_name*

Long Syntax: SRTF.016 fddl invalid ifNum= *interfaceNum* in msg = *message_name*

Description: A fddl call received with interface number. A net structure does not exist for this interface number.

SRTF.017

Level: UI-ERROR

Short Syntax: SRTF.017 fddl macAddrLearnRequest, bad port pointer in database entry: *mac_addr*

Long Syntax: SRTF.017 fddl macAddrLearnRequest, bad port pointer in database entry: *mac_addr*

Description: A fddl learn request error; an entry in the SRT database has a NULL port pointer.

Chapter 109. SuperElan (SE)

This chapter describes SuperElan (SE) messages. For information on message content and how to use the message, refer to the Introduction.

SE.001

Level: UI-ERROR

Short Syntax: SE.001 SE- *se_id*:No buf to dup broadcast frame 0x *source_mac*->0x *dest_mac* to port *port*, net- *network* int *int/ int_no*

Long Syntax: SE.001 SE- *se_id*:No buffer available to duplicate frame from 0x *source_mac* to 0x *dest_mac* on to port *port*, network *network* interface *int/ int_no*

Description: No buffer available to copy a frame in order to send a bridged frame on multiple interfaces. Bridged packets are sent on multiple interfaces either for multicast destination addresses or unknown unicast addresses. No copy of this frame will be sent on the specified port and network.

Cause: Severe packet buffer shortage.

Action: Check memory statistics in GWCON to verify packet buffer level.

Cause: Traffic peak using all available buffers.

Action: This is the problem if this message occurs infrequently.

SE.002

Level: UE-ERROR

Short Syntax: SE.002 SE- *se_id*:Bcast 802.3 bad len *actual_length claimed_length*, 0x *source_Ethernet_address*->0x *destination_Ethernet_address* net- *network*

Long Syntax: SE.002 SE- *se_id*:Broadcast packet received with a bad 802.3 length field actual *actual_length* claimed *claimed_length* from 0x *source_Ethernet_address* to 0x *destination_Ethernet_address* network *network*

Description: A broadcast packet was received with a type field that indicated 802.3 but was shorter than data length claimed in the 802.3 header.

SE.003

Level: UE-ERROR

Short Syntax: SE.003 SE- *se_id*:802.3 bad len *actual_length claimed_length*, 0x *source_Ethernet_address*->0x *destination_Ethernet_address* net- *network*

Long Syntax: SE.003 SE- *se_id*:packet received with a bad 802.3 length field actual *actual_length* claimed *claimed_length* from 0x *source_Ethernet_address* to 0x *destination_Ethernet_address* network *network*

Description: A non-broadcast packet was received with a type field that indicated 802.3 but was shorter than data length claimed in the 802.3 header.

SE.004

Level: UE-ERROR

Short Syntax: SE.004 SE- *se_id*:LOOP odd skip *count*, 0x *source_MAC_address*->0x *destination_MAC_address*, net- *network*

Long Syntax: SE.004 SE- *se_id*:Loopback Protocol, odd skipCount *count* from 0x *source_MAC_address* to 0x *destination_MAC_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet had an odd skipCount in the packet. It will be discarded.

Cause: Programming error on remote node.

SE.005

Level: UI-ERROR

Short Syntax: SE.005 SE- *se_id*:0x *source_mac*->0x *dest_mac* send failed, rsn *reason_code*, port *port* net- *network* int *int/ int_no*

Long Syntax: SE.005 SE- *se_id*:Sending Frame from 0x *source_mac* to 0x *dest_mac* failed, reason *reason_code*, on port *port* network *network* interface *int/ int_no*

Description: The sending of a packet being forwarded failed. The reason is the internal error code for the failure.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

SE.006

Level: CI-ERROR

Short Syntax: SE.006 SE- *se_id*:Input q ovf 0x *source_mac*->0x *dest_mac*, dropped, net- *network* int *int/ int_no*

Long Syntax: SE.006 SE- *se_id*:Input queue overflow on frame from 0x *source_mac* to 0x *dest_mac*, packet dropped from network *network* interface *int/ int_no*

Description: The input queue for frames to be forwarded is too long, and this frame has been dropped to attempt to alleviate the congestion.

Cause: Bursty traffic may be causing congestion.

Action: Wait for burst to subside.

Cause: Too much traffic for forwarder to forward.

Action: Reconfigure network. Increase speed of router.

Cause: Inadequate buffer resources.

Action: Examine memory statistics in GWCON.

SE.007

Level: CI-ERROR

Short Syntax: SE.007 SE- *se_id*:BPDU q ovf 0x *source_mac*->0x *network*, dropped, net- *int* int *int_no/*

Long Syntax: SE.007 SE- *se_id*:Bridge Protocol Data Unit input queue overflow on frame from 0x *source_mac* to 0x *network*, dropped from network *int* interface *int_no/*

Description: The input queue for Spanning Tree Protocol Bridge Protocl Data Units is too long, and this frame has been dropped to attempt to alleviate the congestion.

Cause: Source node streaming BDPU frames.

Action: Correct behavior of source node.

Cause: Too much traffic for forwarder to forward.

Action: Reconfigure network. Increase speed of router.

Cause: Inadequate buffer resources.

Action: Examine memory statistics in GWCON.

SE.008

Level: CE-ERROR

Short Syntax: SE.008 SE- *se_id*:0x *source_mac*->0x *dest_mac* too big (*reformatted_length* > *output_maximum*) for port *port* net- *network* int *int/ int_no*, dropped

Long Syntax: SE.008 SE- *se_id*:Frame from 0x *source_mac* to 0x *dest_mac* is too big (reformatted length *reformatted_length* bytes > output maximum size *output_maximum* bytes) for port *port* network *network* interface *int/ int_no*, dropped

Description: The specified frame is too large to send on this outgoing port and network. The reformatted_length is the size of the frame including MAC headers after any mapping of data link headers.

Cause: Host on network with large maximum frame size sending to host on network with smaller maximum frame size.

Action: Reconfigure sending host to not send such large frames. If frame is of a routable protocol supporting fragmentation (such as IP or ISO) or maximum frame size determination (DNA or XNS), convert to using routing instead of bridging.

Cause: Host on network with large maximum frame size sending to host via an intervening network with smaller maximum frame size.

Action: Reconfigure network to use networks with large maximum frame size (such as FDDI or 802.5) as the backbone networks. Reconfigure port costs in Spanning Tree Protocol to favor spanning trees via networks with large maximum frame sizes.

SE.009

Level: UE-ERROR

Short Syntax: SE.009 SE- *se_id*:0x *source_mac*->0x *dest_mac* dropped, net- *network* down

Long Syntax: SE.009 SE- *se_id*:Frame from 0x *source_mac* to 0x *dest_mac* dropped, input network *network* is down

Description: A frame has been received for bridging on a network that is down. It will be ignored.

Cause: A BDPU has been sent to the unicast address of the router on this interface.

Action: Correct action of sending node.

Cause: Internal state inconsistency.

SE.010

Level: C-INFO

Short Syntax: SE.010 SE- *se_id*:LOOP rcv 0x *source_MAC_address*->0x *destination_MAC_address*, net- *network*

Long Syntax: SE.010 SE- *se_id*:Loopback Protocol frame received from 0x *source_MAC_address* to 0x *destination_MAC_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet was received.

SE.011

Level: C-INFO

Short Syntax: SE.011 SE- *se_id*:0x *source_mac*->0x *dest_mac* dropped, input port *port* net- *network* int *int/ int_no* not forwarding

Long Syntax: SE.011 SE- *se_id*:Frame from 0x *source_mac* to 0x *dest_mac* dropped, input port *port* network *network* interface *int/ int_no* not in forwarding state

Description: A MAC frame was received on a port that is still only in "learning" state. Frames are only bridged when the input port is in "forwarding" state. While the port is still in "learning" state, they are only processed to learn the source addresses for the filtering database. The frame will not be bridged.

Cause: Normal part of transition to "forwarding" state.

SE.012

Level: C-INFO

Short Syntax: SE.012 SE- *se_id*:0x *source_mac*->0x *dest_mac* dropped, output port *port* net- *network* int *int/ int_no* not forwarding

Long Syntax: SE.012 SE- *se_id*:Frame from 0x *source_mac* to 0x *dest_mac* dropped, output port *port* network *network* interface *int/ int_no* not in forwarding state

Description: A MAC frame was being bridged, but the destination port was not in "forwarding" state. It will not be sent on that port.

Cause: Output port still in "learning" state.

Action: None needed, port will transition to "forwarding".

Cause: Static entry in filtering database points to port that is not in "forwarding" state.

SE.013

Level: C-INFO

Short Syntax: SE.013 SE- *se_id*:0x *source_mac*->0x *dest_mac* dropped, dst same LAN, port *port* net- *network* int *int/ int_no*

Long Syntax: SE.013 SE- *se_id*:Frame from 0x *source_mac* to 0x *dest_mac* dropped, destination on same LAN, port *port* network *network* interface *int/ int_no*

Description: A MAC frame has been received whose destination address is known to be on the same side of the bridge as the packet came from. It is dropped by the filtering logic since it does not need to be bridged.

Cause: Normal local traffic on network.

SE.014

Level: CI-ERROR

Short Syntax: SE.014 SE- *se_id*:0x *source_mac*->0x *dest_mac* drp, dst port *port* not enabled, net- *network* int *int/ int_no*

Long Syntax: SE.014 SE- *se_id*:Frame from 0x *source_mac* to 0x *dest_mac* dropped, destination port *port* not enabled, network *network* inteface *int/ int_no*

Description: A bridged frame was destined for a port which is not in "forwarding" state.

Cause: Filtering database points to port that is not in "forwarding" state.

SE.015

Level: C-INFO

Short Syntax: SE.015 SE- *se_id*:0x *source_mac*->0x *dest_mac* fwrd from port *port* net- *network* int *int/ int_no* to port *port* net- *network* int *int/ int_no*

Long Syntax: SE.015 SE- *se_id*:Frame from 0x *source_mac* to 0x *dest_mac* forwarded from port number *port* network *network* interface *int/ int_no* to port number *port* network *network* interface *int/ int_no*

Description: A frame is being bridged from the source MAC to the destination MAC.

SE.016

Level: C-INFO

Short Syntax: SE.016 SE- *se_id*:0x *source_mac*->0x *dest_mac* drp due to VLAN defs, from port *port* net- *network* int *int/ int_no* to port *port* net- *network* int *int/ int_no*

Long Syntax: SE.016 SE- *se_id*:0x *source_mac*->0x *dest_mac* dropped due to VLAN definitions, from port *port* net- *network* int *int/ int_no* to port *port* net- *network* int *int/ int_no*

Description: A frame was not transmitted on a specific interface due to VLAN filter.

SE.017

Level: CE-ERROR

Short Syntax: SE.017 SE- *se_id*:Unreg dst 0x *source_mac*->0x *dest_mac* Etype *Ethernet_type*, drp, net- *network*

Long Syntax: SE.017 SE- *se_id*:Frame from 0x *source_mac* to unregistered destination MAC address 0x *dest_mac*, Ethernet type *Ethernet_type*, dropped, network *network*

Description: A frame has been received for the Ethernet type which corresponds with an enabled protocol, but the destination MAC address is not

registered in the bridge. The frame will be dropped.

Cause: If dest_mac is a unicast address, a station on the LAN is sending frames for this protocol to the wrong next hop MAC address.

Action: Correct action of remote station.

Cause: If dest_mac is a multicast address, a station on the LAN may be sending frames to the wrong multicast address, or perhaps just to one that this router does not have enabled. Depending on the protocol, this may or may not be an error.

Action: Correct action of remote station, if neccessary.

SE.018

Level: CE-ERROR

Short Syntax: SE.018 SE- *se_id*:Unkn SNAP mfr code *number* from 0x *source_MAC* net- *network ID*

Long Syntax: SE.018 SE- *se_id*:Unknown SNAP manufacturer code *number* from 0x *source_MAC* net *network ID*

Description: This message is generated when a frame with an unknown organization code (not 000000) in the SNAP header is received. The frame was a broadcast.

Cause: Host sending packets for unknown proprietary protocol using SNAP.

SE.019

Level: UE-ERROR

Short Syntax: SE.019 SE- *se_id*:DECnet bad len *actual_length claimed_length*, 0x *source_MAC_address*->0x *destination_MAC_address* net- *network*

Long Syntax: SE.019 SE- *se_id*:DECnet packet received with a bad length actual *actual_length* claimed *claimed_length* from 0x *source_MAC_address* to 0x *destination_MAC_address* network *network*

Description: A DECnet packet was received with a length field that was larger than the actual length of the packet.

SE.020

Level: UI-ERROR

Short Syntax: SE.020 SE- *se_id*:LLC RSP to 0x *destination_Ethernet_address* dsc, rsn *code*, net- *network*

Long Syntax: SE.020 SE- *se_id*:LLC response to 0x *destination_Ethernet_address* discarded, for reason *code*, network *network*

Description: An LLC response (XID or TEST) could not be transmitted to the specified address, for the reason specified by code.

SE.021

Level: UE-ERROR

Short Syntax: SE.021 SE- *se_id*:Dropped IPX pkt w/ *encap_seen* encaps - using *encap_used* encaps on int *intnum*

Long Syntax: SE.021 SE- *se_id*:Dropped IPX pkt with encaps *encap_seen* using *encap_used* on interface *intnum*

Description: This message is generated when an IPX packet is recieved with an encapsulation other than that which has been selected for this interface

Cause: Normal for networks using multiple encapsulations on a single wire.

Action: None needed.

SE.022

Level: U-INFO

Short Syntax: SE.022 SE- *se_id*:Unkn SNAP mfr code *number* from 0x *source_MAC* net- *network ID*

Long Syntax: SE.022 SE- *se_id*:Unknown SNAP manufacturer code *number* from 0x *source_MAC* net *network ID*

Description: This message is generated when a frame with an unknown organization code (not 000000) in the SNAP header is received. The frame was addressed to the router.

Cause: Host sending packets for unknown proprietary protocol using SNAP.

SE.023

Level: U-INFO

Short Syntax: SE.023 SE- *se_id*:Unexp U-frame *LLC_control* from 0x *source_MAC* ssap *source_SAP* dsap *dest_SAP* net- *network ID*

Long Syntax: SE.023 SE- *se_id*:Unexpected U-frame *LLC_control* from 0x *source_MAC*, ssap *source_SAP*, dsap *dest_SAP*, net *network ID*

Description: This message is generated when an unexpected 802.2 LLC U (unnumbered) frame type is received. (Only UI, XID, and TEST are supported.) The frame was addressed to the router.

SE.024

Level: CE-ERROR

Short Syntax: SE.024 SE- *se_id*:Unexp U-frame *LLC_control* from 0x *source_MAC* ssap *source_SAP* dsap *dest_SAP* net- *network ID*

Long Syntax: SE.024 SE- *se_id*:Unexpected U-frame *LLC_control* from 0x *source_MAC*, ssap *source_SAP*, dsap *dest_SAP*, net *network ID*

Description: This message is generated when an unexpected 802.2 LLC U (unnumbered) frame type is received. (Only UI, XID, and TEST are supported.) The frame was a broadcast.

SE.025

Level: CI-ERROR

Short Syntax: SE.025 SE- *se_id*:Hello BPDU dropped because STP disabled on prt *port*, net- *network*

Long Syntax: SE.025 SE- *se_id*:Hello BPDU dropped because STP disabled on port *port*, network *network*

Description: A spanning tree Hello BPDU frame was received on a port that has been disabled for SuperELAN Spanning Tree participation. Manual spanning tree support has not been enabled for SuperELAN and therefore this message should never be displayed. If this message is displayed, this indicates a problem with the operational code.

SE.026

Level: C-INFO

Short Syntax: SE.026 SE- *se_id*:Frame dropped, src 0x *source_mac*==dest 0x *dest_mac*, port *port* net- *network* int *int/ int_no*

Long Syntax: SE.026 SE- *se_id*:Frame from 0x *source_mac* to 0x *dest_mac*, source same as destination, dropping, from port *port* network *network* interface *int/ int_no*

Description: Frames to and from the same address are not bridged.

SE.027

Level: UE-ERROR

Short Syntax: SE.027 SE- *se_id*:DECnet bad len *actual_length claimed_length*, 0x *source_Ethernet_address*->0x *destination_Ethernet_address* net- *network*

Long Syntax: SE.027 SE- *se_id*:DECnet packet received with a bad length actual *actual_length* claimed *claimed_length* from 0x *source_Ethernet_address*to 0x *destination_Ethernet_address* network *network*

Description: A DECnet packet was received with a length field that was larger than the actual length of the packet.

SE.028

Level: UE-ERROR

Short Syntax: SE.028 SE- *se_id*:DECnet MOP bad len *actual_length claimed_length*, 0x *source_Ethernet_address*->0x *destination_Ethernet_address* net- *network*

Long Syntax: SE.028 SE- *se_id*:DECnet MOP packet received with a bad length actual *actual_length* claimed *claimed_length* from 0x *source_Ethernet_address* to 0x *destination_Ethernet_address* network *network*

Description: A DECnet MOP packet was received with a length field that was larger than the actual length of the packet.

SE.029

Level: UE-ERROR

Short Syntax: SE.029 SE- *se_id*:Unexp *type* bcast frame *LLC_control* from 0x *source_MAC* ssap *source_SAP* dsap *dest_SAP* net- *network ID*

Long Syntax: SE.029 SE- *se_id*:Unexpected *type* broadcast frame *LLC_control* from 0x *source_MAC*, ssap *source_SAP*, dsap *dest_SAP*, net *network ID*

Description: This message is generated when an unexpected 802.2 LLC frame type is received. Type may be I (information transfer) or S (supervisory). The frame was a broadcast.

Cause: Host attempting to make 802.2 type 2 connection to router.

SE.030

Level: UE-ERROR

Short Syntax: SE.030 SE- *se_id*:IPX pkt in *received_encapsulation* encap ign, using *configured_encapsulation* encaps, net- *network*

Long Syntax: SE.030 SE- *se_id*:IPX pkt in encapsulation *received_encapsulation* ignored, using encapsulation *configured_encapsulation* on network *network*

Description: This message is generated when an IPX packet is received in a data-link encapsulation (frame) other than the one configured for IPX on this interface. The packet will be ignored. The received_encapsulation and configured_encapsulation are one of "ETHERNET_802.3", "ETHERNET_II", "ETHERNET_802.2", or "ETHERNET_SNAP". ETHERNET_802.3 is also known as "Novell", and ETHERNET_II is also known as "Ethernet".

Cause: If only one encapsulation is being used on this network, this node's encapsulation is not the same as all other IPX nodes on the network.

Action: Configure all nodes on network to use same encapsulation.

Cause: If multiple encapsulations are being used on this network, a packet has been received from a node using an encapsulation different from this node.

SE.031

Level: U-INFO

Short Syntax: SE.031 SE- *se_id*:Unexp *type* frame

LLC_control from 0x *source_MAC* ssap *source_SAP* dsap *dest_SAP* net- *network ID*

Long Syntax: SE.031 SE- *se_id*:Unexpected *type* frame *LLC_control* from 0x *source_MAC*, ssap *source_SAP*, dsap *dest_SAP*, net *network ID*

Description: This message is generated when an unexpected 802.2 LLC frame type is received. Type may be I (information transfer) or S (supervisory). The frame was addressed to the router.

Cause: Host attempting to make 802.2 type 2 connection to router.

SE.032

Level: UI-ERROR

Short Syntax: SE.032 SE- *se_id*:SR 0x *source_mac*->0x *dest_mac* send failed, rsn *reason_code*, port *port* net- *network* int *int/ int_no*

Long Syntax: SE.032 SE- *se_id*:Sending source routed frame from 0x *source_mac* to 0x *dest_mac* failed, reason *reason_code*, on port *port* network *network* interface *int/ int_no*

Description: The sending of a source routed frame failed. The reason_code is the internal error code for the failure.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network_name.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

SE.033

Level: UI-ERROR

Short Syntax: SE.033 SE- *se_id*:LOOP fwd to 0x *forward_Ethernet_address* dropped, rsn *code*, net- *network*

Long Syntax: SE.033 SE- *se_id*:Loopback protocol, forward to 0x *forward_Ethernet_address* discarded, for reason *code*, network *network*

Description: A Ethernet Loopback Protocol (Configuration Testing Protocol) packet could not be forwarded to the specified address, for the reason specified by code.

SE.034

Level: U-INFO

Short Syntax: SE.034 SE- *se_id*:Unkn SNAP type *type_code* from 0x *source_MAC* net- *network ID*

Long Syntax: SE.034 SE- *se_id*:Unknown SNAP type *type_code* from 0x *source_MAC* net *network ID*

Description: This message is generated when a frame with an unknown SNAP type (within organization code 000000) is received. The frame was addressed to the router.

Cause: Host sending packets for unknown Ethernet type using SNAP.

SE.035

Level: CE-ERROR

Short Syntax: SE.035 SE- *se_id*:Unkn SNAP type *type_code* from 0x *source_MAC* net- *network ID*

Long Syntax: SE.035 SE- *se_id*:Unknown SNAP type *type_code* from 0x *source_MAC* net *network ID*

Description: This message is generated when a frame with an unknown SNAP type (within organization code 000000) is received. The frame was a broadcast.

Cause: Host sending packets for unknown Ethernet type using SNAP.

SE.036

Level: U-INFO

Short Syntax: SE.036 SE- *se_id*:Unkn SAP *sap_number* from 0x *source_MAC* net- *network ID*

Long Syntax: SE.036 SE- *se_id*:Unknown SAP *sap_number* from 0x *source_MAC* net *network ID*

Description: This message is generated when a frame with an unknown destination SAP is received. The message was addressed to the router.

Cause: Host sending packets for unknown protocol identifier (SAP).

SE.037

Level: U-INFO

Short Syntax: SE.037 SE- *se_id*:Unkn SAP *sap_number* from 0x *source_MAC* net- *network ID*

Long Syntax: SE.037 SE- *se_id*:Unknown SAP *sap_number* from 0x *source_MAC* net *network ID*

Description: This message is generated when a frame with an unknown destination SAP is received. The message was a broadcast.

Cause: Host sending packets for unknown protocol identifier (SAP).

SE.038

Level: C-INFO

Short Syntax: SE.038 SE- *se_id*:Main pkt rcvd on net- *network*

Long Syntax: SE.038 SE- *se_id*:Maintenance packet received on net *network*

Description: The handler received a maintenance packet.

SE.039

Level: CI-ERROR

Short Syntax: SE.039 SE- *se_id*:0x *source_mac*->0x *dest_mac* dropped, dest addr filt, port *port* net- *network* int *int/ int_no*

Long Syntax: SE.039 SE- *se_id*:Frame from 0x *source_mac* to 0x *dest_mac* dropped, destination address filtered, port *port* network *network* interface *int/ int_no*

Description: A MAC frame has been received by the hardware, but is being dropped because the destination MAC address is being administratively filtered by the bridge. The frame will be dropped.

Cause: Receipt of frame whose destination MAC address matches the exclusive filter.

SE.040

Level: C-INFO

Short Syntax: SE.040 SE- *se_id*:Test pkt 0x *mac_address,* src sap *source_sap,* net- *network*

Long Syntax: SE.040 SE- *se_id*:Test packet from 0x *mac_address,* source sap *source_sap,* net *network*

Description: The handler received a test message.

SE.041

Level: C-INFO

Short Syntax: SE.041 SE- *se_id*:XID pkt 0x *mac_address,* sap *source_sap,* net- *network*

Long Syntax: SE.041 SE- *se_id*:XID packet received from 0x *mac_address,* source sap *source_sap,* net *network*

Description: The handler received an xid message.

SE.042

Level: UE-ERROR

Short Syntax: SE.042 SE- *se_id*:LOOP mcast fwd dest 0x *forward_Ethernet_address*, 0x *source_Ethernet_address*->0x *destination_Ethernet_address*, net- *network*

Long Syntax: SE.042 SE- *se_id*:Loopback Protocol, multicast forward address 0x *forward_Ethernet_address* from 0x *source_Ethernet_address* to 0x *destination_Ethernet_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet has a forward address that is a multicast. It will be discarded.

Cause: Programming error in remote node.

SE.043

Level: C_INFO

Short Syntax: SE.043 SE- *se_id*:LOOP fwd 0x *source_Ethernet_address*->0x *forward_Ethernet_address*, net- *network*

Long Syntax: SE.043 SE- *se_id*:Loopback Protocol, forwarding from 0x *source_Ethernet_address* to 0x *forward_Ethernet_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet is being forwarded to the specified next hop.

SE.044

Level: C-INFO

Short Syntax: SE.044 SE- *se_id*:SRF rcv (RIF *RIF*) 0x *source_mac*->0x *dest_mac* from port *port*, net- *network* int *int/ int_no*

Long Syntax: SE.044 SE- *se_id*:Specifically-routed frame received (RIF *RIF*) from 0x *source_mac* to 0x *dest_mac* from port *port*, network *network* interface *int/ int_no*

Description: A Specifically-routed frame has been received on the specified port.

SE.045

Level: C-INFO

Short Syntax: SE.045 SE- *se_id*:Send SRF (RIF *RIF*) 0x *source_mac*->0x *dest_mac* to port *port*, net- *network* int *int/ int_no*

Long Syntax: SE.045 SE- *se_id*:Sending Specifically-routed frame (RIF *RIF*) from 0x *source_mac* to 0x *dest_mac* to port *port*, network *network* interface *int/ int_no*

Description: A Specifically-routed frame is being sent on the specified port.

SE.046

Level: UE-ERROR

Short Syntax: SE.046 SE- *se_id*:LOOP func *function* not fwrd, 0x *source_Ethernet_address*->0x *destination_Ethernet_address*, net- *network*

Long Syntax: SE.046 SE- *se_id*:Loopback Protocol, function *function* not Forward Data from 0x

source_Ethernet_address to 0x *destination_Ethernet_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet did not have a function code of forward (2). It will be discarded.

Cause: Function code was reply (1), because we were the ultimate destination of this packet.

Action: None.

Cause: Undefined function code, due to programming error in remote node.

SE.047

Level: DEBUG

Short Syntax: SE.047 SE- *se_id*:Attempt to learn MAC addr 0x *mac* for our own net- *network* int *int/ intno*

Long Syntax: SE.047 SE- *se_id*:Attempted to learn MAC address 0x *mac* for our own net- *network* int *int/ intno*

Description: Forwarding code tried to cache its own interfaces MAC address. This address must not be dynamically learned. This can occur if there is another MAC address in the network matching the SE interface MAC or an SE BPDU is being looped back to the sending interface.

SE.048

Level: C-INFO

Short Syntax: SE.048 SE- *se_id*:message event not used

Long Syntax: SE.048 SE- *se_id*:message event not used

Description: This message event is not used.

SE.049

Level: C-INFO

Short Syntax: SE.049 SE- *se_id*:message event not used

Long Syntax: SE.049 SE- *se_id*:message event not used

Description: This message event is not used.

SE.050

Level: C-INFO

Short Syntax: SE.050 SE- *se_id*:LOOP rcv 0x *source_Ethernet_address*->0x *destination_Ethernet_address*, net- *network*

Long Syntax: SE.050 SE- *se_id*:Loopback Protocol frame received from 0x *source_Ethernet_address* to 0x *destination_Ethernet_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet was received.

SE.051

Level: UE-ERROR

Short Syntax: SE.051 SE- *se_id*:SRF 0x *source_mac*->0x *dest_mac* too big (*reformatted_length* > *output_maximum*) for port *port* net- *network* int *int/ int_no*, dropped

Long Syntax: SE.051 SE- *se_id*:Specifically-routed frame from 0x *source_mac* to 0x *dest_mac* is too big (reformatted length *reformatted_length* > output maximum size *output_maximum*) for port *port* network *network* interface *int/ int_no*, dropped

Description: The specified Specifically-routed (source-routed) frame is too large to send on this outgoing port and network. The reformatted_length is the size of the frame including MAC headers after any mapping of data link headers.

Cause: Host not honoring LF bit values from its returned explorer frames.

Action: Fix host.

SE.052

Level: C-INFO

Short Syntax: SE.052 SE- *se_id*:ELAN ' *src_elan'*/net- *src_netno* -> ELAN ' *targ_elan'*/net- *targ_netno frame_type* drp due to VLAN defs

Long Syntax: SE.052 SE- *se_id*:ELAN ' *src_elan'*/net- *src_netno* -> ELAN ' *targ_elan'*/net- *targ_netno frame_type* drp due to VLAN defs

Description: LE Control frame was was not transmitted onto specified interface due to VLAN filter.

SE.053

Level: UE-ERROR

Short Syntax: SE.053 SE- *se_id*:Inv RIF len *RIF_length*, 0x *source_mac*->0x *dest_mac* port *port*, net- *network ID*, dropped

Long Syntax: SE.053 SE- *se_id*:Frame with invalid RIF length *RIF_length* from 0x *source_mac* to 0x *dest_mac* from port *port*, network *network ID*, discarded

Description: A source-routing frame was received with an invalid RIF length encoded in the Length bits of the RIF.

Cause: Received frame with RIF length less than 2 or not a multiple of 2 in length.

Action: Correct software in sending node.

SE.054

Level: UE-ERROR

Short Syntax: SE.054 SE- *se_id*:LOOP odd skip *count*, 0x *source_Ethernet_address*->0x

destination_Ethernet_address, net- *network*

Long Syntax: SE.054 SE- *se_id*:Loopback Protocol, odd skipCount *count* from 0x *source_Ethernet_address* to 0x *destination_Ethernet_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet had an odd skipCount in the packet. It will be discarded.

Cause: Programming error on remote node.

SE.055

Level: UE-ERROR

Short Syntax: SE.055 SE- *se_id*:LOOP bd skip *count*, 0x *source_Ethernet_address*->0x *destination_Ethernet_address*, net- *network*

Long Syntax: SE.055 SE- *se_id*:Loopback Protocol, bad skipCount *count* from 0x *source_Ethernet_address* to 0x *destination_Ethernet_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet had a skipCount in the packet that points to beyond the end of the packet. It will be discarded.

Cause: Programming error on remote node.

SE.056

Level: C-INFO

Short Syntax: SE.056 SE- *se_id*:0x *source_mac*->0x *dest_mac* dropped, port block/list, net- *network*

Long Syntax: SE.056 SE- *se_id*:Frame from 0x *source_mac* to 0x *dest_mac* dropped, received on blocked or listening port, network *network*

Description: A MAC frame has been received by the hardware, but is being dropped because the port it was received on is in the "blocking" or "listening" state. Frames are only processed when the port is in the "learning" or "forwarding" state.

Cause: Normal on port bringup.

Action: Wait for port to transition to "learning" and "forwarding" states.

SE.057

Level: C-INFO

Short Syntax: SE.057 SE- *se_id*:Chg state *old_state* to *new_state*, port *port* net- *network*

Long Syntax: SE.057 SE- *se_id*:Changing port state from *old_state* to *new_state* for port *port*, network *network*

Description: The Spanning Tree Protocol has requested this state change for this port in the SRT bridge. The old_state and new_state are one of: FORWARDING (Spanning Tree Protocol Forwarding state), LEARNING (Spanning Tree Protocol Learning state), LISTENING (Spanning Tree Protocol Listening state), BLOCKED (Spanning Tree Protocol Blocking state), CONFIGURING (configuration of port device pending), POSTCONFIGURING (configuration of port device done), PRECONFIGURING (port enabled, configuration of port device to start), and DISABLED (port disabled).

SE.058

Level: P-TRACE

Short Syntax: SE.058 SE- *se_id*:Trace incoming data frame from ELAN ' *elan*'

Long Syntax: SE.058 SE- *se_id*:Trace incoming data frame from ELAN ' *elan*'

Description: Trace incoming data frame.

SE.059

Level: P-TRACE

Short Syntax: SE.059 SE- *se_id*:Trace outgoing data frame to ELAN ' *elan*'

Long Syntax: SE.059 SE- *se_id*:Trace outgoing data frame to ELAN ' *elan*'

Description: Trace outgoing data frame.

SE.060

Level: DEBUG

Short Syntax: SE.060 SE- *se_id*:MAC 0x *mac* found in DFFC mapped to ELAN ' *elan*', net- *network* int *int/ int_no*

Long Syntax: SE.060 SE- *se_id*:MAC 0x *mac* found in DFFC mapped to ELAN ' *elan*', net- *network* int *int/ int_no*

Description: Successful search of DFFC database for MAC address.

SE.061

Level: DEBUG

Short Syntax: SE.061 SE- *se_id*:Rte Desc *ring*. *bridge* found in DFFC mapped to ELAN ' *elan*', net- *network* int *int/ int_no*

Long Syntax: SE.061 SE- *se_id*:Route Descriptor *ring*. *bridge* found in DFFC mapped to ELAN ' *elan*', net- *network* int *int/ int_no*

Description: Successful search of DFFC database for route descriptor.

SE.062

Level: C-INFO

Short Syntax: SE.062 SE- *se_id*:Chg state *old_state* to *new_state*, port *port* net- *network*

Long Syntax: SE.062 SE- *se_id*:Changing port state from *old_state* to *new_state* for port *port*, network *network*

Description: The SE Spanning Tree Protocol has requested this state change for this port in the SE bridge. The old_state and new_state are one of: FORWARDING (Spanning Tree Protocol Forwarding state), LEARNING (Spanning Tree Protocol Learning state), LISTENING (Spanning Tree Protocol Listening state), BLOCKED (Spanning Tree Protocol Blocking state), CONFIGURED (port is configured, waiting for SPT), NETDOWN (port is configured, but interface is down), CONFIGURING (port is configuring), and NOTSETUP (port not configured).

SE.063

Level: C_INFO

Short Syntax: SE.063 SE- *se_id*:0x *source_mac*->0x *dest_mac* SNAP *protocol_identifier*, endnode, net- *network*

Long Syntax: SE.063 SE- *se_id*:Frame from 0x *source_mac* to 0x *dest_mac*, IEEE 802 SNAP Protocol Identifier *protocol_identifier* for endnode protocol, network *network*

Description: A multicast frame has been received for the IEEE 802 Subnetwork Access Protocol (SNAP) Protocol Identifier (PID) which corresponds with an endnode protocol. The frame will be both bridged and locally processed by the endnode protocol.

SE.064

Level: CI_ERROR

Short Syntax: SE.064 SE- *se_id*:DFFC MAC Cache exceeded, 0x *mac* deleted from cache

Long Syntax: SE.064 SE- *se_id*:DFFC MAC Cache exceeded, 0x *mac* deleted from cache

Description: The DFFC MAC Cache has exceeded its configured maximum size. The oldest MAC entry is deleted.

Action: No action is required. The size of the MAC Cache may be increased if there is available free heap memory. The amount of free heap memory can be determined by issuing the mem command from the talk 5 interface console.

SE.065

Level: UI_ERROR

Short Syntax: SE.065 SE- *se_id*:No memory available for allocating cache entry

Long Syntax: SE.065 SE- *se_id*:No memory available for allocating cache entry

Description: A CFFC or DFFC entry could not be allocated due to insufficient memory.

Action: The size of the DFFC MAC Cache may need to be decreased to prevent excess memory heap allocation. Other component configurations should also be checked for inappropiately high parameter settings which could result in waisted memory allocation.

SE.066

Level: CE_ERROR

Short Syntax: SE.066 SE- *se_id*:Bridges connected to SE have diff cnfg ring num, 0x *ring_new* replaces 0x *ring_old*.

Long Syntax: SE.066 SE- *se_id*:Bridges connected to SE have diff cnfg ring number, 0x *ring_new* replaces 0x *ring_old*.

Description: Bridges attached to the SuperELAN are configured with different ring numbers. The SE Cache assumes the latest ring number learned is the correct and current ring number value. The route cache is flushed.

Action: Change the ring number setting on attached bridges to make sure that all bridges directly connected to the SuperELAN are configured with same ring number. Failing to due so causes the SuperELAN to operate less efficiently, but should not effect network connectivity.

SE.067

Level: CI-ERROR

Short Syntax: SE.067 SE- *se_id*:Ctrl q ovf, dropped, net- *network* int *int/ int_no*

Long Syntax: SE.067 SE- *se_id*:Control frame queue overflow, packet dropped from network *network* interface *int/ int_no*

Description: The queue for control frames to be forwarded is too long, and this frame has been dropped to attempt to alleviate the congestion.

SE.068

Level: CE-ERROR

Short Syntax: SE.068 SE- *se_id*:ELAN ' *elan*'/net- *netno* unjoined LES/BUS due to frame sz mismatch

Long Syntax: SE.068 SE- *se_id*:ELAN ' *elan*'/net- *netno*

unjoined LES/BUS due to frame size mismatch

Description: The ELAN successfully joined a LES/BUS with a different frame size than the SuperELAN frame size. The short-cut LEC was removed from the LES/BUS and cannot participate in the SuperELAN.

Action: Change the LES/BUS frame size to which the short-cut LEC was attempting to join or reconfigure the SuperELAN with a different frame size.

SE.069

Level: CI-ERROR

Short Syntax: SE.069 SE- *se_id*:ELAN ' *elan*'/net- *netno* no global bufs, LE Ctrl frame droppe.

Long Syntax: SE.069 SE- *se_id*:ELAN ' *elan*'/net- *netno* no global buffers, LE Ctrl frame dropped

Description: No global buffers were available to copy the LE Control Frame. The frame was dropped by the SuperELAN bridge.

SE.070

Level: CI-ERROR

Short Syntax: SE.070 SE- *se_id*:ELAN ' *elan*'/net- *netno* no mem available to cache LE Ctrl frame

Long Syntax: SE.070 SE- *se_id*:ELAN ' *elan*'/net- *netno* no memory available to cache LE Ctrl frame

Description: No memory available to cache the LE Control Frame. The frame was dropped by the SuperELAN bridge.

SE.071

Level: C-INFO

Short Syntax: SE.071 SE- *se_id*:ELAN ' *elan*'/net- *netno* LE Ctrl frame filt, Target MAC=0x *target_mac*

Long Syntax: SE.071 SE- *se_id*:ELAN ' *elan*'/net- *netno* LE Ctrl frame filtered, Target MAC=0x *target_mac*

Description: LE ARP Request filtered.

SE.072

Level: C-INFO

Short Syntax: SE.072 SE- *se_id*:ELAN ' *elan*'/net- *netno* LE Ctrl frame dropped, port not fwd

Long Syntax: SE.072 SE- *se_id*:ELAN ' *elan*'/net- *netno* LE Ctrl frame dropped, port not forwarding

Description: Port was not in forwarding state, the LE Control frame was discarded.

SE.073

Level: P-TRACE

Short Syntax: SE.073 SE- *se_id*:Trace incoming LE Control frame from ELAN ' *elan*'

Long Syntax: SE.073 SE- *se_id*:Trace incoming LE Control frame from ELAN ' *elan*'

Description: Trace incoming LE Control frame.

SE.074

Level: P-TRACE

Short Syntax: SE.074 SE- *se_id*:Trace outgoing LE Control frame to ELAN ' *elan*'

Long Syntax: SE.074 SE- *se_id*:Trace outgoing LE Control frame to ELAN ' *elan*'

Description: Trace outgoing LE Control frame.

SE.075

Level: C-INFO

Short Syntax: SE.075 SE- *se_id*:New targ route desc *ring*. *bridge* lrnd on ELAN ' *elan*'/net- *net* int *int*/ *intno*

Long Syntax: SE.075 SE- *se_id*:New target route descriptor *ring*. *bridge* learned on ELAN ' *elan*'/net- *net* int *int*/ *intno*

Description: A net target route descriptor was learned. The route descriptor represents a next bridge-ring segment attached to the SuperELAN via a source route bridge.

SE.076

Level: C-INFO

Short Syntax: SE.076 SE- *se_id*:Target route desc *ring*. *bridge* deleted from DFFC

Long Syntax: SE.076 SE- *se_id*:Target route desc *ring*. *bridge* deleted from DFFC

Description: A target route descriptor was removed from the DFFC. This can occur if the DFFC is flushed or the route descriptor was aged-out.

SE.077

Level: C-INFO

Short Syntax: SE.077 SE- *se_id*:New MAC addr 0x *mac* learned on ELAN ' *elan*'/net- *net* int *int*/ *intno*

Long Syntax: SE.077 SE- *se_id*:New MAC addr 0x *mac* learned on ELAN ' *elan*'/net- *net* int *int*/ *intno*

Description: A new MAC address was learned. This MAC address may represent a station directly attached to the ELAN in the SuperELAN or a legacy station behind a transparent bridge.

SE.078

Level: C-INFO

Short Syntax: SE.078 SE- *se_id*:MAC addr 0x *mac* deleted from DFFC

Long Syntax: SE.078 SE- *se_id*:MAC addr 0x *mac* deleted from DFFC

Description: A MAC address was removed from the DFFC. This can occur if the DFFC is flushed or the MAC address was aged-out.

SE.079

Level: C-INFO

Short Syntax: SE.079 SE- *se_id*:Dup MAC 0x *mac* found in DFFC

Long Syntax: SE.079 SE- *se_id*:Dup MAC 0x *mac* found in DFFC

Description: Search of DFFC database for a MAC address resulted in a duplicate MAC match.

SE.080

Level: C-INFO

Short Syntax: SE.080 SE- *se_id*:Dup MAC 0x *mac* detected on ELAN ' *elan*', net- *network* int *int/ int_no*

Long Syntax: SE.080 SE- *se_id*:Duplicate MAC 0x *mac* detected on ELAN ' *elan*', net- *network* int *int/ int_no*

Description: A duplicate MAC was detected. MAC added to ELAN on which it was learned.

SE.081

Level: UE-ERROR

Short Syntax: SE.081 DIX V2 bd ln *actual_length claimed_length source_Ethernet_address* -> *destination_Ethernet_address* nt *network*

Long Syntax: SE.081 packet received with a bad length field actual *actual_length* claimed *claimed_length* from *source_Ethernet_address* to *destination_Ethernet_address* network *network*

Description: A DIX V2 packet was received with a type field of 0x80D5 (IBM SNA or Netbios) which had a length field that was larger than the actual length of the packet.

Fatal seiisrt

Short Syntax: SE: Invalid i_srt on input

Description: The i_srt flag passed from the handler to forwarder has an invalid value.

Cause: Software bug.

Action: Take a crash dump and contact customer service.

Fatal seuimed

Short Syntax: SE: unknown input media

Description: The input net type is not one of the ones understood by the SRT bridge (802.3/Ethernet, FDDI, or 802.5).

Cause: Software bug.

Action: Take a crash dump and contact customer service.

Chapter 110. SuperELAN Spanning Tree Protocol (SEST)

This chapter describes SuperELAN Spanning Tree Protocol (SEST) messages. For information on message content and how to use the message, refer to the Introduction.

SEST.001

Level: C-INFO

Short Syntax: SEST.001 Cfg BPDU rcv frame *source_address bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.001 Configuration BPDU received frm *source_address* on *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*

Description: A configuration BPDU has been received from the specified MAC address.

Cause: Another SE bridge on the same network as this bridge on this port.

SEST.002

Level: C-INFO

Short Syntax: SEST.002 Tcn BPDU rcv frame *source_address bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.002 Topology change notification BPDU received frame *source_address* on *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*

Description: A topology change notification BPDU has been received from the specified MAC address.

Cause: Topology change has been detected at or downstream of the sending bridge.

Action: None needed, the message should stop when the topology change is acknowledged by the root bridge.

SEST.003

Level: UE-ERROR

Short Syntax: SEST.003 Ukn BPDU type *BDPU_type* rcv frame *source_address bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.003 Unkown BPDU type *BDPU_type* received frame *source_address* on *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*

Description: A BPDU with an undefined value in the BPDU Type field was received from the specified host. It will be ignored.

Cause: Programming error at remote bridge.

Action: Correct remote node.

Cause: Data corruption in received packet.

Action: Eliminate source of data corruption.

SEST.004

Level: UE-ERROR

Short Syntax: SEST.004 BPDU bad ID *Protocol_Identifier* frame *source_address bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.004 BPDU bad protocol identifier *Protocol_Identifier* frame *source_address* on *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*

Description: A configuration BPDU has been received with a Protocol Identifier that is not 0000. It will be ignored.

Cause: Programming error at remote bridge.

Action: Correct remote node.

Cause: Data corruption in received packet.

Action: Eliminate source of data corruption.

SEST.005

Level: UE-ERROR

Short Syntax: SEST.005 BPDU bad ver *Protocol_Version_Identifier* frame *source_address bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.005 BPDU bad Version *Protocol_Version_Identifier* frame *source_address* on *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*

Description: A configuration BPDU has been received with a Protocol Version Identifier that is not 00. It will be ignored.

Cause: Programming error at remote bridge.

Action: Correct remote node.

Cause: Data corruption in received packet.

Action: Eliminate source of data corruption.

SEST.006

Level: UE-ERROR

Short Syntax: SEST.006 Cfg BPDU trunc (*length* byt) frame *source_address bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.006 Configuration BPDU tuncated (*length* bytes) frame *source_address* on *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*

Description: A configuration BPDU has been received which is not the proper bytes in length. It will be ignored.

Cause: Programming error at remote bridge.

Action: Correct remote node.

Cause: Data corruption in received packet.

Action: Eliminate source of data corruption.

SEST.007

Level: UE-ERROR

Short Syntax: SEST.007 Cfg BPDU unk flg *flags* frame *source_address bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.007 Configuration BPDU unknown flags *flags* frame *source_address* on *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*

Description: A configuration BPDU has been received which has undefined bits set in the flags field. It will be ignored.

Cause: Programming error at remote bridge.

Action: Correct remote node.

Cause: Data corruption in received packet.

Action: Eliminate source of data corruption.

SEST.008

Level: UE-ERROR

Short Syntax: SEST.008 Tcn BPDU trunc (*length* byt) frame *source_address bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.008 Topology change notification BPDU tuncated (*length* bytes) frame *source_address* on *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*

Description: A topology change notificaton BPDU has been received that is not the proper bytes in length. It will be ignored.

Cause: Programming error at remote bridge.

Action: Correct remote node.

Cause: Data corruption in received packet.

Action: Eliminate source of data corruption.

SEST.009

Level: UI-ERROR

Short Syntax: SEST.009 No buf for BPDU *bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.009 No buffer to send BDPU on *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*

Description: No packet buffer was available to construct and send a BDPU on the specified port.

Cause: Severe packet buffer shortage.

Action: Check memory statistics in GWCON to verify packet buffer level.

Cause: Traffic peak using all available buffers.

Action: This is the problem if this message occurs infrequently.

SEST.010

Level: C-INFO

Short Syntax: SEST.010 Sndg cfg BPDU *bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.010 Sending Configuration BPDU on *bridge_type- se_id* port *bridge_port* network *network* interface *int/ int_no*

Description: A Configuration BPDU will be sent on the specified port. This is done normally on a periodic basis as part of the SE spanning tree protocol. The flags field in this BPDU is zero, e.g., neither the Topology Change or the Topology Change Acknowledgement bits are set.

SEST.011

Level: C-INFO

Short Syntax: SEST.011 Sndg Cfg BPDU flgs *TC TCA bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.011 Sending Configuration BPDU with flags *TC TCA* on *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*

Description: A Configuration BPDU will be sent on the specified port. This is done normally on a periodic basis as part of the SE spanning tree protocol. TC will be displayed if the Topology Change bit is set in the Flags byte of the BPDU, TCA will be displayed if the Topology Change Acknowledge bit is set in the flags byte.

Cause: The Topology Change flag is set if this bridge is the root and it knows that there is a topology change in process. Also, non-root bridges propogate this bit

received in incoming Configuration BPDUs.

Action: None needed, this flag will be set only for the sum of the current maximum age and current forward delay parameters (as propagated by the root bridge).

Cause: The Topology Change Acknowledge flag is set if this bridge has received a Topology Change Notification BPDU, and this port is the Designated Bridge on its LAN.

Action: None needed, this flag will only be sent on one BDPU.

SEST.012

Level: C-INFO

Short Syntax: SEST.012 Sndg tcn BPDU *bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.012 Sending Topology Change Notification BPDU on *bridge_type- se_id* port *bridge_port* network *network* interface *int/ int_no*

Description: A Topology Change Notification BPDU will be sent on the specified port. These are sent on the root port of non-root ports when they detect a topology change in the spanning tree.

Cause: A bridge, or an interface on a bridge, has gone up or down in this spanning tree.

Action: None needed. This state persists only until a topology change acknowledgement is received, or a timeout that indicates that the old root bridge is no longer reachable.

SEST.013

Level: UI-ERROR

Short Syntax: SEST.013 BPDU snd failed, rsn *reason_code*, *bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.013 BPDU send failed for reason code *reason_code* on *bridge_type- se_id* port *bridge_port* network *network* interface *int/ int_no*

Description: The attempt to queue a BPDU for transmission on the specified port failed.

Cause: Miscellaneous handler error. (Reason code 1.)

Action: Check for error messages from handler for network.

Cause: Output queue overflow, or other flow control. (Reason code 2.)

Action: Alleviate congestion.

Cause: Network down. (Reason code 3.)

Action: See why handler thinks network is down.

Cause: Dropped by handler to avoid looping, or bad broadcast. (Reason code 4.)

Action: Check configuration.

Cause: Host down. (Reason code 5.)

Action: See why handler thinks host is down.

SEST.014

Level: U-INFO

Short Syntax: SEST.014 Blocking *bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*, det topol chg

Long Syntax: SEST.014 Blocking *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*, detecting topology change

Description: This port has just been placed in Blocking state. This is a change in the topology, so this bridge detects a topology change. This will in turn cause topology change notifications to be sent.

Cause: A bridge, or an interface on a bridge, has gone up or down in this spanning tree.

Action: None needed. This is normal when there are changes.

SEST.015

Level: U-INFO

Short Syntax: SEST.015 Topol chg detected *bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.015 Topology change detected on *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*

Description: A topology change notification has been received on this port, and this port is the designated port on its LAN. This causes the protocol to enter topology change notification state. The topology change will be acknowledged towards the sender, and propagated towards the root.

Cause: A bridge, or an interface on a bridge, has gone up or down in this Spanning Tree.

Action: None needed. This is normal when there are changes.

SEST.016

Level: U-INFO

Short Syntax: SEST.016 Select as root *bridge_type- se_id*, det topol chg

Long Syntax: SEST.016 Selected as root on *bridge_type- se_id*, detecting topology change

Description: This bridge has just selected itself as the root of the spanning tree when it previously had not been. This causes the bridge to enter topology change notification state.

Cause: A bridge, or an interface on a bridge, has gone

up or down in this spanning tree.

Action: None needed. This is normal when there are changes.

Cause: This is the first bridge up, thus it is the root of the tree.

SEST.017

Level: C-INFO

Short Syntax: SEST.017 Tply chg ackd *bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.017 Topology change acknowledged on *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*

Description: A topology change acknowledgement has been detected on the specified port. This port is the root port of the bridge.

Cause: Bridge on same LAN as our root port has set topology change acknowledgement flag in outgoing Configuration BDPU. This was in response to a topology change notification that this bridge originated or propagated.

Action: None needed. This is the normal conclusion of topology change notification.

SEST.018

Level: C-INFO

Short Syntax: SEST.018 Acking tply chg *bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.018 Acknowledging topology change on *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*

Description: A topology change notification is being acknowledged on the specified port. This is done when a topology change notification is received on a port that is the designated port for that LAN.

Cause: Change on bridge topology downstream of this bridge.

Action: None needed. This is a normal part of reconfiguration of the spanning tree.

SEST.019

Level: C-INFO

Short Syntax: SEST.019 Tplgy chg notif timer expired *bridge_type- se_id*

Long Syntax: SEST.019 Topology Change Notification timer expired on *bridge_type- se_id*

Description: The Topology Change timer expired. This bridge will cease sending topology change notification BPDU's on its root port.

Cause: This timer expires when the bridge has been in Topology Change Notification state for the bridge hello timer period.

Action: None needed, this is the normal conclusion of this state.

SEST.020

Level: C-INFO

Short Syntax: SEST.020 Tplgy chg timer expired *bridge_type- se_id*

Long Syntax: SEST.020 Topology Change timer expired on *bridge_type- se_id*

Description: The Topology Change timer expired. This bridge, which is the root, will cease sending the Topology Change in its Configuration BPDUs.

Cause: This happens when this root bridge has been in Topology Change state for the sum of current maximum age and current forward delay parameters.

Action: None needed, this is the normal conclusion of this state.

SEST.021

Level: U-INFO

Short Syntax: SEST.021 Msg age timer exp *bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*, try Root

Long Syntax: SEST.021 Message age timer expired on *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*, will try and become root

Description: The message age timer has expired on this port. The bridge will attempt to become the root. It will become the designated port on that LAN.

Cause: No Configuration BPDU's being received on this interface. Either there are no bridges on this LAN, or they are down.

SEST.022

Level: C-INFO

Short Syntax: SEST.022 Hello timer exp *bridge_type- se_id*

Long Syntax: SEST.022 Hello timer expired on *bridge_type- se_id*

Description: The hello timer has expired on this port. Configuration BPDUs will be sent on all ports.

SEST.023

Level: C-INFO

Short Syntax: SEST.023 Stop msg age timer

bridge_type- se_id port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.023 Stopping message age timer for *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*

Description: Stopping the message age timer on this port because is it the designated port on its LAN.

SEST.024

Level: U-INFO

Short Syntax: SEST.024 Not root *bridge_type- se_id*, stop hello timer

Long Syntax: SEST.024 Not root anymore on *bridge_type- se_id*, stopping hello timer

Description: This bridge has just decided that it is no longer the root bridge of the spanning tree. The hello timer will also be cancelled.

SEST.025

Level: C-INFO

Short Syntax: SEST.025 Stop tplgy chg age timer *bridge_type- se_id*

Long Syntax: SEST.025 Stopping topology change timer for *bridge_type- se_id*

Description: Stopping the topology change timer because this bridge is no longer the root.

SEST.026

Level: U-INFO

Short Syntax: SEST.026 Root *bridge_type- se_id*, start hello timer

Long Syntax: SEST.026 Selected as root on *bridge_type- se_id*, starting hello timer

Description: This bridge has just decided that it is the root bridge of the spanning tree. The hello timer will be started.

SEST.027

Level: C-INFO

Short Syntax: SEST.027 Strt msg age timer *bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.027 Starting message age timer for *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*

Description: Starting the message age timer on this port.

SEST.028

Level: C-INFO

Short Syntax: SEST.028 Attmpt root *bridge_type- se_id*, strt hello timer

Long Syntax: SEST.028 Attempting to become root on *bridge_type- se_id*, starting hello timer

Description: This bridge is attempting to become the root bridge of the spanning tree. The hello timer will be started.

SEST.029

Level: UI-ERROR

Short Syntax: SEST.029 Cfg BPDU frame *source_address* ignored *bridge_type- se_id*, inact port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.029 Configuration BPDU from *source_address* on *bridge_type- se_id* ignored, inactive port *bridge_port*, network *network* interface *int/ int_no*

Description: A configuration BPDU has been received from the specified MAC address, but the port is not participating in the spanning tree protocol.

SEST.030

Level: UI-ERROR

Short Syntax: SEST.030 Tcn BPDU frame *source_address* ign *bridge_type- se_id*, inact port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.030 Topology change notification BPDU from *source_address* on *bridge_type- se_id* ignored, inactive port *bridge_port*, network *network* interface *int/ int_no*

Description: A topology change notification BPDU has been received from the specified MAC address, but the port is not participating in the spanning tree protocol.

SEST.031

Level: C-INFO

Short Syntax: SEST.031 *bridge_type- se_id* desig port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.031 *bridge_type- se_id* becoming designated port *bridge_port*, network *network* interface *int/ int_no*

Description: This bridge is declaring itelf the designated port on the LAN connected to this port.

SEST.032

Level: UI-ERROR

Short Syntax: SEST.032 DROP: *bpdu_type* BPDU frame recvd on non-parti port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.032 DROP: *bpdu_type* BPDU frame received on non-participating port *bridge_port*, network *network* interface *int/ int_no*

Description: BPDU has been received, but the port is not participating in the SE Spanning Tree Protocol.

SEST.033

Level: C-INFO

Short Syntax: SEST.033 BPDU from SE ID *bpdu_se_id* rcvd frame *source_address bridge_type- se_id* port *bridge_port*, net- *network* int *int/ int_no*

Long Syntax: SEST.033 BPDU from SE ID *bpdu_se_id* rcvd frame *source_address* on *bridge_type- se_id* port *bridge_port*, network *network* interface *int/ int_no*

Description: A configuration BPDU has been received from a SuperELAN bridge participating in different SuperELAN. BPDU is discarded.

Cause: Two SuperELAN interfaces are connected to the same ELAN or This first network implementation is not recommended, since the SuperELAN spanning tree operates only within the context of a single SuperELAN. In this case, the spanning tree may form improperly causing network loops.

Action: Remove SuperELAN interface from the ELAN which has more than one SuperELAN interface attached.

SEST.034

Level: P-TRACE

Short Syntax: SEST.034 SE- *se_id*:Trace incoming SE Spanning Tree frame from ELAN ' *elan*'

Long Syntax: SEST.034 SE- *se_id*:Trace incoming SE Spanning Tree frame from ELAN ' *elan*'

Description: Trace incoming SE Spanning Tree frame.

SEST.035

Level: P-TRACE

Short Syntax: SEST.035 SE- *se_id*:Trace outgoing SE Spanning Tree frame to ELAN ' *elan*'

Long Syntax: SEST.035 SE- *se_id*:Trace outgoing SE Spanning Tree frame to ELAN ' *elan*'

Description: Trace outgoing SE Spanning Tree frame.

SEST.036

Level: UE-ERROR

Short Syntax: SEST.036 SE- *se_id*:SPT frame rcvd on ELAN ' *elan*' fwd by a non-short-cut brdg, frame drop.

Long Syntax: SEST.036 SE- *se_id*:SPT frame received on ELAN ' *elan*' forwarded by a non-short-cut bridge, frame dropped.

Description: A SuperELAN Spanning Tree configuration frame was received on an interface that was forwarded by a proxy device which does not support short-cut bridging. The frame is dropped to prevent the SE SPT topology from converging across a non-short-cut network. If message persists, check legacy bridge topology for possible spanning tree errors.

SEST.037

Level: UI-ERROR

Short Syntax: SEST.037 SE- *se_id*:Unable to get LES address for ELAN ' *elan*'

Long Syntax: SEST.037 SE- *se_id*:Unable to get LES address for ELAN ' *elan*'

Description: The LEC interface's LES address for the specified ELAN could not be retrieved. If message persists, contact customer support.

SEST.038

Level: UE-ERROR

Short Syntax: SEST.038 SE- *se_id*:Cfg BPDU rcvd on ifc from which it was sent ELAN ' *elan*', frame drop

Long Syntax: SEST.038 SE- *se_id*:Cfg BPDU received on interface from which it was sent ELAN ' *elan*', frame dropped

Description: A SuperELAN Configuration BPDU was looped back and received on the interface from which it was sent. BPDU was dropped. Installing SNAP filter '10005A-80D7' to all bridges connected to the SuperELAN will prevent loopback of SE STP BPDU frames.

Fatal sestubpdu

Short Syntax: Attempt to send unknown SE-BPDU type

Description: The code attempted to send an unknown type of SE-BPDU.

Cause: Possible software bug.

Action: Get crash dump, contact customer service.

Chapter 111. Synchronous Data Link Control (SDLC)

This chapter describes Synchronous Data Link Control (SDLC) messages. For information on message content and how to use the message, refer to the Introduction.

SDLC.001

Level: C-INFO

Short Syntax: SDLC.001 nt *network ID* - SDLC support installed for QSL

Long Syntax: SDLC.001 SDLC support installed for QSL, on network *network ID*

Description: DLSw SDLC has been initialized for operation over the serial device and is now available for use in the router.

SDLC.002

Level: C-INFO

Short Syntax: SDLC.002 dflt cfg used on stn *Address*, nt *network ID*

Long Syntax: SDLC.002 No remote configuration was defined for SDLC address *Address* - default settings used, on network *network ID*

Description: An open was attempted on an SDLC station, but a configuration record was not defined via the ADD STATION command. The station will be opened using default configuration value. This station will be listed in the monitor process LIST STATION ALL command. An asterisk * next to the station address signifies that a default configuration is in use.

SDLC.003

Level: CE-ERROR

Short Syntax: SDLC.003 no mem to copy to SDLC, nt *network ID*

Long Syntax: SDLC.003 Cannot copy a user buffer to to SDLC - Out of memory, on network *network ID*

Description: There is currently no memory available to copy user data to SDLC. Another attempt will be made at a later time.

SDLC.004

Level: U-INFO

Short Syntax: SDLC.004 frame dropped nt *network ID* not active

Long Syntax: SDLC.004 Inbound frame was dropped - SDLC not yet active on network *network ID*.

Description: A frame was received on an interface that is not yet owned by an SDLC client. All packets received are dropped until an SDLC client opens the port for use.

SDLC.005

Level: CE-ERROR

Short Syntax: SDLC.005 nt *network ID* congested - pkt dropped

Long Syntax: SDLC.005 Packet dropped due to no SDLC credit or memory shortage, on network *network ID* . Temporary.

Description: An incoming packet was dropped due to no SDLC receive credit or a temporary memory shortage.

SDLC.006

Level: CE-ERROR

Short Syntax: SDLC.006 nt *network ID* - I_ERR on rcv

Long Syntax: SDLC.006 Packet received with I_ERR set, on network *network ID*

Description: real_sdlc_in received a packet with I_ERR set

SDLC.007

Level: CE-ERROR

Short Syntax: SDLC.007 nt *network ID* - tx to dev fail (*status*)

Long Syntax: SDLC.007 Failure to send packet to device, on network *network ID*, status = *status*

Description: The call to netsend() failed while trying to send a frame from SDLC.

SDLC.008

Level: P-TRACE

Short Syntax: SDLC.008 tx *count* bytes to *address* (nt *network ID*): *octets*

Long Syntax: SDLC.008 Transmit to link station *count*, *address* bytes, on network *network ID*: *octets*

Description: The router transmitted an SDLC frame.

This is the entire frame, including the SDLC header. The router logs all transmitted SDLC frames with this message. To log only transmitted SDLC I-frames, use SDLC_53.

SDLC.009

Level: P-TRACE

Short Syntax: SDLC.009 rx *count* bytes from *address* (nt *network ID*): *octets*

Long Syntax: SDLC.009 Received *count* bytes from link station *address*, on network *network ID*: *octets*

Description: The router received an SDLC frame. This is the entire frame, including the SDLC header. The router logs all received SDLC frames with this message. To log only received SDLC I-frames, use SDLC_52.

SDLC.010

Level: C-INFO

Short Syntax: SDLC.010 port ACTIVE, nt *network ID*

Long Syntax: SDLC.010 Request to bring up SDLC, on network *network ID*

Description: An entity in the router has attached to the interface and can now use SDLC services.

SDLC.011

Level: C-INFO

Short Syntax: SDLC.011 port INACTIVE, nt *network ID*

Long Syntax: SDLC.011 Request to bring down SDLC, on network *network ID*

Description: An entity in the router is no longer using SDLC services on this interface.

SDLC.012

Level: C-INFO

Short Syntax: SDLC.012 Link status: *Exception*, nt *network ID*

Long Syntax: SDLC.012 Link status change *Exception* occurred, on network *network ID*

Description: An interface signal has changed state. Note: an unwieldy number of these messages will be generated when the interface is operating in half duplex mode.

SDLC.013

Level: C-INFO

Short Syntax: SDLC.013 addr *Address* -> NRM, nt *network ID*

Long Syntax: SDLC.013 Station *Address* is now UP, on network *network ID*

Description: The SDLC link is now operating in Normal Response Mode, meaning that a SDLC connection is now in progress.

SDLC.014

Level: C-INFO

Short Syntax: SDLC.014 SNRM refused, addr *Address* nt *network ID*

Long Syntax: SDLC.014 Remote station refused SNRM, link station *Address* remains DOWN on network *network ID*

Description: An attempt by the router to connect to a remote link station has been refused.

SDLC.015

Level: C-INFO

Short Syntax: SDLC.015 addr *Address* -> NDM, nt *network ID*

Long Syntax: SDLC.015 Station *Address* is now DOWN, on network *network ID*

Description: The SDLC link is now operating in Normal Disconnect Mode, meaning that a SDLC connection has been terminated in an orderly fashion.

SDLC.016

Level: U-INFO

Short Syntax: SDLC.016 LnkStn *Address* excpt *Exception*, nt *network ID*

Long Syntax: SDLC.016 Exception *Address* occurred on Link Station *Exception*, on network *network ID*

Description: The SDLC protocol has been initialized.

SDLC.017

Level: U-INFO

Short Syntax: SDLC.017 nt *network ID* - inter-poll delay forced to 0

Long Syntax: SDLC.017 Network *network ID*, cannot use inter-poll delay and Primary or Secondary Poll Pause

Description: Link inter-poll Delay cannot be enabled when Primary or Secondary Poll Pause is enabled. Primary or Secondary Poll Pause are the preferred poll timers.

SDLC.018

Level: U-INFO

Short Syntax: SDLC.018 nt *network ID* Stn *address* - Secondary Poll Pause forced to 0

Long Syntax: SDLC.018 Network *network ID* Station *address*, cannot use Secondary Poll Pause since link inter-poll is enabled

Description: There may have been residual configuration information which created an inconsistency: the user has two secondary pauses enabled. It is best to use Secondary Poll Pause and disable the link inter-poll delay by reconfiguring. This ELS message indicates that Secondary Poll Pause has been forced to 0 and inter-poll has been left unchanged.

SDLC.020

Level: U-INFO

Short Syntax: SDLC.020 nt *network ID* stn *address*: DLC_LINK_FAULT_CONDITION

Long Syntax: SDLC.020 Network *network ID* SDLC station *address*: DLC_LINK_FAULT_CONDITION

Description: SDLC detected a fault on the link, and terminated all active SDLC connections on the link. This is usually due to a loss of DSR, CTS, or DCD on a full-duplex line, or loss of DSR on a half-duplex line.

SDLC.021

Level: U-INFO

Short Syntax: SDLC.021 nt *network ID* stn *address*: DLC_RX_EXCEED_WINDOW_SIZE

Long Syntax: SDLC.021 Network *network ID* SDLC station *address*: DLC_RX_EXCEED_WINDOW_SIZE

Description: SDLC has received more than the number of frames configured as the RECEIVE WINDOW before SDLC could respond.

SDLC.022

Level: U-INFO

Short Syntax: SDLC.022 nt *network ID* stn *address*: DLC_RX_LOCAL_PROTOCOL_ERROR

Long Syntax: SDLC.022 Network *network ID* SDLC station *address*: DLC_RX_LOCAL_PROTOCOL_ERROR

Description: The router detected a SDLC protocol error. As a result, the router terminated the SDLC connection.

SDLC.023

Level: U-INFO

Short Syntax: SDLC.023 nt *network ID* stn *address*: DLC_XID_RETRY_LIMIT_REACHED

Long Syntax: SDLC.023 Network *network ID* SDLC station *address*: DLC_XID_RETRY_LIMIT_REACHED

Description: The remote link station is not responding to XID frames sent by the router.

SDLC.024

Level: U-INFO

Short Syntax: SDLC.024 nt *network ID* stn *address*: DLC_TEST_RETRY_LIMIT_REACHED

Long Syntax: SDLC.024 Network *network ID* SDLC station *address*: DLC_TEST_RETRY_LIMIT_REACHED

Description: The remote link station is not responding to TEST frames sent by the router.

SDLC.025

Level: U-INFO

Short Syntax: SDLC.025 nt *network ID* stn *address*: DLC_SNRM_RETRY_LIMIT_REACHED

Long Syntax: SDLC.025 Network *network ID* SDLC station *address*: DLC_SNRM_RETRY_LIMIT_REACHED

Description: The remote link station is not responding to SNRM frames sent by the router. The connection attempt has failed.

SDLC.026

Level: U-INFO

Short Syntax: SDLC.026 nt *network ID* stn *address*: DLC_POLL_RETRY_LIMIT_REACHED

Long Syntax: SDLC.026 Network *network ID* SDLC station *address*: DLC_POLL_RETRY_LIMIT_REACHED

Description: The remote link station is not responding to polls (RR or RNR) sent by the router. As a result, the router terminated the connection.

SDLC.027

Level: U-INFO

Short Syntax: SDLC.027 nt *network ID* stn *address*: DLC_RX_FRMR_INV_CTL_FIELD

Long Syntax: SDLC.027 Network *network ID* SDLC station *address*: DLC_RX_FRMR_INV_CTL_FIELD

Description: SDLC has received a Frame Reject (FRMR) frame indicating that the remote link station received a frame with an invalid control field.

SDLC.028

Level: U-INFO

Short Syntax: SDLC.028 nt *network ID* stn *address*: DLC_RX_FRMR_INV_LENGTH

Long Syntax: SDLC.028 Network *network ID* SDLC station *address*: DLC_RX_FRMR_INV_LENGTH

Description: SDLC has received a Frame Reject (FRMR) frame indicating that the remote link station received a frame that was too short.

SDLC.029

Level: U-INFO

Short Syntax: SDLC.029 nt *network ID* stn *address*: DLC_RX_FRMR_LONG_I_FIELD

Long Syntax: SDLC.029 Network *network ID* SDLC station *address*: DLC_RX_FRMR_LONG_I_FIELD

Description: SDLC has received a Frame Reject (FRMR) frame indicating that the remote link station received a frame that was too long.

SDLC.030

Level: U-INFO

Short Syntax: SDLC.030 nt *network ID* stn *address*: DLC_RX_FRMR_INV_NR

Long Syntax: SDLC.030 Network *network ID* SDLC station *address*: DLC_RX_FRMR_INV_NR

Description: SDLC has received a Frame Reject (FRMR) frame indicating that the remote link station received a frame with an invalid N(r) in the control field.

SDLC.031

Level: U-INFO

Short Syntax: SDLC.031 nt *network ID* stn *address*: DLC_RX_FRMR_NO_I_FIELD

Long Syntax: SDLC.031 Network *network ID* SDLC station *address*: DLC_RX_FRMR_NO_I_FIELD

Description: SDLC has received a Frame Reject (FRMR) frame indicating that the remote link station received an I-frame with no data in the I field.

SDLC.032

Level: U-INFO

Short Syntax: SDLC.032 nt *network ID* stn *address*: DLC_RX_FRAME_INV_CTL_FIELD

Long Syntax: SDLC.032 Network *network ID* SDLC station *address*: DLC_RX_FRAME_INV_CTL_FIELD

Description: SDLC has received a frame with an invalid control field.

SDLC.033

Level: U-INFO

Short Syntax: SDLC.033 nt *network ID* stn *address*: DLC_RX_FRAME_INV_LENGTH

Long Syntax: SDLC.033 Network *network ID* SDLC station *address*: DLC_RX_FRAME_INV_LENGTH

Description: SDLC has received a frame that was too short.

SDLC.034

Level: U-INFO

Short Syntax: SDLC.034 nt *network ID* stn *address*: DLC_RX_FRAME_LONG_I_FIELD

Long Syntax: SDLC.034 Network *network ID* SDLC station *address*: DLC_RX_FRAME_LONG_I_FIELD

Description: SDLC has received a frame that was too long.

SDLC.035

Level: U-INFO

Short Syntax: SDLC.035 nt *network ID* stn *address*: DLC_RX_FRAME_INV_NR

Long Syntax: SDLC.035 Network *network ID* SDLC station *address*: DLC_RX_FRAME_INV_NR

Description: SDLC has received a frame with an invalid N(r) in the control field.

SDLC.036

Level: U-INFO

Short Syntax: SDLC.036 nt *network ID* stn *address*: DLC_RX_DM

Long Syntax: SDLC.036 Network *network ID* SDLC station *address*: DLC_RX_DM

Description: SDLC received a Disconnected Mode (DM) frame. A remote secondary link station sent the frame to indicate that it accepted a previously received DISC frame. The link disconnection is now complete.

SDLC.037

Level: U-INFO

Short Syntax: SDLC.037 nt *network ID* stn *address*: DLC_RX_RD

Long Syntax: SDLC.037 Network *network ID* SDLC station *address*: DLC_RX_RD

Description: SDLC received a Request Disconnect (RD) frame. The SDLC client should respond to this by sending a Disconnect (DISC) frame.

SDLC.038

Level: U-INFO

Short Syntax: SDLC.038 nt *network ID* stn *address*: DLC_RX_RIM

Long Syntax: SDLC.038 Network *network ID* SDLC station *address*: DLC_RX_RIM

Description: SDLC received a Request Initialization Mode (RIM) frame. The SDLC client should respond to this by sending a Set Initialization Mode (SIM) frame.

SDLC.039

Level: U-INFO

Short Syntax: SDLC.039 nt *network ID* stn *address*: DLC_LINK_INACTIVITY_DETECTION

Long Syntax: SDLC.039 Network *network ID* SDLC station *address*: DLC_LINK_INACTIVITY_DETECTION

Description: Reserved for possible future use.

SDLC.040

Level: U-INFO

Short Syntax: SDLC.040 nt *network ID* stn *address*: DLC_TX_FRMR_INV_CTL_FIELD

Long Syntax: SDLC.040 Network *network ID* SDLC station *address*: DLC_TX_FRMR_INV_CTL_FIELD

Description: SDLC entered a frame reject (FRMR) state because it received a frame with an invalid control field.

SDLC.041

Level: U-INFO

Short Syntax: SDLC.041 nt *network ID* stn *address*: DLC_TX_FRMR_INV_LENGTH

Long Syntax: SDLC.041 Network *network ID* SDLC station *address*: DLC_TX_FRMR_INV_LENGTH

Description: SDLC entered a frame reject (FRMR) state because it received a frame that was too short.

SDLC.042

Level: U-INFO

Short Syntax: SDLC.042 nt *network ID* stn *address*: DLC_TX_FRMR_LONG_I_FIELD

Long Syntax: SDLC.042 Network *network ID* SDLC station *address*: DLC_TX_FRMR_LONG_I_FIELD

Description: SDLC entered a frame reject (FRMR) state because it received a frame that was too long.

SDLC.043

Level: U-INFO

Short Syntax: SDLC.043 nt *network ID* stn *address*: DLC_TX_FRMR_INV_NR

Long Syntax: SDLC.043 Network *network ID* SDLC station *address*: DLC_TX_FRMR_INV_NR

Description: SDLC entered a frame reject (FRMR) state because it received a frame with an invalid N(r) in the control field.

SDLC.044

Level: U-INFO

Short Syntax: SDLC.044 nt *network ID* stn *address*: DLC_RX_SNRM_WHILE_IN_NRM

Long Syntax: SDLC.044 Network *network ID* SDLC station *address*: DLC_RX_SNRM_WHILE_IN_NRM

Description: Reserved for possible future use.

SDLC.045

Level: U-INFO

Short Syntax: SDLC.045 nt *network ID* stn *address*: DLC_PORT_DISABLED

Long Syntax: SDLC.045 Network *network ID* SDLC station *address*: DLC_PORT_DISABLED

Description: The user disabled the interface from the SDLC console. All current connections are terminated.

SDLC.046

Level: U-INFO

Short Syntax: SDLC.046 nt *network ID* stn *address*: DLC_PORT_ENABLED

Long Syntax: SDLC.046 Network *network ID* SDLC station *address*: DLC_PORT_ENABLED

Description: The user enabled the interface from the SDLC console.

SDLC.047

Level: U-INFO

Short Syntax: SDLC.047 nt *network ID*: CLOSED

Long Syntax: SDLC.047 Network *network ID* SDLC link: DLC_STATION_CLOSED

Description: The interface has been closed by SDLC. SDLC is no longer running over this interface.

SDLC.048

Level: U-INFO

Short Syntax: SDLC.048 nt *network ID* stn *address*: DISABLED

Long Syntax: SDLC.048 Network *network ID* SDLC station *address*: DLC_LS_DISABLED

Description: The user disabled a remote link station on this interface from the SDLC console. Any existing connection was terminated.

SDLC.049

Level: U-INFO

Short Syntax: SDLC.049 nt *network ID* stn *address*: ENABLED

Long Syntax: SDLC.049 Network *network ID* SDLC station *address*: DLC_LS_ENABLED

Description: The user enabled a remote link station on this interface from the SDLC console.

SDLC.050

Level: P-TRACE

Short Syntax: SDLC.050 nt *network ID* stn *address* - rx UI bytes *length*: *byte_count*

Long Syntax: SDLC.050 Network *network ID* received UI from SDLC addr *address length* bytes: *byte_count*

Description: The router received an Unnumbered Information (UI) frame on this interface.

SDLC.051

Level: P-TRACE

Short Syntax: SDLC.051 nt *network ID* stn *address* - tx UI bytes *byte_count*: *octets*

Long Syntax: SDLC.051 Network *network ID* sent UI to SDLC addr *address byte_count* bytes: *octets*

Description: The router transmitted an Unnumbered Information (UI) frame on this interface.

SDLC.052

Level: P-TRACE

Short Syntax: SDLC.052 nt *network ID* - rx I on *address byte_count* bytes: *octets*

Long Syntax: SDLC.052 Network *network ID* received I from SDLC addr *address byte_count* bytes: *octets*

Description: The router received an Information (I) frame on this interface. To log all received SDLC frames, use SDLC_9.

SDLC.053

Level: P-TRACE

Short Syntax: SDLC.053 nt *network ID* - tx I on *address byte_count* bytes: *octets*

Long Syntax: SDLC.053 Network *network ID* sent I to SDLC addr *address byte_count* bytes: *octets*

Description: The router transmitted an Information (I) frame on this interface. To log all received SDLC frames, use SDLC_8.

SDLC.054

Level: U-INFO

Short Syntax: SDLC.054 nt *network ID* Stn *address* - MaxBTU too large for link - adjusted (*oldBTUSize* -> *newBTUSize*)

Long Syntax: SDLC.054 Network *network ID* Station *address*, Max BTU size too large for link - adjusted (*oldBTUSize* -> *newBTUSize*)

Description: The max BTU size configured for a remote link station exceeds that defined for the link. The router adjusted the value for the remote link station temporarily. To avoid this message in the future, change the max BTU size with the SET REMOTE MAX-PACKET command.

SDLC.055

Level: U-INFO

Short Syntax: SDLC.055 nt *network ID* Stn *address* - Rx wdw sz not compat w/modulo - adjusted (*oldRxWindow* -> *newRxWindow*)

Long Syntax: SDLC.055 Network *network ID* Station *address*, Window size is inconsistent with modulo for link - adjusted (*oldRxWindow* -> *newRxWindow*)

Description: The modulo for this link has been changed by the user, rendering the window sizes for all pre-defined remote link stations invalid. The window size has been temporarily adjusted. When a link is configured for mod-8, the valid window sizes are 0 to 7. When the link is configured for mod-128, the valid window sizes are 8 to 128. To avoid this message in the future, change the receive window size with the SET REMOTE RECEIVE-WINDOW command.

SDLC.056

Level: U-INFO

Short Syntax: SDLC.056 nt *network ID* Stn *address* - Tx wdw sz not compat w/modulo - adjusted (*oldTxWindow* -> *newTxWindow*)

Long Syntax: SDLC.056 Network *network ID* Station *address*, Window size is inconsistent with modulo for link - adjusted (*oldTxWindow* -> *newTxWindow*)

Description: The modulo for this link has been changed by the user, rendering the window sizes for all pre-defined remote link stations invalid. The window size has been temporarily adjusted. When a link is configured for mod-8, the valid window sizes are 0 to 7. When the link is configured for mod-128, the valid window sizes are 8 to 128. To avoid this message in the future, change the transmit window size with the 'SET REMOTE TRANSMIT-WINDOW' command.

SDLC.057

Level: U-INFO

Short Syntax: SDLC.057 nt *network ID* - Link cfg corrupted - using default

Long Syntax: SDLC.057 Network *network ID* - Link configuration corrupted, using defaults.

Description: The link configuration was somehow corrupted, possibly due to a software upgrade. A default link configuration has been created. Before operating SDLC, review the newly-created configuration and adjust as necessary.

SDLC.058

Level: U-INFO

Short Syntax: SDLC.058 nt *network ID* - cfg XID/TEST timeout corrupted - fixed

Long Syntax: SDLC.058 Network *network ID* - Configured XID/TEST timeout corrupted - fixed.

Description: An invalid XID/TEST timeout value was detected in the link configuration, possibly due to a software upgrade. It has been temporarily changed to a valid value. To avoid this message in the future, set the XID/TEST timeout from the SDLC config prompt for this interface.

SDLC.059

Level: U-INFO

Short Syntax: SDLC.059 nt *network ID* - cfg XID/TEST retry count corrupted - fixed

Long Syntax: SDLC.059 Network *network ID* - Configured XID/TEST retry count corrupted - fixed.

Description: An invalid XID/TEST retry value was detected in the link configuration, possibly due to a software upgrade. It has been temporarily changed to a valid value. To avoid this message in the future, set the XID/TEST retry count from the SDLC config prompt for this interface.

SDLC.060

Level: U-INFO

Short Syntax: SDLC.060 nt *network ID* - cfg SNRM timeout corrupted - fixed

Long Syntax: SDLC.060 Network *network ID* - Configured XID/TEST timeout value corrupted - fixed.

Description: An invalid SNRM timeout value was detected in the link configuration, possibly due to a software upgrade. It has been temporarily changed to a valid value. To avoid this message in the future, set the SNRM timeout from the SDLC config prompt for this interface.

SDLC.061

Level: U-INFO

Short Syntax: SDLC.061 nt *network ID* - cfg SNRM retry count corrupted - fixed

Long Syntax: SDLC.061 Network *network ID* - Configured SNRM retry count corrupted - fixed.

Description: An invalid SNRM retry value was detected in the link configuration, possibly due to a software upgrade. It has been temporarily changed to a valid value. To avoid this message in the future, set the SNRM retry count from the SDLC config prompt for this interface.

SDLC.062

Level: U-INFO

Short Syntax: SDLC.062 nt *network ID* - cfg POLL timeout corrupted - fixed

Long Syntax: SDLC.062 Network *network ID* - Configured POLL timeout value corrupted - fixed.

Description: An invalid POLL timeout value was detected in the link configuration, possibly due to a software upgrade. It has been temporarily changed to a valid value. To avoid this message in the future, set the POLL timeout from the SDLC config prompt for this interface.

SDLC.063

Level: U-INFO

Short Syntax: SDLC.063 nt *network ID* - cfg inter-POLL delay corrupted - fixed

Long Syntax: SDLC.063 Network *network ID* - Configured inter-POLL delay value corrupted - fixed.

Description: An invalid inter-POLL delay value was detected in the link configuration, possibly due to a software upgrade. It has been temporarily changed to a valid value. To avoid this message in the future, set the

inter-POLL delay from the SDLC config prompt for this interface.

SDLC.064

Level: U-INFO

Short Syntax: SDLC.064 nt *network ID* - cfg POLL retry count corrupted - fixed

Long Syntax: SDLC.064 Network *network ID* - Configured POLL retry count corrupted - fixed.

Description: An invalid POLL retry value was detected in the link configuration, possibly due to a software upgrade. It has been temporarily changed to a valid value. To avoid this message in the future, set the POLL retry count from the SDLC config prompt for this interface.

SDLC.065

Level: U-INFO

Short Syntax: SDLC.065 nt *network ID* - cfg inactivity timeout corrupted - fixed

Long Syntax: SDLC.065 Network *network ID* - Configured inactivity timeout value corrupted - fixed.

Description: Reserved for possible future use.

SDLC.066

Level: U-INFO

Short Syntax: SDLC.066 nt *network ID* - cfg RTS hold duration corrupted - fixed

Long Syntax: SDLC.066 Network *network ID* - Configured RTS hold duration value corrupted - fixed.

Description: An invalid RTS hold value was detected in the link configuration, possibly due to a software upgrade. It has been temporarily changed to a valid value. To avoid this message in the future, set the RTS hold value from the SDLC config prompt for this interface.

SDLC.067

Level: U-INFO

Short Syntax: SDLC.067 nt *network ID* - cfg max frame size corrupted - fixed

Long Syntax: SDLC.067 Network *network ID* - Configured max frame size value corrupted - fixed.

Description: An invalid maximum frame size value was detected in the link configuration, possibly due to a software upgrade. It has been temporarily changed to a valid value. To avoid this message in the future, set the maximum frame size from the SDLC config prompt for this interface.

SDLC.068

Level: C-INFO

Short Syntax: SDLC.068 nt *network ID* - link ctrs reset by usr

Long Syntax: SDLC.068 Network *network ID* - SDLC link counters were reset by user.

Description: The link counters have been reset from the SDLC console.

SDLC.069

Level: U-INFO

Short Syntax: SDLC.069 nt *network ID* - cannot reset link ctrs

Long Syntax: SDLC.069 Network *network ID* - SDLC link counters cannot be reset.

Description: The link counters could not be reset from the SDLC console. This is usually a temporary condition. Retry the operation.

SDLC.070

Level: C-INFO

Short Syntax: SDLC.070 nt *network ID* lnk stn *address* - link ctrs reset by usr

Long Syntax: SDLC.070 Network *network ID* Link Station *address* - SDLC link counters were reset by user.

Description: The user reset the counters for a remote secondary station from the SDLC console.

SDLC.071

Level: U-INFO

Short Syntax: SDLC.071 nt *network ID* Stn *address* - Protocol forced to TWO-WAY-ALTERNATING since Group Poll is used

Long Syntax: SDLC.071 Network *network ID* Station *address*, cannot use Primary or Secondary Group Poll with TWO-WAY-SIMULTANEOUS

Description: There may have been residual configuration information which created an inconsistency: TWO-WAY-SIMULTANEOUS link protocol cannot be used with either primary or secondary group poll. As the station was activated the protocol was forced to TWO-WAY-ALTERNATING.

SDLC.072

Level: U-INFO

Short Syntax: SDLC.072 nt *network ID* stn *address*: CLOSED

Long Syntax: SDLC.072 Network *network ID* SDLC

station *address*: DLC_SAP_CLOSED

Description: The remote link station has been closed by the router and is no longer active.

SDLC.073

Level: CE-ERROR

Short Syntax: SDLC.073 nt *network ID*: lo prio buffer alloc failed

Long Syntax: SDLC.073 Network *network ID*: low priority buffer request failed

Description: SDLC failed to allocate memory from the router's heap because it has already allocated its fair share. SDLC will recover from this usually temporary problem.

SDLC.074

Level: CE-ERROR

Short Syntax: SDLC.074 nt *network ID*: hi prio buffer alloc failed

Long Syntax: SDLC.074 Network *network ID*: high priority buffer request failed

Description: SDLC failed to allocate a high-priority buffer from the router's heap. As a result, SDLC will temporarily experience a loss of performance. SDLC sessions could possibly be lost if the condition persists.

SDLC.075

Level: CE-ERROR

Short Syntax: SDLC.075 nt *network ID*: buffer alloc failed - credit ok

Long Syntax: SDLC.075 Network *network ID*: buffer alloc request failed, but credit is okay

Description: SDLC failed to allocate memory from the router's heap because the heap is exhausted. SDLC will recover from this usually temporary problem.

SDLC.076

Level: CE-ERROR

Short Syntax: SDLC.076 nt *network ID*: cannot xmit I-frame. Will re-transmit

Long Syntax: SDLC.076 Network *network ID*: cannot transmit I-frame. Will re-transmit later

Description: SDLC could not send an Information (I) frame to the interface. SDLC will attempt to re-transmit it later.

SDLC.077

Level: CE-ERROR

Short Syntax: SDLC.077 nt *network ID*: cannot xmit S-frame - dropped

Long Syntax: SDLC.077 Network *network ID*: cannot transmit S-frame - dropped

Description: SDLC could not send a Supervisory (S) frame to the interface. SDLC will attempt to re-transmit it later.

SDLC.078

Level: CE-ERROR

Short Syntax: SDLC.078 nt *network ID*: cannot xmit U-frame. SDLC will recover

Long Syntax: SDLC.078 Network *network ID*: cannot transmit U-frame - SDLC will recover

Description: SDLC could not send an Unnumbered Information (UI) frame to the interface. SDLC will attempt to re-transmit it later.

SDLC.079

Level: CE-ERROR

Short Syntax: SDLC.079 nt *network ID*: cannot xmit XID/TEST frame. Will re-xmit

Long Syntax: SDLC.079 Network *network ID*: cannot transmit U-frame - SDLC will re-transmit

Description: SDLC could not send an XID or TEST frame to the interface. SDLC will attempt to re-transmit it later.

SDLC.080

Level: CE-ERROR

Short Syntax: SDLC.080 nt *network ID*: rx inv frame type - dropped

Long Syntax: SDLC.080 Network *network ID*: received invalid frame type - dropped

Description: SDLC received an invalid frame type. This frame was ignored.

SDLC.081

Level: CE-ERROR

Short Syntax: SDLC.081 nt *network ID*: rx frame from sec stn not polled - dropped

Long Syntax: SDLC.081 Network *network ID*: received a frame from a secondary station that was not polled - dropped

Description: SDLC received a frame from a remote link station that it had not polled. This frame was

ignored. This error may also occur if the poll response timeout is too short.

SDLC.082

Level: CE-ERROR

Short Syntax: SDLC.082 nt *network ID*: rx UI frame from sec stn w/o F-bit - dropped

Long Syntax: SDLC.082 Network *network ID*: received a UI frame from a secondary station without the Final bit set - dropped

Description: SDLC received an Unnumbered Information (UI) frame without the Final (F) bit set. This frame was dropped.

SDLC.083

Level: CE-ERROR

Short Syntax: SDLC.083 nt *network ID*: rx bcast on mpt line - dropped

Long Syntax: SDLC.083 Network *network ID*: received a broadcast frame from a secondary station on a multipoint line - dropped

Description: SDLC received a frame to the broadcast address on a multipoint line. The frame was dropped.

SDLC.084

Level: UE-ERROR

Short Syntax: SDLC.084 SDLC not up on nt *network ID* - no LINK config

Long Syntax: SDLC.084 Network *network ID*: SDLC not brought up because no LINK configuration is defined

Description: SDLC could not be initialized because there is no SDLC link configuration for this interface.

SDLC.085

Level: UE-ERROR

Short Syntax: SDLC.085 nt *network ID*: signal ctl rq failed - *reason*

Long Syntax: SDLC.085 Network *network ID*: signal control request failed because *reason*

Description: SDLC could not control one or more signals on the interface. This could occur if you attempt to run SDLC over an unsupported interface.

SDLC.086

Level: CE-ERROR

Short Syntax: SDLC.086 HDX, DCD hi during xmit, nt *network ID*

Long Syntax: SDLC.086 HDX, DCD went high during HDX transmit, on network *network ID*

Description: DCD went high during transmission of a frame in half duplex mode. This is a protocol violation, and the interface will go down shortly in order to correct the problem.

SDLC.087

Level: C-INFO

Short Syntax: SDLC.087 HDX, CTS now low. Premature DCD recovery complete, nt *network ID*

Long Syntax: SDLC.087 HDX, CTS now low. Premature DCD recovery complete, on network *network ID*

Description: The CTS signal has transitioned to low while the interface was recovering from a half-duplex protocol violation. The link is now in the correct state and data transfer may resume.

SDLC.088

Level: CE-ERROR

Short Syntax: SDLC.088 HDX, unsolicited signal while idle, nt *network ID*

Long Syntax: SDLC.088 HDX, unsolicited signal while idle, on network *network ID*

Description: A signal was asserted by the connected device while the line was idle. When operating in half-duplex mode, only the DTR/DSR signal shuold be asserted on an idle interface.

SDLC.089

Level: CE-ERROR

Short Syntax: SDLC.089 HDX, DSR low during xmit, nt *network ID*

Long Syntax: SDLC.089 HDX, DSR went low during HDX transmit, on network *network ID*

Description: The DSR signal went low during transmission. This is a protocol violation, and the interface will go down shortly in order to correct the problem.

SDLC.090

Level: CE-ERROR

Short Syntax: SDLC.090 nt *network ID*: rx inv frame type while closing link *address* - dropped

Long Syntax: SDLC.090 Network *network ID*: received inappropriate frame while closing link *address* - dropped

Description: SDLC received a frame other than a UA

while trying to close the link station. The router ignored this frame.

SDLC.091

Level: C-INFO

Short Syntax: SDLC.091 addr *Address* added, nt *network ID*

Long Syntax: SDLC.091 Secondary station *Address* has been dynamically added, on network *network ID*

Description: An SDLC remote link station was added by the user from the SDLC console and is now available for use.

SDLC.092

Level: C-INFO

Short Syntax: SDLC.092 addr *Address* deleted, nt *network ID*

Long Syntax: SDLC.092 Secondary station *Address* has been dynamically deleted, on network *network ID*

Description: An SDLC remote link station was deleted by the user from the SDLC console and is no longer available for use.

SDLC.093

Level: CE-ERROR

Short Syntax: SDLC.093 nt *network ID*: rx frame from invalid stn addr - dropped

Long Syntax: SDLC.093 Network *network ID*: received a frame from an invalid station address - dropped

Description: SDLC received a frame from a remote link station that contains an unrecognized station address. The router ignored this frame.

SDLC.094

Level: UE-ERROR

Short Syntax: SDLC.094 nt *network ID* lnk stn *address* - stn disabled, rx frame dropped

Long Syntax: SDLC.094 Network *network ID* Link Station *address* - station is disabled; frame ignored.

Description: The router ignored the received frame for this station, because the target station is in a disabled state.

SDLC.095

Level: UE-ERROR

Short Syntax: SDLC.095 nt *network ID* stn *address*: DLC_RX_NXID_WHILE_IN_NRM

Long Syntax: SDLC.095 Network *network ID* SDLC station *address*: DLC_RX_NXID_WHILE_IN_NRM

Description: The local SDLC secondary link station received a null XID frame from a remote link station while in NRM. We treat this as an indication that the link failed at the sending end and this is an attempt to reestablish the connection.

SDLC.096

Level: C-INFO

Short Syntax: SDLC.096 port FULL, nt *network ID*

Long Syntax: SDLC.096 Transmit data queue has reached its limit on network *network ID*

Description: The SDLC user (such as DLS or APPN) should not queue more data to this interface.

SDLC.097

Level: C-INFO

Short Syntax: SDLC.097 port newly AVAILABLE, nt *network ID*

Long Syntax: SDLC.097 Transmit data queue has dropped below its threshold on network *network ID*

Description: The SDLC user (such as DLS or APPN) may now queue more data to this interface.

SDLC.098

Level: C-INFO

Short Syntax: SDLC.098 nt *network ID*: rx invalid control field for station *address* -- no link station timer running

Long Syntax: SDLC.098 Network *network ID*: received a UI frame for station *address* and no link station timer is running

Description: SDLC received an Unnumbered Information (UI) frame and it was found that no link station timer was running and a station is not available to start a timer.

SDLC.099

Level: C-INFO

Short Syntax: SDLC.099 nt *network ID*: Station (addr *address*, timeout ID X\' *timeoutId*\') timeout has occurred.

Long Syntax: SDLC.099 Network *network ID*: station (addr X\' *address*\', timeout ID X\' *timeoutId*\') timer has expired without response.

Description: SDLC sent a frame and the time limit to receive a response has expired without receiving the response. In talk 5, under SDLC net, do list link r to see timeout ID meaning.

Chapter 112. Thin Server Disk Task (TSDK)

This chapter describes Thin Server Disk Task (TSDK) messages. For information on message content and how to use the message, refer to the Introduction.

TSDK.001

Level: U-INFO

Short Syntax: TSDK.001 Deleting Host File: *host_file_name* (StagedFile: *cached_file_name*, DiskInfo: *disk_info_file_name*)

Long Syntax: TSDK.001 Deleting Host File: *host_file_name* (StagedFile: *cached_file_name*, DiskInfo: *disk_info_file_name*)

Description: This message is generated when a cached file is deleted from the disk drive.

TSDK.002

Level: UI-ERROR

Short Syntax: TSDK.002 Couldn't schedule *request_type* of *request_name* file (*file_name*), rc: *return_code*(*rc*)

Long Syntax: TSDK.002 Couldn't schedule *request_type* of *request_name* file (*file_name*), rc: *return_code*(*rc*)

Description: This message is an internal error generated when the Disk Task is unable to perform a file request.

TSDK.003

Level: UE-ERROR

Short Syntax: TSDK.003 An error occured while reading *file_name*

Long Syntax: TSDK.003 An error occured while reading *file_name*

Description: This message is generated if the Disk Task is unable to read a file from the hard file.

Cause: Corrupted hard file

Action: Restart the Thin Server Feature. If the error persists, restore the hard file.

TSDK.004

Level: UI-ERROR

Short Syntax: TSDK.004 Mismatch between actual file size (of *file_size*) and expected file size (of *file_size*)

Long Syntax: TSDK.004 Mismatch between actual file size (of *file_size*) and expected file size (of *file_size*)

Description: This message is generated if the files stored on the hard file are inconsistant.

Cause: Corrupted hard file

Action: Restart the Thin Server Feature. If the error persists, restore the hard file.

TSDK.005

Level: U-INFO

Short Syntax: TSDK.005 Owner of *file_name* isn't a configured Master File Server

Long Syntax: TSDK.005 Owner of *file_name* isn't a configured Master File Server

Description: This message is generated if the configuration is changed between NFS and RFS/400. All existing cached files will be deleted.

TSDK.006

Level: C-INFO

Short Syntax: TSDK.006 Found file: *directory/ file_name extension*

Long Syntax: TSDK.006 Found file: *directory/ file_name extension*

Description: This message is generated while the Thin Server is initializing. It is generated for each file found on the hard file. The file names specified are the actual names of the files on the Thin Server's hard file (these names can be found in the first column of the Talk 5 List Cached-Files output).

TSDK.007

Level: C-INFO

Short Syntax: TSDK.007 Adding: *cached_file_name* (*host_file_name*) to cache

Long Syntax: TSDK.007 Adding: *cached_file_name* (*host_file_name*) to cache

Description: This message is generated to indicate that a file stored on the hard file has been added to the set of files actually available to be served.

TSDK.008

Level: U-TRACE

Short Syntax: TSDK.008 *activate_or_terminate* disk task

Long Syntax: TSDK.008 *activate_or_terminate* disk task

Description: This message indicates when the Thin Server disk task is starting or ending.

TSDK.009

Level: ALWAYS

Short Syntax: TSDK.009 Update Finished: *host_file_name*, InTable: *in_table_state*(*value*)

Long Syntax: TSDK.009 Update Finished: *host_file_name*, InTable: *in_table_state*(*value*)

Description: This message is generated when a cached file update is complete. InTable indicates whether the file is still being cached.

TSDK.010

Level: C-INFO

Short Syntax: TSDK.010 Creating File Does Not Exist entry for *host_file_name*

Long Syntax: TSDK.010 Creating File Does Not Exist entry for *host_file_name*

Description: This message is generated when an entry is created for a file which does not exist on the Master File Server.

TSDK.011

Level: UI-ERROR

Short Syntax: TSDK.011 *function* returning *error_code* because *reason*

Long Syntax: TSDK.011 *function* returning *error_code* because *reason*

Description: This internal error is generated when the Disk Task is unable to process a request.

Cause: Couldn't Allocate A Request

Action: The Thin Server Feature ran out of memory while processing requests. Re-start any failed clients. If error persists reduce the number of clients accessing files simultaneously.

TSDK.012

Level: UI-ERROR

Short Syntax: TSDK.012 *function* returning *error_code*

Long Syntax: TSDK.012 *function* returning *error_code*

Description: This message is generated if a Disk Task function fails.

TSDK.013

Level: C-INFO

Short Syntax: TSDK.013 Create entry *file_name*

Long Syntax: TSDK.013 Create entry *file_name*

Description: This message is generated when a new file is added to the cached files on disk.

TSDK.014

Level: C-TRACE

Short Syntax: TSDK.014 Create Files for *host_file_name* (*file_number*), rc: *return_code_description*(*return_code*)

Long Syntax: TSDK.014 Create Files for *host_file_name* (*file_number*), rc: *return_code_description*(*return_code*)

Description: This message is generated when a new file is added to the cached files on disk. The hex file_number relates to the actual file name on the hard file (as shown in the first column of the Talk 5 List Cached-Files output).

TSDK.015

Level: UE-ERROR

Short Syntax: TSDK.015 Couldn't *create_or_access* Thin Server File Cache Directory (*directory*) because *reason*

Long Syntax: TSDK.015 Couldn't *create_or_access* Thin Server File Cache Directory (*directory*) because *reason*

Description: This message is generated when the Thin Server Feature is unable to create or access the cached files directory for some reason. If this occurs, there is some incompatibility with how the hard file has been used and use of the Thin Server Feature. Another possibility is that the hard file has become corrupted.

TSDK.016

Level: U-INFO

Short Syntax: TSDK.016 Attempting to create Thin Server Directory: *directory_name*

Long Syntax: TSDK.016 Attempting to create Thin Server Directory: *directory_name*

Description: This message is generated to indicate that Disk Task is creating the sub-directory used to store cached files on the hard file.

TSDK.017

Level: U-INFO

Short Syntax: TSDK.017 Thin Server Directory Created Successfully

Long Syntax: TSDK.017 Thin Server Directory Created Successfully

Description: This message is generated to indicate that Disk Task is creation of the sub-directory used to store cached files on the hard file succeeded.

TSDK.018

Level: U-INFO

Short Syntax: TSDK.018 Thin Server File Cache Directory Was Empty

Long Syntax: TSDK.018 Thin Server File Cache Directory Was Empty

Description: This message is generated when there are no cached files when the Thin Server Feature starts.

TSDK.019

Level: UE-ERROR

Short Syntax: TSDK.019 The following entry is a duplicate

Long Syntax: TSDK.019 The following entry is a duplicate

Description: This message is generated to indicate that a duplicate file was found in the Thin Server cache. A TSDK.1 message will also be generated indicating that the older file was deleted.

TSDK.020

Level: UE-ERROR

Short Syntax: TSDK.020 Read-Only file found: *file_name*

Long Syntax: TSDK.020 Read-Only file found: *file_name*

Description: This message is generated when a read-only file is found on the hard file in the Thin Server directory. This should never happen and indicates corruption of the hard file.

TSDK.021

Level: UI-ERROR

Short Syntax: TSDK.021 Internal Error: *part_1 part_2*(*number*)

Long Syntax: TSDK.021 Internal Error: *part_1 part_2*(*number*)

Description: This message is generated when a fatal internal error occurs in the Thin Server Disk Task.

TSDK.022

Level: UE-ERROR

Short Syntax: TSDK.022 Abort on: *file_name*, LastByteAvail: *bytes*, FileSize: *bytes*

Long Syntax: TSDK.022 Abort on: *file_name*, LastByteAvail: *bytes*, FileSize: *bytes*

Description: This message is generated if a file update is interrupted (such as by loss of the connection to the Master File Server.

TSDK.023

Level: UI-ERROR

Short Syntax: TSDK.023 Err in obj dtor: *specific_error_message*

Long Syntax: TSDK.023 Error in object destructor: *specific_error_message*

Description: This message is generated if an object destructor detects some invalid condition.

TSDK.024

Level: C-TRACE

Short Syntax: TSDK.024 TRC: *trace_message*

Long Syntax: TSDK.024 TRACE: *trace_message*

Description: This message is generated when Disk Task internal traces are used.

TSDK.025

Level: U-INFO

Short Syntax: TSDK.025 Rmv *file_name* (not in preload)

Long Syntax: TSDK.025 Remove *file_name* (not in preload list or include directories)

Description: This message is generated when Disk Task removes a file from the cache because it is no longer in the pre-load list or include directories.

TSDK.026

Level: ALWAYS

Short Syntax: TSDK.026 Refresh *protocol* files *starting_or_finished*

Long Syntax: TSDK.026 Refresh *protocol* files *starting_or_finished*

Description: This message is generated when a refresh of the files belonging to a particular protocol is starting or completing. It may indicate that the refresh was aborted for some reason. If this occurs, the appropriate client messages should be observed (TSNC for NFS, TSRC for RFS/400).

TSDK.027

Level: UI-ERROR

Short Syntax: TSDK.027 Unexpected RC from *routine* is *error*(*number*) *optional_info*

Long Syntax: TSDK.027 Unexpected RC from *routine* is *error*(*number*) *optional_info*

Description: This message is generated when any Disk Task function will return an unexpected return code

---

TSDK.028

Level: C-TRACE

Short Syntax: TSDK.028 Acceptable RC from *routine* is *error*(*number*) *optional_info*

Long Syntax: TSDK.028 Acceptable RC from *routine* is *error*(*number*) *optional_info*

Description: This message is generated when Disk Task returns one of the following returns codes:

Cause: TSDK_NO_SAME_VERSION

Action: The NFS or RFS/400 client was checking if a file needs updating and the cached file as the same version as on the master file server.

Cause: TSDK_NO_NOT_ON_DISK

Action: The TFTP, NFS, or RFS/400 servers were attempting to access a file which is in the pre-load list, but not currently on the hard file.

Cause: TSDK_NO_NOT_AVAILABLE

Action: The TFTP, NFS, or RFS/400 servers were attempting to access a file which is not in the pre-load list.

Cause: TSDK_NO_NODE_NOT_FOUND

Action: The NFS server is attempting to access a file which is not in the cache.

TSDK.029

Level: UE-ERROR

Short Syntax: TSDK.029 RAM memory cache was configured for *configured_memory*k, using *actual_memory*k

Long Syntax: TSDK.029 RAM memory cache was configured for *configured_memory*k, using *actual_memory*k

Description: This message is generated when the Thin Server feature was not able to use the configured amount of RAM memory cache.

Cause: The amount of memory to use for RAM cache may have been changed and the Thin Server Feature restarted without restarting the whole router.

Cause: More memory has been configured for the RAM cache than available in the router.

TSDK.030

Level: UI-ERROR

Short Syntax: TSDK.030 Invalid Handle, Task: *task_name*(0x *task_id*), Handle: 0x *handle*, rc: *error*(*error_number*)

Long Syntax: TSDK.030 Invalid Handle, Task: *task_name*(0x *task_id*), Handle: 0x *handle*, rc: *error*(*error_number*)

Description: This message is generated any time Disk Task is passed an invalid handle.

TSDK.031

Level: UE-ERROR

Short Syntax: TSDK.031 Aborting Disk Requests for *host_file_name* (FileState: *file_state*)

Long Syntax: TSDK.031 Aborting Disk Requests for *host_file_name* (FileState: *file_state*)

Description: This message is generated if a Refresh, Delete File, Flush, or verification of file version with the Master Server results in a file being deleted while being accessed. This may also happen when the Thin Server feature is Disabled or Restarted.

TSDK.032

Level: CE-ERROR

Short Syntax: TSDK.032 *Canceling_or_Aborting* all Disk Requests for handle: *task*, 0x *handle*

Long Syntax: TSDK.032 *Canceling_or_Aborting* all Disk Requests for handle: *task*, 0x *handle*

Description: This message is generated when a Refresh, Delete File, Flush, or verification of file version with the Master Server results in a file being deleted while being accessed. This may also happen when the Thin Server feature is Disabled or Restarted.

TSDK.033

Level: UE-ERROR

Short Syntax: TSDK.033 No room *cache_or_hard_file* for *host_file_name*

Long Syntax: TSDK.033 No room *cache_or_hard_file* for *host_file_name*

Description: This message is generated when a file is unable to be stored on the Thin Server because there is no more room.

Cause: RAM only mode (no hard file configured)

Action: Make sure the MEMORY-CACHE is SET to a

value somewhat larger than the total space used by all the files to be cached.

Cause: Hard File Configured

Action: Make sure the hard file is operational and being used (check the configuration in Talk 5 for example). If the hard file is being used, make sure there are not too many files being downloaded.

TSDK.034

Level: ALWAYS

Short Syntax: TSDK.034 Thin Server running without hard file because *problem*

Long Syntax: TSDK.034 Thin Server running without hard file because *problem*

Description: This message is generated if some problem causes the thin server to be unable to use the hard file.

Cause: General Problem

Action: Verify that the hard file is not corrupted and that there are not invalid files in the ″TS″ and ″TS/DIR″ directories. All files in these directories should be of the form ″xxxxxxxx.INF″ or ″xxxxxxxx.DAT″, where ″xxxxxxxx″ is a hexadecimal number. The files should also all be read-write.

Cause: Hardfile Not Useable

Action: Make sure a hard file is present. If one is, make sure there is enough free space on it.

Cause: Couldn’t create Thin Server Directory

Action: Make sure there is space on the hard file, and that either the file ″TS″ doesn’t exist, or if so, it is a directory. Also make sure that ″TS/DIR″ is either a directory, or non-existant.

Cause: Couldn’t access Thin Server Directory

Action: Make sure there is a directory ″TS″ on the hard file. Also make sure that ″TS/DIR″ is a directory.

Chapter 113. Thin Server NFS (TSNS)

This chapter describes Thin Server NFS (TSNS) messages. For information on message content and how to use the message, refer to the Introduction.

TSNS.001

Level: UI-ERROR

Short Syntax: TSNS.001 System Error: *error_description*

Long Syntax: TSNS.001 System Error: *error_description*

Description: The NFS server was not able to complete initial setup successfully. Errors occured during portmapper registration or initial socket setup.

TSNS.002

Level: UI-ERROR

Short Syntax: TSNS.002 Unable to send *command-name* reply

Long Syntax: TSNS.002 Unable to send *command-name* reply

Description: An attempt to send a reply to a Network Station failed.

TSNS.003

Level: UI-ERROR

Short Syntax: TSNS.003 Unable to decode *command_name* args

Long Syntax: TSNS.003 Decode failed with *command_name*

Description: An attempt to decode the arguments for a RPC request failed.

TSNS.004

Level: UI-ERROR

Short Syntax: TSNS.004 Memory Allocation error getting *storage_name*

Long Syntax: TSNS.004 Memory Allocation error failed with *storage_name*

Description: The memory allocated for use by the Thin Server feature is depleted

TSNS.005

Level: C-TRACE

Short Syntax: TSNS.005 *function_name* failed with *rc*

Long Syntax: TSNS.005 *function_name* failed with *rc*

Description: A function call failed with the specified return code or errno.

TSNS.006

Level: C-INFO

Short Syntax: TSNS.006 NFSD transport handle is *transport_handle*

Long Syntax: TSNS.006 NFSD transport handle is *transport_handle*

Description: This message displays the NFS transport handle that was defined during NFS Server initialization.

TSNS.007

Level: UI-ERROR

Short Syntax: TSNS.007 System Error: *function_name* errno is *errno*

Long Syntax: TSNS.007 System Error: *function_name* errno is *errno*

Description: A system error occured during an attempt to create an RPC service transport.

TSNS.008

Level: UI-ERROR

Short Syntax: TSNS.008 *function_description* , sock number is *sock_descriptor*, errno *err_number*

Long Syntax: TSNS.008 *function_description* , sock number is *sock_descriptor*, errno *err_number*

Description: The binding of the socket defined by sock_descriptor has failed with the specified errno.

TSNS.009

Level: CE-ERROR

Short Syntax: TSNS.009 Unsupported RPC invoked prog= *program_number*, vers= *version_number*, proc= *procedure_number*

Long Syntax: TSNS.009 Unsupported RPC invoked prog= *program_number*, vers= *version_number*, proc= *procedure_number*

Description: An attempt was made to invoke an unsupported RPC routine

Cause: The NFS client attempted to invoke an unsupported RPC routine.

Action: Check the procedure number specified in the message to determine the routine that was requested and verify its usage. WRITE and CREATE commands are the most likely candidates.

TSNS.010

Level: C-INFO

Short Syntax: TSNS.010 MOUNT from *ip_addr* on directory *directory_name* was successful

Long Syntax: TSNS.010 MOUNT request from *ip_addr* on directory *directory_name* was successful

Description: The client at the specified ip address successfully mounted.

TSNS.011

Level: C-TRACE

Short Syntax: TSNS.011 NFS DISPATCH received prog= *program_number*, vers= *version_number*, proc= *procedure_number* (*procedure_name*)

Long Syntax: TSNS.011 NFS DISPATCH received prog= *program_number*, vers= *version_number*, proc= *procedure_number* (*procedure_name*)

Description: This message indicates that the NFS Dispatch routine was invoked.

TSNS.012

Level: C-TRACE

Short Syntax: TSNS.012 MOUNT DISPATCH received prog= *program_number*, vers= *version_number*, proc= *procedure_number* (*procedure_name*)

Long Syntax: TSNS.012 MOUNT DISPATCH received prog= *program_number*, vers= *version_number*, proc= *procedure_number* (*procedure_name*)

Description: This message indicates that the MOUNT Dispatch routine was invoked.

TSNS.013

Level: P-TRACE

Short Syntax: TSNS.013 *function_name* was invoked

Long Syntax: TSNS.013 *function_name* was invoked

Description: This message indicates that the specified function was invoked

---

TSNS.014

Level: C-TRACE

Short Syntax: TSNS.014 TRC: *message_description*

Long Syntax: TSNS.014 TRACE: *message_description*

Description: Informational message used for debugging purposes.

TSNS.015

Level: UI-ERROR

Short Syntax: TSNS.015 MOS IP Interface Failed to Come Up after *time* seconds

Long Syntax: TSNS.015 MOS IP Interface Failed to Come Up after *time* seconds

Description: This error is generated if the routers IP interface fails to start.

Cause: The IP interface is not configured correctly or a hardware problem exists.

Action: Check the IP configuration definitions and physical connectivity.

TSNS.016

Level: UI-ERROR

Short Syntax: TSNS.016 *function-name* failed with rc: *rc*, error: *error-description*(*error-code*)

Long Syntax: TSNS.016 *function-name* failed with rc: *rc*, error: *error-description*(*error-code*)

Description: A function call failed with the specified error. This may prevent the NFS server thread from starting

---

TSNS.017

Level: UI-ERROR

Short Syntax: TSNS.017 INT ERR: *function-name* failed with *reason*

Long Syntax: TSNS.017 INT ERR: *function-name* failed with *reason*

Description: This error reports a serious internal code problem.

TSNS.018

Level: UI-ERROR

Short Syntax: TSNS.018 INT ERR: *message_description*

Long Syntax: TSNS.018 INT ERR: *message_description*

Description: This error reports that a corrupted packet was detected by the NFS server. The packet will be discared.

TSNS.019

Level: C-TRACE

Short Syntax: TSNS.019 LOOKUP of file *filename* was received

Long Syntax: TSNS.019 LOOKUP of file *filename* was received

Description: This message displays the filename for which a lookup was performed.

TSNS.020

Level: C-INFO

Short Syntax: TSNS.020 LOOKUP by *filename* successful for file/dir ( )

Long Syntax: TSNS.020 LOOKUP by *filename* completed successfully for file/directory ( )

Description: A client issued an NFS Lookup to the Thin Server and the Thin Server responded positively because it is caching the file or directory for which the Lookup was requested

---

TSNS.021

Level: C-INFO

Short Syntax: TSNS.021 LOOKUP by *ipaddr* for (*filename*) discarded, TSF attempting to obtain

Long Syntax: TSNS.021 LOOKUP by *ipaddr* for (*filename*) has been temporarily discarded while TSF attempts to obtain the file/dir

Description: An NFS Lookup request by a client (specified by ipaddr) for the file specified by filename has been discarded while the TSF attempts to obtain the file from the mater file server. The client will retry the Lookup and subsequent retries will be responded to with file available or file not found depending on Thin Server's ability to obtain the file.

TSNS.022

Level: C-INFO

Short Syntax: TSNS.022 LOOKUP by *ipaddr* for (*filename*) - not found because

Long Syntax: TSNS.022 LOOKUP by *ipaddr* for file/directory (*filename*) not found because

Description: The NFS Lookup request by the client for the file or directory called filename has failed.

Cause: The file is a softlink

Action: Create a hardlink on the master file server

Cause: The file does not exist

Action: The client may sometimes ask for files that do not exist. If the file does not exist on the master file server then this should be considered a normal condition (not an error condition) and the client will operate normally.

TSNS.023

Level: C-INFO

Short Syntax: TSNS.023 GETATTR by *ipaddr* for (*filename*) successful

Long Syntax: TSNS.023 GETATTRIBUTE by *ipaddr* for (*filename*) successful

Description: The NFS command GETATTR (Get Attribute) was successful

---

TSNS.024

Level: UI-ERROR

Short Syntax: TSNS.024 GETATTR by *ipaddr* for (*filename*) failed, reason = *reason_code*

Long Syntax: TSNS.024 GETATTRIBUTE by *ipaddr* for (*filename*) failed, reason = *reason_code*

Description: The NFS command GETATTR (Get Attribute) failed for the specified reason

---

TSNS.025

Level: C-INFO

Short Syntax: TSNS.025 Starting to read file *filename* by *ipaddr*

Long Syntax: TSNS.025 Starting to read file *filename* by *ipaddr*

Description: The client is starting to read the specified file (i.e. reading from offset zero)

TSNS.026

Level: C-TRACE

Short Syntax: TSNS.026 READ of offest (*offset*) of file *filename* by *ipaddr* returned *amount_read* bytes

Long Syntax: TSNS.026 READ of offest (*offset*) of file *filename* by *ipaddr* returned *amount_read* bytes

Description: The client is reading the specified file.

TSNS.027

Level: UI-ERROR

Short Syntax: TSNS.027 READ failed for offset (*offset*) of file *filename* by *ipaddr*, reason = *reason_code*

Long Syntax: TSNS.027 READ failed for offset (*offset*) of file *filename* by *ipaddr*, reason = *reason_code*

Description: The read of the file failed for the reason specified

TSNS.028

Level: UI-ERROR

Short Syntax: TSNS.028 ReadDir for *filename* by *ipaddr* failed, reason = *reason_code*

Long Syntax: TSNS.028 ReadDir for *filename* by *ipaddr* failed, reason = *reason_code*

Description: The NFS command Read Directory failed for the specified reason

---

TSNS.029

Level: C-INFO

Short Syntax: TSNS.029 ReadDir for *filename* by *ipaddr* successful

Long Syntax: TSNS.029 ReadDir for *filename* by *ipaddr* successful

Description: The NFS command Read Directory was successful

TSNS.030

Level: C-INFO

Short Syntax: TSNS.030 ReadSymLink for *filename* by *ipaddr* successful

Long Syntax: TSNS.030 ReadSymLink for *filename* by *ipaddr* successful

Description: The NFS command Read Symbolic Link was successful

---

TSNS.031

Level: UI-ERROR

Short Syntax: TSNS.031 ReadSymLink for *filename* by *ipaddr* failed, reason = *reason_code*

Long Syntax: TSNS.031 ReadSymLink for *filename* by *ipaddr* failed, reason = *reason_code*

Description: The NFS command Read Symbolic Link failed

TSNS.032

Level: UI_ERROR

Short Syntax: TSNS.032 MOUNT from *ipaddr* on directory *directory_name* failed

Long Syntax: TSNS.032 MOUNT request from *ipaddr* on directory *directory_name* failed

Description: A Mount request failed

Cause: There is a mismatch between the directories that the Thin Server is configured to cache and the directory that the client is configured to use.

Action: verify that the client is configured correctly.

TSNS.033

Level: C-INFO

Short Syntax: TSNS.033 MOUNT from *ipaddr* on directory *directory_name* failed, max mounts(*max_mounts_allowed*) exceeded

Long Syntax: TSNS.033 MOUNT from *ipaddr* on directory *directory_name* failed, max mounts(*max_mounts_allowed*) exceeded

Description: A mount request failed because the maximum number of mounts allowed by all attached clients has been exceeded

---

TSNS.034

Level: C-INFO

Short Syntax: TSNS.034 UNMOUNT from *ip_addr* on directory *directory_name*

Long Syntax: TSNS.034 UNMOUNT from *ip_addr* on directory *directory_name*

Description: An unmount was received for the specified directory

Chapter 114. Thin Server NFS Client (TSNC)

This chapter describes Thin Server NFS Client (TSNC) messages. For information on message content and how to use the message, refer to the Introduction.

TSNC.001

Level: U-INFO

Short Syntax: TSNC.001 Starting NFS Client full Refresh

Long Syntax: TSNC.001 Starting NFS Client full Refresh

Description: NFS Client has started full refresh

TSNC.002

Level: U-INFO

Short Syntax: TSNC.002 Completing NFS Client full Refresh

Long Syntax: TSNC.002 Completing NFS Client full Refresh

Description: NFS Client has completed full refresh

TSNC.003

Level: U-INFO

Short Syntax: TSNC.003 Refresh for Directory *host_file_name* will be processed (*reason*)

Long Syntax: TSNC.003 Refresh for Directory *host_file_name* will be processed (*reason*)

Description: This message is generated during a refresh to indicate that a sub-directory will be processed and why.

TSNC.004

Level: U-INFO

Short Syntax: TSNC.004 Refresh for Directory *host_file_name* will be skipped (*reason*)

Long Syntax: TSNC.004 Refresh for Directory *host_file_name* will be skipped (*reason*)

Description: This message is generated during a refresh to indicate that a sub-directory will be skipped and why.

TSNC.005

Level: UI-ERROR

Short Syntax: TSNC.005 Invalid file of type *st_mode* (*host_file_name*), mode :

Long Syntax: TSNC.005 Invalid file of type *st_mode* (*host_file_name*), mode :

Description: This message is generated when a file is found during a refresh or individual file update which is not a regular file, symbolic link, or a directory.

TSNC.006

Level: UI-ERROR

Short Syntax: TSNC.006 No memory to process *host_file_name*

Long Syntax: TSNC.006 No memory to process *host_file_name*

Description: This message is generated when the Thin Server Feature is unable to process a file or directory because it ran out of memory.

Cause: The master file server is an NT server with NSM code prior to the August 1998 fix level.

Action: Update the NSM code.

TSNC.007

Level: U-INFO

Short Syntax: TSNC.007 Refresh for File *host_file_name* will be processed

Long Syntax: TSNC.007 Refresh for File *host_file_name* will be processed

Description: This message is generated during a refresh to indicate that a file will be processed.

TSNC.008

Level: U-INFO

Short Syntax: TSNC.008 Refresh for File *host_file_name* is complete

Long Syntax: TSNC.008 Refresh for File *host_file_name* is complete

Description: This message is generated when the update for a file is complete.

TSNC.009

Level: U-INFO

Short Syntax: TSNC.009 Refresh for File *host_file_name* will be skipped (*reason*)

Long Syntax: TSNC.009 Refresh for File *host_file_name* will be skipped (*reason*)

Description: This message is generated during a refresh to indicate that a file will be skipped.

TSNC.010

Level: UI-ERROR

Short Syntax: TSNC.010 OpenFile call for *file_name* returned *error_description*(*error_code*)

Long Syntax: TSNC.010 OpenFile call for *file_name* returned *error_description*(*error_code*)

Description: NFS Client couldn't open a cached copy of host_file_name on hard file.

Cause: TSDK_NO_CANT_WRITE

Action: Too many files are being cached. Make sure sufficient memory is available on the router or adjust the configuration to reduce the number of cached files.

Cause: Any other code

Action: An internal code error has occured.

TSNC.011

Level: UI-ERROR

Short Syntax: TSNC.011 *nfs_proc* call for *host_file_name* returned *error_description*(*error_code*)

Long Syntax: TSNC.011 *nfs_proc* call for *host_file_name* returned *error_description*(*error_code*)

Description: A problem has occured in accessing a file on the master file server.

Cause: NFSERR_PERM, NFSERR_ACCES, MNT_EPERM, MNT_EACCESS

Action: The file being accessed may have permissions set such that the Thin Server does not have permission to access the file. Correct the permissions on the file if the file is intended to be accessible to the Thin Server.

Cause: NFSERR_NOENT, MNT_ENOENT, NFSERR_EXIST

Action: The file doesn't exist on the Master File Server. This may indicate a failure in operation of the NFS or Mount daemons on the Master File Server. Use the REFRESH command in Talk 5 or SNMP to force a new refresh, if the error continues to occur, verify operation of the Master File Server

Cause: NFSERR_IO, NFSERR_NXIO

Action: I/O Errors occured on the Master File Server. Verify correct operation of the Master File Server.

Cause: NFSERR_NOTDIR, NFSERR_ISDIR, NFSERR_STALE

Action: Files were changed on the Master File Server while a refresh or single file update was in progress. Use the REFRESH command in Talk 5 or SNMP to force a new refresh after the file updates on the Master File Server are complete.

Cause: NFS client operation not available

Action: The Master File Server is not functioning correctly, check that the NFS and Mount daemons are running and properly configured.

Cause: NFS client operation timed out

Action: The NFS Client retries have been exhausted due to time outs. Make sure the connection to the Master File Server is operational, and that the Master File Server itself is operational.

Cause: others

Action: Verify correct operation of the Master File Server.

TSNC.012

Level: UI-ERROR

Short Syntax: TSNC.012 Update File for *host_file_name* returned *error_description*(*error_code*)

Long Syntax: TSNC.012 Update File for *host_file_name* returned *error_description*(*error_code*)

Description: The NFS client was unable to update a cached file.

Cause: TSDK_NO_TOO_MANY_NODES or TSDK_NO_CANT_ADD

Action: Too many files are being cached, modify the configuration to reduce the number of files cached. A flush from Talk 5 may also be helpfull to eliminate files which are no longer used.

Cause: TSDK_NO_NO_ROOM

Action: The cached files are using too much space. If a hard file is not used, increase the amount of memory used by the cache. If this is not possible, or a hard file is being used, modify the configuration to reduce the number of files cached. A flush from Talk 5 may also be helpfull to eliminate files which are no longer used.

Cause: Any other error

Action: An internal code error has occured.

TSNC.013

Level: UI-ERROR

Short Syntax: TSNC.013 *file_operation* of *host_file_name* at Offset *file_offset* returned *error_description*(*error_code*)

Long Syntax: TSNC.013 *file_operation* of *host_file_name* at Offset *file_offset* returned *error_description*(*error_code*)

Description: A problem has occured in accessing a file on the master file server. See the causes and actions for TSNC_11.

TSNC.014

Level: UI-ERROR

Short Syntax: TSNC.014 WriteFileRequest call for *host_file_name* returned *error_description*(*error_code*)

Long Syntax: TSNC.014 WriteFileRequest call for *host_file_name* returned *error_description*(*error_code*)

Description: This message is generated when a write to the Thin Server cache fails.

Cause: TSDK_CACHE_WRITE_ERROR

Action: The thin server is not using a hard file and ran out of memory in the cache for the specified file.

Cause: TSDK_SCHED_CANCELED

Action: A restart or other action caused the file update to be interrupted. No action is necessary, the next refresh will recover the file.

Cause: TSDK_DISK_IO_ERROR

Action: A disk IO error occured while writing the file to the hard file. Make sure the hard file is not corrupted.

Cause: Any other error

Action: An internal code error has occured.

TSNC.015

Level: UI-ERROR

Short Syntax: TSNC.015 LookUpNode call for *host_file_name* returned *error_description*(*error_code*)

Long Syntax: TSNC.015 LookUpNode call for *host_file_name* returned *error_description*(*error_code*)

Description: The NFS client was unable to resolve a file name in the thin server cache. This is an internal code error.

TSNC.016

Level: UI-ERROR

Short Syntax: TSNC.016 Resolve for *host_file_name* returned *error_description*(*error_code*)

Long Syntax: TSNC.016 Resolve for *host_file_name* returned *error_description*(*error_code*)

Description: This message is generated when the NFS client is attempting to resolve a file name on the master file server.

Cause: NFSERR_NOENT

Action: The file was being requested from a "Yes, inlcude sub-directories" directory and does not exist on the Master File Server. If the file is supposed to exist on the Master File Server, make sure that the Thin Server has permission to access the file.

Cause: NFSERR_PERM, NFSERR_ACCES, MNT_EPERM, MNT_EACCESS

Action: The file being accessed may have permissions set such that the Thin Server does not have permission to access the file. Correct the permissions on the file if the file is intended to be accessible to the Thin Server.

Cause: MNT_ENOENT, NFSERR_EXIST

Action: The file doesn't exist on the Master File Server. This may indicate a failure in operation of the NFS or Mount daemons on the Master File Server. Use the REFRESH command in Talk 5 or SNMP to force a new refresh, if the error continues to occur, verify operation of the Master File Server

Cause: NFSERR_IO, NFSERR_NXIO

Action: I/O Errors occured on the Master File Server. Verify correct operation of the Master File Server.

Cause: NFSERR_NOTDIR, NFSERR_ISDIR, NFSERR_STALE

Action: Files were changed on the Master File Server while a refresh or single file update was in progress. Use the REFRESH command in Talk 5 or SNMP to force a new refresh after the file updates on the Master File Server are complete.

Cause: NFS client operation not available

Action: The Master File Server is not functioning correctly, check that the NFS and Mount daemons are running and properly configured.

Cause: NFS client operation timed out

Action: The NFS Client retries have been exhausted due to time outs. Make sure the connection to the Master File Server is operational, and that the Master File Server itself is operational.

Cause: TSDK_NO_TOO_MANY_NODES

Action: Too many files are being cached or too many directories exist within the directories configured for caching all files in all sub-directories. Modify the configuration to reduce the number of files or directories cached. A flush from Talk 5 may also be helpfull to eliminate files which are no longer used.

Cause: Any other error

Action: An internal code error has occured.

TSNC.017

Level: UI-ERROR

Short Syntax: TSNC.017 WriteFileRequest call for *host_file_name* returned errno *error_description*(*error_code*)

Long Syntax: TSNC.017 WriteFileRequest call for *host_file_name* returned errno *error_description*(*error_code*)

Description: A disk IO error occured while writing a file into the cache. The hard file may be corrupted.

TSNC.018

Level: UI-ERROR

Short Syntax: TSNC.018 AddDirectory for *host_file_name* returned *error_description*(*error_code*)

Long Syntax: TSNC.018 AddDirectory for *host_file_name* returned *error_description*(*error_code*)

Description: This message is generated when the NFS client is unable to create a directory on the thin server corresponding to a directory on the master file server.

Cause: TSDK_NO_TOO_MANY_NODES

Action: Too many files are being cached or too many directories exist within the directories configured for caching all files in all sub-directories. Modify the configuration to reduce the number of files or directories cached. A flush from Talk 5 may also be helpfull to eliminate files which are no longer used.

Cause: Any other error

Action: An internal code error has occured.

TSNC.019

Level: C-TRACE

Short Syntax: TSNC.019 TRC: *trace_message*

Long Syntax: TSNC.019 TRACE: *trace_message*

Description: This message is generated when NFS Client internal traces are used.

TSNC.020

Level: C-INFO

Short Syntax: TSNC.020 Create Incl Dir *include_directory*, sub-dirs: *local_directory*

Long Syntax: TSNC.020 Create Include Directory *include_directory*, include sub-directories: *local_directory*

Description: This message is generated when the configuration is processed.

TSNC.021

Level: C-INFO

Short Syntax: TSNC.021 RESERVED *x*

Long Syntax: TSNC.021 RESERVED *x*

Description: RESERVED

TSNC.022

Level: UI-ERROR

Short Syntax: TSNC.022 INT ERR: *function* for *object* returned *error_description*(*error_code*)

Long Syntax: TSNC.022 INTERNAL ERROR: *function* for *object* returned *error_description*(*error_code*)

Description: This message is generated when NFS Client internal errors occur.

TSNC.023

Level: C-INFO

Short Syntax: TSNC.023 RESERVED *x*

Long Syntax: TSNC.023 RESERVED *x*

Description: RESERVED

TSNC.024

Level: C-INFO

Short Syntax: TSNC.024 Update check *host_file_name*

Long Syntax: TSNC.024 Update check *host_file_name*

Description: This message is generated when a network station accesses a file. The NFS client will verify that the file is up to date. If the file is in a directory which is part of a tree of directories specified by a configured include directory which includes all sub-directories, the file will be added to the cache if necessary.

TSNC.025

Level: CE-ERROR

Short Syntax: TSNC.025 RESERVED *x*

Long Syntax: TSNC.025 RESERVED *x*

Description: RESERVED

TSNC.026

Level: UE-ERROR

Short Syntax: TSNC.026 RESERVED *x*

Long Syntax: TSNC.026 RESERVED *x*

Description: RESERVED

TSNC.027

Level: UE-ERROR

Short Syntax: TSNC.027 RPC Error in *function* is *error_description*(*error_code*), *additional_description*(*additional_code*, *additional_code*)

Long Syntax: TSNC.027 RPC Error in *function* is

error_description(*error_code*), *additional_description*(*additional_code*, *additional_code*)

Description: This message is generated when an RPC call fails.

Cause: Can't decode result, Incompatible versions of RPC, Server can't decode arguments, Port mapper failure, Program not registered

Action: The Master File Server has returned an invalid response. Make sure the Master File Server is running the correct server software.

Cause: Unable to send, Unable to receive, Unknown host, Unknown protocol

Action: There may be a problem with communication to the Master File Server. Make sure the IP interface being used is operational.

Cause: Timed out

Action: The operation has timed out. A few timeouts are expected. If too many timeouts are occuring, check the connection to the Master File Server.

Cause: Authentication error

Action: Some Master File Servers must be able to resolve a name for the Thin Server. If the Thin Server does not have an entry in a Domain Name Server for the IP address on the interface used to connect to the Master File Server, an entry might be added to the Master File Server's ″hosts″ file.

Cause: Program unavailable, Program/version mismatch, Procedure unavailable

Action: The Port Mapper, NFS, or Mount daemons are not running on the Master File Server. Make sure they are running and properly configured.

Cause: Remote system error

Action: Some kind of error has occurred on the Master File Server. Make sure the Master File Server is operating correctly.

Cause: Others

Action: An internal error has occurred.

TSNC.028

Level: UE-ERROR

Short Syntax: TSNC.028 Include directory *directory* could not be resolved because *reason*

Long Syntax: TSNC.028 Include directory *directory* could not be resolved because *reason*

Description: This message is generated when one of the configured include directories is unable to be resolved to a valid mount exported by the master file server.

Cause: No nfs directories exported

Action: Make sure the master file server is exporting the required directories, and that the thin server has access to them

Cause: No export matched

Action: Make sure that the include directories are properly configured and that the correct master file server is configured. Also note that file names are cause sensitive.

TSNC.029

Level: UI-ERROR

Short Syntax: TSNC.029 NFS Client initialization failed because *function* returned *error_description*(*error_code*)

Long Syntax: TSNC.029 NFS Client initialization failed because *function* returned *error_description*(*error_code*)

Description: This error is recorded if the Thin Server NFS Client fails to initialize because of an internal error.

TSNC.030

Level: UE-ERROR

Short Syntax: TSNC.030 Failed to connect to Master File Server *daemon* daemon, error *error_description*

Long Syntax: TSNC.030 Failed to connect to Master File Server *daemon* daemon, error *error_description*

Description: This error is reported when the NFS fails to connect to either the <ount or NFS daemons on the Master File Server.

TSNC.031

Level: UE-ERROR

Short Syntax: TSNC.031 No exported directories on master server

Long Syntax: TSNC.031 No exported directories on master server

Description: This error is reported when the master server has no exported directories.

TSNC.032

Level: U-INFO

Short Syntax: TSNC.032 *function* failed, retrying *count* of *maximum* times

Long Syntax: TSNC.032 *function* failed, retrying *count* of *maximum* times

Description: This error is reported to indicate that the NFS Client is retrying an operation.

Chapter 115. Thin Server RFS Client (TSRC)

This chapter describes Thin Server RFS Client (TSRC) messages. For information on message content and how to use the message, refer to the Introduction.

TSRC.001

Level: UI-ERROR

Short Syntax: TSRC.001 *generic_string*

Long Syntax: TSRC.001 *generic_string*

Description: This is a generic RFS client error message that reports the specfied error.

TSRC.004

Level: UI-ERROR

Short Syntax: TSRC.004 ERROR: (errno= *error_number*): *error_number_string*

Long Syntax: TSRC.004 ERROR: (errno= *error_number*): *error_number_string*

Description: This is a generic RFS client error message that reports the string explanation associated with a given error number.

TSRC.005

Level: UI-ERROR

Short Syntax: TSRC.005 *function_name*() call failed rc= *return_code* errno= *error_number*

Long Syntax: TSRC.005 *function_name*() call failed rc= *return_code* errno= *error_number*

Description: The specified function call failed. The return code from the failed function call and the errno at the time of failure are indicated.

TSRC.006

Level: UI-ERROR

Short Syntax: TSRC.006 *function_name*() call failed rc= *return_code*

Long Syntax: TSRC.006 *function_name*() call failed rc= *return_code*

Description: The specified function call failed. The return code from the failed function is indicated.

TSRC.007

Level: UI-ERROR

Short Syntax: TSRC.007 Bad RC on *function_name* rc= *return_code* errno= *error_number*

Long Syntax: TSRC.007 Bad return code on *function_name* rc= *return_code* errno= *error_number*

Description: The specified function_name returned a bad return code. The return code from the failed function call and the errno at the time of failure are indicated.

TSRC.008

Level: UI-ERROR

Short Syntax: TSRC.008 ERROR: Bad RC from *function_name* (rc: *return_code*)

Long Syntax: TSRC.008 ERROR: Bad return code from *function_name* (rc: *return_code*)

Description: The specified function call returned a bad return code. The return code received from the specified function call is shown.

TSRC.009

Level: C-INFO

Short Syntax: TSRC.009 *generic_string*

Long Syntax: TSRC.009 *generic_string*

Description: This is a generic RFS informational message used to print any informational text string.

TSRC.015

Level: C-INFO

Short Syntax: TSRC.015 INFO: About to enter: *function_name*

Long Syntax: TSRC.015 INFO: About to enter: *function_name*

Description: This informational message indicates that the specified function is about to be invoked.

TSRC.016

Level: C-INFO

Short Syntax: TSRC.016 Starting func *function_name*()

Long Syntax: TSRC.016 Starting function *function_name*()

Description: This message is generated at the start of the specified function. It indicates that the we are currently inside the specified routine.

TSRC.017

Level: C-INFO

Short Syntax: TSRC.017 The *function_name* call completed ok

Long Syntax: TSRC.017 The *function_name* call completed successfully

Description: The specified function call completed successfully.

TSRC.018

Level: C-INFO

Short Syntax: TSRC.018 RFS client thread is terminating

Long Syntax: TSRC.018 RFS client thread is terminating

Description: The RFS client thread is terminating.

Cause: This will most commonly occur when the user initiates a manual disable or restart of the thin-server feature. The detection of an unrecoverable exceptional error condition may also cause the RFS client thread to terminate.

TSRC.019

Level: UE-ERROR

Short Syntax: TSRC.019 ERROR: Rply is NOT *expected_reply_id* (ReqRplyID: *received_reply_id*)

Long Syntax: TSRC.019 ERROR: Reply is NOT *expected_reply_id* (ReqRplyID: *received_reply_id*)

Description: The RFS client received an unexpected reply from the master RFS file server. The expected and received identifiers are shown.

TSRC.020

Level: UE-ERROR

Short Syntax: TSRC.020 ERROR: *rfs_reply_msg* rply does NOT have a matching correlation id (req cid: *request_cid* rply cid: *reply_cid*)

Long Syntax: TSRC.020 ERROR: *rfs_reply_msg* reply does NOT have a matching correlation id (request cid: *request_cid* reply cid: *reply_cid*)

Description: The RFS client received an RFS reply that did NOT match the correlation identifier that was sent on the last request.

TSRC.021

Level: UI-ERROR

Short Syntax: TSRC.021 ERROR: on *rfs_request_msg* req write() (rc= *return_code* errno= *error_number*)

Long Syntax: TSRC.021 ERROR: on *rfs_request_msg* request write() (rc= *return_code* errno= *error_number*)

Description: An error occurred while writing the specified RFS client request command to the master RFS file server socket stream. The return code from the write() and the thread-specific errno at the time the error was detected are both shown.

TSRC.022

Level: UI-ERROR

Short Syntax: TSRC.022 ERROR: *rfs_request_msg* rply read() failed (rc= *return_code* errno= *error_number*)

Long Syntax: TSRC.022 ERROR: *rfs_request_msg* reply read() failed (rc= *return_code* errno= *error_number*)

Description: An error occurred while reading the specified RFS client reply from the master RFS file server socket stream. The return code from the read() and the thread-specific errno at the time the error was detected are both shown.

TSRC.023

Level: UE-ERROR

Short Syntax: TSRC.023 ERROR: *rfs_reply_msg* rply is too small (bytes read: *num_bytes_read*)

Long Syntax: TSRC.023 ERROR: *rfs_reply_msg* reply is too small (bytes read: *num_bytes_read*)

Description: The RFS client received a RFS reply which was smaller than expected. The number of bytes actually read is shown.

TSRC.024

Level: CE-ERROR

Short Syntax: TSRC.024 ERROR: File *pathname* not found on master RFS file server

Long Syntax: TSRC.024 ERROR: File *pathname* not found on master RFS file server

Description: The specified file was not found on the master RFS file server.

Cause: The specified file is not located on the master RFS file server.

Action: Place the specified file on the master RFS file server.

Cause: The specified file is in the preload list, but is not required for the thin-server cache.

Action: Remove the specified file from the preload list.

TSRC.025

Level: UE-ERROR

Short Syntax: TSRC.025 ERROR: *rfs_reply_msg* rply has a trunc header (bytes read: *num_bytes_read*)

Long Syntax: TSRC.025 ERROR: *rfs_reply_msg* reply has a truncated header (bytes read: *num_bytes_read*)

Description: The RFS client received an RFS reply from the master RFS file server with a truncated header.

TSRC.026

Level: UE-ERROR

Short Syntax: TSRC.026 ERROR: Unexpected rply to *rfs_reply_msg* req (ReqRplyId: *reply_id*)

Long Syntax: TSRC.026 ERROR: Unexpected reply to *rfs_reply_msg* request (ReqRplyId: *reply_id*)

Description: The RFS client received an unexpected RFS reply to the specified RFS request from the master RFS file server.

Cause: (for OpenNode request): The RFS client does not have sufficient permissions to retrieve the file from the master RFS file server.

Action: Change the permissions of the associated file on the master RFS file server to allow userid QTFTP to access the file.

TSRC.027

Level: C-INFO

Short Syntax: TSRC.027 Received portmapper resp (*response_character*) port: *port_number*

Long Syntax: TSRC.027 Received portmapper response (*response_character*) port: *port_number*

Description: The RFS client received the specified portmapper response from the master RFS file server portmapper daemon.

TSRC.028

Level: C-INFO

Short Syntax: TSRC.028 *rfs_request_msg* req sent ok (*num_bytes_sent* bytes sent)

Long Syntax: TSRC.028 *rfs_request_msg* request sent successfully (*num_bytes_sent* bytes sent)

Description: The RFS client successfully sent the specified RFS request to the master RFS file server

TSRC.029

Level: C-INFO

Short Syntax: TSRC.029 RC from *rfs_reply_msg* is ok

Long Syntax: TSRC.029 Return Code from *rfs_reply_msg* is ok

Description: The return code the RFS client received from the specified function call was ok.

TSRC.030

Level: C-INFO

Short Syntax: TSRC.030 *rfs_reply_msg* rply looks ok (*num_bytes_received* bytes received)

Long Syntax: TSRC.030 *rfs_reply_msg* reply looks ok (*num_bytes_received* bytes received)

Description: The RFS client received the expected number of bytes for the indicated reply from the master RFS file server. The type of reply is specified along with the number of bytes received.

TSRC.031

Level: C-INFO

Short Syntax: TSRC.031 RC from DS_RC rply: *return_code_in_msg*

Long Syntax: TSRC.031 Return Code contained in DS_RC reply: *return_code_in_msg*

Description: The specified Return Code was returned in a DS_RC RFS reply from the master RFS file server.

Cause: A DS_RC is a Return Code Data Structure used by the RFS file server to return a return code the the RFS client in response to an RFS request. Additional information on this return code may be found in the RFS specification.

TSRC.032

Level: C-INFO

Short Syntax: TSRC.032 Remote/local timestamp match; thus file *pathname* will not be downloaded.

Long Syntax: TSRC.032 Remote/local timestamp match; thus file *pathname* will not be downloaded.

Description: The timestamp of the cache entry for the specified file in the cache matches the timestamp of the file on the master RFS file server. Since the cache entry for this file is up-to-date, the file will not be downloaded.

TSRC.033

Level: C-INFO

Short Syntax: TSRC.033 Remote/local timestamp mismatch; thus file *pathname* will be downloaded.

Long Syntax: TSRC.033 Remote/local timestamp mismatch; thus file *pathname* will be downloaded.

Description: The timestamp of the cache entry for the specified file in the cache does NOT match the timestamp on the file on the master RFS file server. Since the cache entry for this file is not up-to-date, the file WILL be downloaded.

TSRC.034

Level: UE-ERROR

Short Syntax: TSRC.034 ERROR: StartServerReply reported bad primary RC: *primary_return_code*

Long Syntax: TSRC.034 ERROR: StartServerReply reported bad primary return code: *primary_return_code*

Description: The RFS client received a StartServerReply message from the master RFS server with a bad primary return code.

Cause: The RFS file server was NOT successfully started on the master server.

Action: Check the error log of the master server for further information on this error condition and for appropriate action.

TSRC.035

Level: C-INFO

Short Syntax: TSRC.035 StartServerReply primary RC indicates success

Long Syntax: TSRC.035 StartServerReply primary return code indicates success

Description: The RFS client received a StartServerReply message with a successful primary return code.

TSRC.036

Level: C-INFO

Short Syntax: TSRC.036 File length returned on Lookup rply: *file_length*

Long Syntax: TSRC.036 File length returned on Lookup reply: *file_length*

Description: The RFS client received a Lookup reply from the master RFS file server with the specified file length.

TSRC.037

Level: C-INFO

Short Syntax: TSRC.037 Checking state of master RFS file server connection

Long Syntax: TSRC.037 Checking state of master RFS file server connection

Description: The RFS client is checking the state of its RFS master file server connection.

TSRC.038

Level: C-INFO

Short Syntax: TSRC.038 Establishing communication with master RFS file server

Long Syntax: TSRC.038 Establishing communication with master RFS file server

Description: The RFS client is establishing communication with the master RFS file server.

TSRC.039

Level: C-INFO

Short Syntax: TSRC.039 Disconnected from master RFS file server

Long Syntax: TSRC.039 Disconnected from master RFS file server

Description: The RFS client has disconnected from the master RFS file server.

TSRC.040

Level: C-INFO

Short Syntax: TSRC.040 *handle_name* Handle: *handle_value*

Long Syntax: TSRC.040 *handle_name* Handle: *handle_value*

Description: The RFS client is reporting the value of the specified handle. The master RFS file server returns both Lookup and Open file handles.

TSRC.041

Level: C-INFO

Short Syntax: TSRC.041 File handle returned on *parameter_name* rply: *reply_name*

Long Syntax: TSRC.041 File handle returned on *parameter_name* reply: *reply_name*

Description: The RFS client is reporting the value of the handle it received from the specified reply (from the master RFS file server).

TSRC.042

Level: C-INFO

Short Syntax: TSRC.042 *reply_name* has correct syntax

Long Syntax: TSRC.042 *reply_name* has correct syntax

Description: The RFS client is reporting that the specified reply has correct syntax.

TSRC.043

Level: UI-ERROR

Short Syntax: TSRC.043 Failed ASCII->UNICODE *variable_name* conversion of *ascii_string*; conv_rc= *conversion_return_code*

Long Syntax: TSRC.043 Failed ASCII->UNICODE *variable_name* conversion of *ascii_string*; conv_rc= *conversion_return_code*

Description: The ASCII->UNICODE conversion of the given ASCII string failed with the specified return code.

TSRC.044

Level: C-INFO

Short Syntax: TSRC.044 Successful ASCII->UNICODE *variable_name* conversion of *ascii_string*

Long Syntax: TSRC.044 Successful ASCII->UNICODE *variable_name* conversion of *ascii_string*

Description: The ASCII->UNICODE conversion of the given ASCII string was successful.

TSRC.045

Level: CE-ERROR

Short Syntax: TSRC.045 ERROR: Unable to connect to master RFS server

Long Syntax: TSRC.045 ERROR: Unable to connect to master RFS server

Description: The RFS client is unable to connect to the master RFS file server.

Cause: This is often a symptom of a network problem or a symptom that the master RFS server daemon is not running.

Action: Verify connection with the master RFS server (for example, using ping or traceroute). Verify the master RFS server's subsystem processes are running. If the problem persists, try stopping and restarting the RFS server's subsystem processes on the master server.

TSRC.046

Level: CE-ERROR

Short Syntax: TSRC.046 Aborted Connection Retry: Exceeded the maximum number of retry attempts

Long Syntax: TSRC.046 Aborted Connection Retry: Exceeded the maximum number of retry attempts

Description: The RFS client has tried multiple times to connect with the master RFS file server and had given up.

Cause: This is often a symptom of a network problem or a symptom that the master RFS file server daemon is not running.

Action: Verify connection with the master RFS file server (for example, using ping or traceroute). Verify the master RFS file server's subsystem processes are running (for example, using wrksbs).

Action: If the problem persists, try stopping and restarting the RFS file server and TCPIP subsystem processes on the (AS/400) master server.

Action: If problem persists, restart the router's thin-server feature.

Action: If problem persists, restart the router.

TSRC.047

Level: UI-ERROR

Short Syntax: TSRC.047 Warning: RFS Server connection status is not valid

Long Syntax: TSRC.047 Warning: RFS Server connection status is not valid

Description: The RFS client has detected an invalid RFS server connection status.

TSRC.048

Level: UI-ERROR

Short Syntax: TSRC.048 seedrply is not valid

Long Syntax: TSRC.048 seedrply is not valid

Description: The RFS client detected an invalid seedrply

TSRC.049

Level: C-INFO

Short Syntax: TSRC.049 Attempting to connect to *ip_address* (port *port_number*)

Long Syntax: TSRC.049 Attempting to connect to *ip_address* (port *port_number*)

Description: The RFS client is attempting to connect to the master server at the specified IP address and port number.

TSRC.050

Level: C-INFO

Short Syntax: TSRC.050 Connected to the AS/400 master RFS file server (as-file) thread

Long Syntax: TSRC.050 Connected to the AS/400 master RFS file server (as-file) thread

Description: The RFS client has successfully connected to the AS/400's master RFS file server (as-file) thread.

TSRC.051

Level: C-INFO

Short Syntax: TSRC.051 Initializing *socket_descriptor* to port *port_number*, family *family_number*

Long Syntax: TSRC.051 Initializing *socket_descriptor* to port *port_number*, family *family_number*

Description: The RFS client is initializing the indicated socket descriptor to the specified port and family numbers.

TSRC.052

Level: C-INFO

Short Syntax: TSRC.052 Sending req to obtain portnumber for: *server_process*

Long Syntax: TSRC.052 Sending request to obtain portnumber for: *server_process*

Description: The RFS client is issuing a request to the AS/400's port mapper to obtain the portnumber for the specified server process.

TSRC.053

Level: C-INFO

Short Syntax: TSRC.053 Connected to the AS/400 portmapper server thread

Long Syntax: TSRC.053 Connected to the AS/400 portmapper server thread

Description: The RFS client has successfully connected to the AS/400's portmapper server thread.

TSRC.054

Level: C-INFO

Short Syntax: TSRC.054 INFO: Value of refresh timeout: *num_sec* sec *num_nsec* nsec

Long Syntax: TSRC.054 INFO: Value of refresh timeout: *num_sec* sec *num_nsec* nsec

Description: The RFS client is reporting the current value of the refresh timeout variable. This variable is used to calculate the time remaining until the next automatic refresh of files in the preload list.

TSRC.055

Level: C-INFO

Short Syntax: TSRC.055 Refreshing file: *pathname*

Long Syntax: TSRC.055 Refreshing file: *pathname*

Description: The RFS client is refreshing the specified file.

TSRC.056

Level: C-INFO

Short Syntax: TSRC.056 Refresh timer expired

Long Syntax: TSRC.056 Refresh timer expired

Description: The RFS client refresh timer expired. If periodic refreshing is enabled, the files in the preload list will now be refreshed. See additional messages below to determine whether the preload list files are actually being refreshed.

TSRC.057

Level: C-INFO

Short Syntax: TSRC.057 (time_of_refresh == 0), NOT sending refresh msg

Long Syntax: TSRC.057 (time_of_refresh == 0), NOT sending refresh message

Description: The time_of_refresh parameter indicates files should NOT be periodically refreshed. The preload list file will not be refreshed.

TSRC.058

Level: C-INFO

Short Syntax: TSRC.058 *action_name* refresh msg

Long Syntax: TSRC.058 *action_name* refresh message

Description: A thread is performing the specified action with a refresh message to the RFS client thread.

TSRC.059

Level: UI-ERROR

Short Syntax: TSRC.059 ERROR: Received UNRECOGNIZED msg (in mq) - Discarding

Long Syntax: TSRC.059 ERROR: Received UNRECOGNIZED message (in message queue) - Discarding

Description: The RFS client received an unrecognized message from its message queue. The message will be discarded and RFS client processing will continue as normal.

TSRC.060

Level: UI-ERROR

Short Syntax: TSRC.060 Unable to *action_name* RFS client (pr_mq) mq rc= *return_code* errno= *error_number*

Long Syntax: TSRC.060 Unable to *action_name* RFS client (pr_mq) message queue rc= *return_code* errno= *error_number*

Description: The RFS client was unable to perform the specified action on the RFS client (pr_mq) message queue. The return code from the failed operation and the thread-specific error number at the time of failure are shown.

TSRC.061

Level: C-INFO

Short Syntax: TSRC.061 INFO: Received *message_name* msg (in mq)

Long Syntax: TSRC.061 INFO: Received *message_name* message (in mq)

Description: The RFS client is reporting the receipt of a message from its pr_mq message queue.

TSRC.062

Level: C-INFO

Short Syntax: TSRC.062 Either received msg, signal, or timer expired

Long Syntax: TSRC.062 Either received message, signal, or timer expired

Description: The RFS client detected that there is work to do as a result of either a message on its message queue, a signal, or an expiration of the refresh timer.

TSRC.063

Level: C-INFO

Short Syntax: TSRC.063 Exited mq_timedreceive() with: rc= *return_code* errno= *error_number*

Long Syntax: TSRC.063 Exited mq_timedreceive() with: rc= *return_code* errno= *error_number*

Description: The RFS client exited its block mq_timedreceive routine with the indicated return code. The current value of errno for the RFS client thread is also displayed.

TSRC.064

Level: C-INFO

Short Syntax: TSRC.064 Processing msg (received from mq)

Long Syntax: TSRC.064 Processing message (received from message queue)

Description: The RFS client is processing the message it received from its message queue.

TSRC.065

Level: C-INFO

Short Syntax: TSRC.065 Msg successfully sent to RFS client

Long Syntax: TSRC.065 Message successfully sent to RFS client

Description: A thread has successfully sent a message to the RFS client's message queue.

TSRC.066

Level: C-INFO

Short Syntax: TSRC.066 RFS client (pr_mq) mq *action_name*

Long Syntax: TSRC.066 RFS client (pr_mq) mq *action_name*

Description: The indicated action is occurring to the RFS client (pr_mq) message queue.

TSRC.067

Level: UI-ERROR

Short Syntax: TSRC.067 ERROR: mqtimedreceive() call failed - No msg of desired type rc= *return_code* local_errno= *error_number*

Long Syntax: TSRC.067 ERROR: mqtimedreceive() call failed - No message of desired type rc= *return_code* local_errno= *error_number*

Description: The mqtimedreceive call failed because it did NOT find a message of the correct type in its pr_mq message queue.

Cause: A thread has sent an invalid message directly to the RFS client pr_mq message queue.

Action: The thread should use the RFS client interface multi-thread safe function call to send messages to the RFS client.

Cause: The RFS client has been triggered to look at its pr_mq message buffer for a message when no message is there.

Action: Save a dump of all TSRC ELS messages leading up to this error and report this error to your system administrator.

TSRC.068

Level: UI-ERROR

Short Syntax: TSRC.068 ERROR: Msg on pr_mq mq exceeds mq_recv_msg_buf size

Long Syntax: TSRC.068 ERROR: Message on pr_mq message queue exceeds mq_recv_msg_buf size

Description: The message being received from the pr_mq message queue exceeds mq_recv_msg_buf size.

Cause: A thread is sending a message to the RFS client thread which is larger than the RFS client thread's receive message buffer. The offending thread may be sending garbage data to the RFS client.

Action: For users: Contact your system administrator.

Action: For developers: The thread should ensure it is using the RFS client interface multi-thread safe function call to send messages to the RFS client. If the message is valid, either the thread should reduce the amount of data it is sending to the RFS client OR the RFS client should increase the size of its receive buffer to accomodate the incoming message.

TSRC.069

Level: UI-ERROR

Short Syntax: TSRC.069 ERROR: *semaphore_name* Sem creat failed rc= *return_code* errno= *error_number*

Long Syntax: TSRC.069 ERROR: *semaphore_name* Semaphore creation failed rc= *return_code* errno= *error_number*

Description: The RFS client failed to create the specified semaphore. The return code from the failed call and the thread-specific errno at the time of failure are shown.

TSRC.070

Level: C-INFO

Short Syntax: TSRC.070 Creating sems

Long Syntax: TSRC.070 Creating semaphores

Description: The RFS client is creating its semaphores.

TSRC.071

Level: UI-ERROR

Short Syntax: TSRC.071 ERROR: Bad disk_io_parms.Status returned to WriteFileRequestCallbackFunc() by DiskTask; Status= *disk_io_parms_status*, Errno= *disk_io_parms_errno*

Long Syntax: TSRC.071 ERROR: Bad disk_io_parms.Status returned to WriteFileRequestCallbackFunc() by DiskTask; Status= *disk_io_parms_status*, Errno= *disk_io_parms_errno*

Description: The RFS client received a disk_io_parms block from DiskTask with a bad disk_io_parms Status. The Status and Errno from the disk_io_parms structure are reported. Consult the DiskTask (TSDK) ELS messages for more information on the error condition and for the appropriate action to take.

TSRC.072

Level: UI-ERROR

Short Syntax: TSRC.072 ERROR: Stopping reading the preload list due to bad disk_io_parms.Status returned from DiskTask: *disk_io_parms_status*

Long Syntax: TSRC.072 ERROR: Stopping reading the preload list due to bad disk_io_parms.Status returned from DiskTask: *disk_io_parms_status*

Description: The RFS client is terminating the ReadFileData() while loop due to a bad disk_io_parms.Status returned by DiskTask. The bad Status value is shown.

Cause: The DiskTask ReadFileData() interface function call failed.

Action: Perform a thin-server refresh.

Action: If problem persists, consult the DiskTask (TSDK) ELS messages for more information on the error condition and for the appropriate action to take.

TSRC.073

Level: C-INFO

Short Syntax: TSRC.073 UpdateFileInfo() returned a RC: rc= *return_code*

Long Syntax: TSRC.073 UpdateFileInfo() returned a return code: rc= *return_code*

Description: The RFS client is reporting that the UpdateFileInfo() interface function call to DiskTask returned with the specified return code. Refer to the thin-server DiskTask documentation for more information on this return code.

TSRC.074

Level: C-INFO

Short Syntax: TSRC.074 disk_io_parms.Errno returned by DiskTask = *disk_io_parms_errno*

Long Syntax: TSRC.074 disk_io_parms.Errno returned by DiskTask = *disk_io_parms_errno*

Description: The RFS client is reporting the disk_io_parms.Errno that was returned by DiskTask. Refer to the thin-server DiskTask documentation for more information on this return code.

TSRC.075

Level: C-INFO

Short Syntax: TSRC.075 Exiting ReadFileData() while loop due to ActualLen: *actual_length_value*

Long Syntax: TSRC.075 Exiting ReadFileData() while loop due to ActualLen: *actual_length_value*

Description: The RFS client is reporting that it exitied the ActualLen while loop (which stores a file to the cache) due to there being no more bytes left in the file. A non-zero ActualLen value specified here is cause for concern.

Cause: During normal operation, this message will report value=0 A non-zero value indicates there is a problem.

Action: If a non-zero value is observerd, save all TSRC and TSDK error ELS messages and report the problem to your system administrator.

TSRC.076

Level: UI-ERROR

Short Syntax: TSRC.076 ERROR: *action_name* file: *pathname*

Long Syntax: TSRC.076 ERROR: *action_name* file: *pathname*

Description: The specified file operation failed on the specified file.

TSRC.077

Level: C-INFO

Short Syntax: TSRC.077 Attempting to open local file on disk: *pathname*

Long Syntax: TSRC.077 Attempting to open local file on disk: *pathname*

Description: The RFS client is attempting to open the specified file on the local hard disk.

TSRC.078

Level: C-INFO

Short Syntax: TSRC.078 Local file write() was successful; bytes written: *num_bytes_written*

Long Syntax: TSRC.078 Local file write() was successful; bytes written: *num_bytes_written*

Description: The RFS client successful wrote the specified number of bytes to the local hard disk.

TSRC.079

Level: C-INFO

Short Syntax: TSRC.079 INFO: Filename: *pathname*

Long Syntax: TSRC.079 INFO: Filename: *pathname*

Description: The RFS client is reporting a specific filename associated with the previous ELS message.

TSRC.080

Level: P-TRACE

Short Syntax: TSRC.080 File data to temp buffer read() looks ok; bytes received: *num_bytes_received*, bytes remaining: *num_bytes_remaining*

Long Syntax: TSRC.080 File data to temp buffer read() looks ok; bytes received: *num_bytes_received*, bytes remaining: *num_bytes_remaining*

Description: The RFS client is downloading a file from the master RFS file server. This message reports each packet that the RFS client reads from the socket stream to the thin-server cache.

TSRC.081

Level: C-INFO

Short Syntax: TSRC.081 *counter_config* value *variable_name = value*

Long Syntax: TSRC.081 *counter_config* value *variable_name = value*

Description: The value of the specified counter/configuration parameter is shown.

TSRC.082

Level: C-INFO

Short Syntax: TSRC.082 *counter_config* value *variable_name = value_string*

Long Syntax: TSRC.082 *counter_config* value *variable_name = value_string*

Description: The value of the specified counter/configuration parameter string is shown.

TSRC.083

Level: C-INFO

Short Syntax: TSRC.083 *function_name*() call failed rc= *return_code*

Long Syntax: TSRC.083 *function_name*() call failed rc= *return_code*

Description: The specified function call failed. The return code from the failed function is indicated. This message does not necessarily indicate that an error

occurred. This message occurs during the course of normal operations.

Cause: The RFS client reached the end of the preload list and tried to read past the end of the file.

Action: Take no action. This is a normally occurring message.

Chapter 116. Thin Server RFS Server (TSRS)

This chapter describes Thin Server RFS Server (TSRS) messages. For information on message content and how to use the message, refer to the Introduction.

TSRS.001

Level: C-TRACE

Short Syntax: TSRS.001 Start func *function_name*, sh = *socket_handle*

Long Syntax: TSRS.001 Starting function *function_name*, socket handle = *socket_handle*

Description: This message is generated at the start of each function.

TSRS.002

Level: CI-ERROR

Short Syntax: TSRS.002 *allocating_or_deallocating ctl_blk_name.* old (*current_number*), new (*new_number*)

Long Syntax: TSRS.002 *allocating_or_deallocating ctl_blk_name.* old amount(*current_number*), new amount(*new_number*)

Description: This message is used to indicate that we are attempting to acquire more resources in order to fulfill a request or we are freeing resources which we currently do not need.

TSRS.003

Level: C-TRACE

Short Syntax: TSRS.003 Start func *function_name*

Long Syntax: TSRS.003 Starting function *function_name*

Description: This message is generated at the start of a function.

TSRS.004

Level: UI-ERROR

Short Syntax: TSRS.004 Func *function_name* failed, rc = *return_code*

Long Syntax: TSRS.004 Function *function_name* failed with rc = *return_code*

Description: A function returned an unexpected return code.

TSRS.005

Level: P-TRACE

Short Syntax: TSRS.005 Func *function_name* compl, rc = *return_code*

Long Syntax: TSRS.005 Function *function_name* completed. rc = *return_code*

Description: A function completed successfully and returned the value specified.

TSRS.006

Level: UI-ERROR

Short Syntax: TSRS.006 Alloc failed for *control_block_name*

Long Syntax: TSRS.006 Allocate failed for *control_block_name*

Description: There was not enough memory to allocate the specified control block.

Cause: Not enough memory available to support the number of clients.

Action: Reduce the number of clients or increase the amount of memory in the device.

TSRS.007

Level: UI-ERROR

Short Syntax: TSRS.007 NS ctl blk not found. corrID = *corr_id*

Long Syntax: TSRS.007 NS control block not found. correlation ID received = *corr_id*

Description: The TSF maintains information about each client which connects to it. Each time a client makes a request, the TSF needs to find its information, in this case, that information was not found. The corr_id is from the data frame which was received.

TSRS.008

Level: C-INFO

Short Syntax: TSRS.008 Mas file svr *rfs_or_login* port not started

Long Syntax: TSRS.008 Master file server *rfs_or_login* port not started

Description: The attempt to establish the connection

to the master file server was unsuccessful. This connection is necessary to properly validate that the thin server has the same version of the file as the master file server. Clients powered on when no master file server connection exists will get the version of the file which the Thin Server currently has.

Cause: Incorrect master file server IP address

Action: Verify that the master file server address is correct

Cause: No path to the master file server

Action: Verify that an IP path exists between the master file server and the client.

TSRS.009

Level: C-INFO

Short Syntax: TSRS.009 Mas file svr *rfs_or_login* port started

Long Syntax: TSRS.009 Master file server *rfs_or_login* port started

Description: The connection to the master file server has been successfully established. The thin server will validate that it has the correct version of a file each time a file is opened. Also, if the Thin Server does not have a particular file, the request will be relayed to the master file server to be processed.

TSRS.010

Level: P-TRACE

Short Syntax: TSRS.010 Port mapper rply, use port *port_number*

Long Syntax: TSRS.010 Port mapper reply directing client to port *port_number*

Description: A client has contacted the thin server using RFS and the thin server has directed the client to the specified port for rfs or login activity.

TSRS.011

Level: P-TRACE

Short Syntax: TSRS.011 Port mapper req *request_string* is invalid

Long Syntax: TSRS.011 Port mapper request *request_string* is invalid

Description: A client made an invalid request of the port mapper.

TSRS.012

Level: P-TRACE

Short Syntax: TSRS.012 RFS *cmd_or_reply* = *cmd_op_code* (*cmd_name*) rcvd for fh = *file_handle* corrID = *rfs_correlation_id*

Long Syntax: TSRS.012 RFS *cmd_or_reply* = *cmd_op_code* (*cmd_name*) received for file handle = *file_handle* corrID = *rfs_correlation_id*

Description: A RFS/400 request was received. The file handle can be used to determine which file the request is associated with and the correlation ID can be used to determine which client issued the request.

TSRS.013

Level: UI-ERROR

Short Syntax: TSRS.013 Rcvd RFS rply(*reply_op_code*), exp(*cmd_op_code*)

Long Syntax: TSRS.013 Received RFS reply(*reply_op_code*), expected(*cmd_op_code*)

Description: An unexpected reply was received. The correlation ID is used to match replies with requests and in this case the correlation ID in the reply indicates that it is a reply for cmd_op_code and the reply is actually repl_op_code

---

TSRS.014

Level: C-TRACE

Short Syntax: TSRS.014 Lookup *cmd_or_reply* (*file_name*) rc(*return_code*) (*client_ip_addr*)

Long Syntax: TSRS.014 Lookup *cmd_or_reply* (*file_name*) rc(*return_code*) (*client_ip_addr*)

Description: A file lookup request or reply was received. The lookup request will contain the file name for which information is being requested. The lookup reply contains the file information and an initial file handle. A return code of BD1 indicates that the file does not exist on the master file server.

TSRS.015

Level: UI-ERROR

Short Syntax: TSRS.015 8001 rcvd, rc(*return_code*), sh(*socket_handle*) cmd(*rfs_cmd*), file = *file_name*

Long Syntax: TSRS.015 8001(Return Code) received, rc(*return_code*), socket handle(*socket_handle*) cmd(*rfs_cmd*), file name = *file_name*

Description: The master file server returned an error to the client.

TSRS.016

Level: UI-ERROR

Short Syntax: TSRS.016 sckt call *socket_function* failed. hndl = *socket_handle*, errno = *errno*

Long Syntax: TSRS.016 socket call *socket_function*

failed. handle = *socket_handle*, errno = *errno*

Description: A socket call failed. The socket call returned the specified error number (errno) which contains information as to the reason for the failure.

TSRS.017

Level: C-TRACE

Short Syntax: TSRS.017 sck call *socket_function_name* successful. hndl = *socket_handle*, rc = *return_code*

Long Syntax: TSRS.017 socket call *socket_function_name* successful. handle = *socket_handle*, rc = *return_code*

Description: A socket call completed successfully.

TSRS.018

Level: P-TRACE

Short Syntax: TSRS.018 sckt *send_receive* (*from*) sh(*socket_handle*) rc(*return_code*) len(*length*) fh(*file_handle*) corrID(*rfs_correlation_id*)

Long Syntax: TSRS.018 socket *send_receive* (*from*) socket handle(*socket_handle*) rc(*return_code*) len(*length*) fh(*file_handle*) corrID(*rfs_correlation_id*)

Description: A frame was sent/received successfully.

TSRS.019

Level: P-TRACE

Short Syntax: TSRS.019 sckt snd() *cmd_name* rply. rc = *return_code*

Long Syntax: TSRS.019 socket send() *cmd_name* reply generated by TSF. rc = *return_code*

Description: The thin server has generated a reply for the client's request. The request was not passed through to the master file server.

TSRS.020

Level: P-TRACE

Short Syntax: TSRS.020 sckt lost. sckt(*socket_handle*) rc(*return_code*) port(*port_number*) listening(*listening_yes_or_no*)

Long Syntax: TSRS.020 socket lost. socket(*socket_handle*) rc(*return_code*) port(*port_number*) listening(*listening_yes_or_no*)

Description: The socket connection was lost. This is a normal condition, connections are initiated and terminated on a continual basis.

TSRS.021

Level: C-INFO

Short Syntax: TSRS.021 Open (*file_name*) to (*disk_yes_or_no*) by (*client_ip_addr*)

Long Syntax: TSRS.021 Open file(*file_name*) to (*disk_yes_or_no*) by (*client_ip_addr*)

Description: A file was opened by the client specified. The second parameter indicates whether the thin server will attempt to serve this file or whether the master file server will serve this file.

TSRS.022

Level: C-TRACE

Short Syntax: TSRS.022 read offset(base= *base_offset*, rel= *relative_offset*), len(*length*)

Long Syntax: TSRS.022 Processing read offset(base= *base_offset*, rel= *relative_offset*), length(*length*)

Description: The client has made a read request.

TSRS.023

Level: C-INFO

Short Syntax: TSRS.023 Conn est to *ip_addr*

Long Syntax: TSRS.023 Connection established to client at ip addr *ip_addr*

Description: A connection has been made by a client to the Thin Server RFS daemon.

TSRS.024

Level: C-INFO

Short Syntax: TSRS.024 Conn lost to *ip_addr*

Long Syntax: TSRS.024 Connection lost to client at ip address *ip_addr*

Description: All of the TCP connections to the specified client have been lost.

Cause: Network error.

Action: Verify that the path between the thin server and the client is available.

Cause: Client was powered off

Action: none.

TSRS.025

Level: C-TRACE

Short Syntax: TSRS.025 Call ReadFileRequest() buf strt(*buffer_addr*) end(*buffer_addr*)

Long Syntax: TSRS.025 Calling ReadFileRequest() buf start(*buffer_addr*) end(*buffer_addr*)

Description: Calling the hard file to read a file from the local disk.

TSRS.026

Level: C-INFO

Short Syntax: TSRS.026 Close (*file_name*) by (*client_ip_addr*), *num_bytes* bytes served

Long Syntax: TSRS.026 Close file (*file_name*) by (*client_ip_addr*), *num_bytes* bytes served

Description: The file specificed is being closed. The second parameter indicates the ip address of the client to which this file was served and the third parameter indicates the number of bytes which were served.

TSRS.027

Level: P-TRACE

Short Syntax: TSRS.027 Hndl based Lookup Req. hndl(*file_handle*)

Long Syntax: TSRS.027 Handle based Lookup Request. handle(*file_handle*)

Description: The client issued a handle based lookup request (most lookup resuests are name based) the the specified handle.

TSRS.028

Level: P-TRACE

Short Syntax: TSRS.028 snd() to(*target_device*) *rfs_command* for *file_name* (hndl = *file_handle*)

Long Syntax: TSRS.028 send() to(*target_device*) *rfs_command* for *file_name* (handle = *file_handle*)

Description: The specified RFS/400 command was generated by the thin server.

TSRS.029

Level: C-TRACE

Short Syntax: TSRS.029 *ctl_blk_type* ctl blk destroyed. name = *name_assoc_with_ctl_blk*

Long Syntax: TSRS.029 *ctl_blk_type* control block destroyed. name = *name_assoc_with_ctl_blk*

Description: The specified control block is being destroyed.

TSRS.030

Level: UI-ERROR

Short Syntax: TSRS.030 Congestion snding, waiting *num_seconds* seconds to retry, errno = *errno*

Long Syntax: TSRS.030 Congestion sending, waiting *num_seconds* seconds to retry, errno = *errno*

Description: The RFS Server is experiencing congestion when attempting to send data. It pauses to allow the congestion to subside.

TSRS.031

Level: UI-ERROR

Short Syntax: TSRS.031 Req Ctl blk not fnd. corrID = *corr_id*

Long Syntax: TSRS.031 Request Control block not found. corrID received = *corr_id*

Description: When a reply is received from the master file server, the TSF needs to access information stored when the request to which the reply corresponds was processed. That information could not be found. The corr_id is the value of the correlation ID received in the data frame.

TSRS.032

Level: UI-ERROR

Short Syntax: TSRS.032 File ctl blk not found using *search_type* (*search_parameter*)

Long Syntax: TSRS.032 File control block not found using *search_type* (*search_parameter*)

Description: File control block could not be found. search_type indicates the method which was used to try to find the control block, which dependent on the type of frame received. search_parameter is the parameter input into the search algorithm.

TSRS.033

Level: UI-ERROR

Description: The RFS daemon has reached the limit of the number of client connections it can accept. If many clients have been re-booted on recently, then this condition should subside once the original connections time out. If this condition persists, then verify that the number of clients does not exceed the maximum allowed.

Chapter 117. Thin Server TFTPD and TFTPD Relay Server (TSTD)

This chapter describes Thin Server TFTPD and TFTPD Relay Server (TSTD) messages. For information on message content and how to use the message, refer to the Introduction.

TSTD.001

Level: UI-ERROR

Short Syntax: TSTD.001 TFTP Relay *message* socket *value* msg *value*

Long Syntax: TSTD.001 TFTP Relay *message* socket *value* msg *value*

Description: Information messages for TFTP Relay to Master Host.

TSTD.002

Level: C-INFO

Short Syntax: TSTD.002 Request from client *IP-address* about *file* peer number= *peer-number*

Long Syntax: TSTD.002 Request from client *IP-address* about *file* peer number= *peer-number*

Description: Request from client with IP address about a file has been received. The peer number has been assigned to this request.

TSTD.003

Level: UI-ERROR

Short Syntax: TSTD.003 System resource error: *error-code*

Long Syntax: TSTD.003 System resource error: *error-code*

Description: A system resource error has occurred. Record the error code. Check the configuration parameters.

TSTD.004

Level: C-TRACE

Short Syntax: TSTD.004 Port number assigned to this request *port-number* peer number *peer*

Long Syntax: TSTD.004 Port number assigned to this request *port-number* peer number *peer*

Description: Internal port number and peer number assigned to this TFTP request.

TSTD.005

Level: UI-ERROR

Short Syntax: TSTD.005 Error sending file to client error code= *error-code*

Long Syntax: TSTD.005 Error sending file to client error code= *error-code*

Description: Error sending file to client check error for appropriate action.

TSTD.006

Level: U-TRACE

Short Syntax: TSTD.006 Max. number of concurrent requests *count* current number *value*

Long Syntax: TSTD.006 Max. number of concurrent requests *count* current number *value*

Description: Maximum number of concurrent TFTP requests processed from the NetworkStations and the current number of requests.

TSTD.007

Level: C-TRACE

Short Syntax: TSTD.007 Number of current TFTP Relay requests = *number*

Long Syntax: TSTD.007 Number of current TFTP Relay requests = *number*

Description: The number of current TFTP Relay requests.

TSTD.008

Level: C-INFO

Short Syntax: TSTD.008 Request relayed to Master Server *Master-Server-IP-address*

Long Syntax: TSTD.008 Request relayed to Master Server *Master-Server-IP-address*

Description: A TFTP file request has been received from a client and is being relayed to the Master Server.

TSTD.009

Level: U-TRACE

Short Syntax: TSTD.009 Packet Request options error *options-error*

Long Syntax: TSTD.009 Packet Request options error *options-error*

Description: TFTP packet request options error from client.

TSTD.010

Level: C-INFO

Short Syntax: TSTD.010 File *file* - *was(was* found in local cache

Long Syntax: TSTD.010 File *file* - *was(was* found in local cache

Description: The requested file was (was not) found in the local cache.

TSTD.011

Level: C-INFO

Short Syntax: TSTD.011 File *file* - *was(was* retrieved from Master server

Long Syntax: TSTD.011 File *file* - *was(was* retrieved from Master server

Description: The requested file was (was not) retrieved from the Master Server.

TSTD.012

Level: UI-ERROR

Short Syntax: TSTD.012 Recoverable network or system resource error *error-code*

Long Syntax: TSTD.012 Recoverable network or system resource error *error-code*

Description: A recoverable network or system resource error has occurred.

TSTD.013

Level: UI-ERROR

Short Syntax: TSTD.013 Error *error-number* sending OACK for block size to client *peer-number*

Long Syntax: TSTD.013 Error *error-number* sending OACK for block size to client *peer-number*

Description: Error sending OACK for block size negotiation to client.

TSTD.014

Level: UI-ERROR

Short Syntax: TSTD.014 Error *error-number* in ACK to OACK block size from client *peer-number*

Long Syntax: TSTD.014 Error *error-number* in ACK to OACK block size from client *peer-number*

Description: Error receiving ACK for block size negotiation to client.

TSTD.015

Level: UI-ERROR

Short Syntax: TSTD.015 Error packet received from client *peer-number*

Long Syntax: TSTD.015 Error packet received from client *peer-number*

Description: Error packet received from client during block size negotiation.

TSTD.016

Level: UI-ERROR

Short Syntax: TSTD.016 Error opening file with return code= *value*

Long Syntax: TSTD.016 Error opening file with return code= *value*

Description: Error opening file with return code.

TSTD.017

Level: UI-ERROR

Short Syntax: TSTD.017 Error reading file with return code= *value*

Long Syntax: TSTD.017 Error reading file with return code *value*

Description: Error reading file with return code.

TSTD.018

Level: UI-ERROR

Short Syntax: TSTD.018 Error sending file to client peer= *peer*, at block= *number*, error= *value*

Long Syntax: TSTD.018 Error sending file to client peer= *peer*, at block= *number*, error= *value*

Description: Error sending file to client at block number with error value.

TSTD.019

Level: UI-ERROR

Short Syntax: TSTD.019 Error receiving ACK from client peer= *peer*

Long Syntax: TSTD.019 Error receiving ACK from client peer= *peer*

Description: Error receiving ACK from client.

TSTD.020

Level: UI-ERROR

Short Syntax: TSTD.020 Error packet received from client peer= *peer*

Long Syntax: TSTD.020 Error packet received from client peer= *peer*

Description: Error packet received from client.

TSTD.021

Level: C-TRACE

Short Syntax: TSTD.021 Request for file complete peer= *peer-number*

Long Syntax: TSTD.021 Request for file complete peer= *peer-number*

Description: Request for file complete for client with peer number.

TSTD.022

Level: C-TRACE

Short Syntax: TSTD.022 Wrong block ack.ed resend frame *IP-Address*, block= *number*, last block= *number*

Long Syntax: TSTD.022 Wrong block ack.ed resend frame *IP-Address* block= *number*, last block= *number*

Description: Wrong block ACK.ed from client. Resending frame to client.

TSTD.023

Level: UI-ERROR

Short Syntax: TSTD.023 Timeout on response from client peer= *peer*

Long Syntax: TSTD.023 Timeout on response from client peer= *peer*

Description: Timeout occurred on response from client.

TSTD.024

Level: U-INFO

Short Syntax: TSTD.024 Maximum TFTP threads exceeded *thread-count*

Long Syntax: TSTD.024 Maximum TFTP threads exceeded *thread-count*

Description: The Maximum number concurrent TFTP requests has been exceeded. The TFTP request will be processed on the next available thread.

TSTD.025

Level: C-TRACE

Short Syntax: TSTD.025 Send frame to client, peer= *peer*, block#= *block*, block size= *block-size*

Long Syntax: TSTD.025 Send frame to client, peer= *peer*, block#= *block*, block size= *block-size*

Description: Sending frame to client. Block number and block size.

TSTD.026

Level: C-TRACE

Short Syntax: TSTD.026 Resend frame to client *IP-Address*, block#= *block*, last block rec.ed= *last-block-received*

Long Syntax: TSTD.026 Resend frame to client *IP-Address*, block#= *block*, last block rec.ed= *last-block-received*

Description: Resend frame to client for block number.

TSTD.027

Level: C-TRACE

Short Syntax: TSTD.027 ACK received from peer= *peer*, block rec.ed= *block-number-received*, block expected= *block-number-expected*

Long Syntax: TSTD.027 ACK received from peer= *peer*, block rec.ed= *block-number-received*, block expected= *block-number-expected*

Description: ACK received from client for block.

TSTD.028

Level: UI-ERROR

Short Syntax: TSTD.028 NAK sent to client, peer= *peer*, error= *error-code*

Long Syntax: TSTD.028 NAK sent to client, peer= *peer*, error= *error-code*

Description: NAK sent to client.

TSTD.029

Level: UI-ERROR

Short Syntax: TSTD.029 Open request failed on Host Server *IP-address*, *error-code*

Long Syntax: TSTD.029 Open request failed on Host Server *IP-address*, *error-code*

Description: Open request failed on Host Server check configuration parameters and network for Host Server access.

TSTD.030

Level: UI-ERROR

Short Syntax: TSTD.030 Read request failed on Host Server *IP-address*, *error-code*

Long Syntax: TSTD.030 Read request failed on Host Server *IP-address*, *error-code*

Description: Read request failed on Host Server check configuration parameters and Host Server file access.

TSTD.031

Level: UI-ERROR

Short Syntax: TSTD.031 TFTP thread(0x *threadid*) stack util(*stack_util* percent) file (*file_name*) file avail(*IsFileAvailable_rc*)

Long Syntax: TSTD.031 TFTP thread(0x *threadid*) stack util(*stack_util* percent) file (*file_name*) file avail(*IsFileAvailable_rc*)

Description: The stack utilization for thread threadid was stack_util which is dangerously high. The file_name and IsFileAvailable_rc may indicate a possible cause.

Chapter 118. Token Ring Network Interface (TKR)

This chapter describes Token Ring Network Interface (TKR) messages. For information on message content and how to use the message, refer to the Introduction.

TKR.001

Level: U-INFO

Short Syntax: TKR.001 unexp *type* frm *LLC_control* fm *source_MAC* ssap *source_SAP* dsap *dest_SAP* nt *network ID*

Long Syntax: TKR.001 Unexpected *type* frame *LLC_control* from *source_MAC*, ssap *source_SAP*, dsap *dest_SAP*, net *network ID*

Description: This message is generated when an unexpected 802.2 LLC frame type is received. Type may be I (information transfer) or S (supervisory). The frame was addressed to the router.

Cause: Host attempting to make 802.2 type 2 connection to router.

TKR.002

Level: P-TRACE

Short Syntax: TKR.002 unexp *type* brd frm *LLC_control* fm *source_MAC* ssap *source_SAP* dsap *dest_SAP* nt *network ID*

Long Syntax: TKR.002 Unexpected *type* broadcast frame *LLC_control* from *source_MAC*, ssap *source_SAP*, dsap *dest_SAP*, net *network ID*

Description: This message is generated when an unexpected 802.2 LLC frame type is received. Type may be I (information transfer) or S (supervisory). The frame was a broadcast.

Cause: Host attempting to make 802.2 type 2 connection to router.

TKR.003

Level: U-INFO

Short Syntax: TKR.003 unkn SNAP mfr cd *number* fm *source_MAC* nt *network ID*

Long Syntax: TKR.003 Unknown SNAP manufacturer code *number* from *source_MAC* net *network ID*

Description: This message is generated when a frame with an unknown organization code (not 000000) in the SNAP header is received. The frame was addressed to the router.

Cause: Host sending packets for unknown proprietary protocol using SNAP.

TKR.004

Level: P-TRACE

Short Syntax: TKR.004 unkn SNAP mfr code *number* fm *source_MAC* nt *network ID*

Long Syntax: TKR.004 Unknown SNAP manufacturer code *number* from *source_MAC* net *network ID*

Description: This message is generated when a frame with an unknown organization code (not 000000) in the SNAP header is received. The frame was a broadcast.

Cause: Host sending packets for unknown proprietary protocol using SNAP.

TKR.005

Level: U-INFO

Short Syntax: TKR.005 unkn SNAP type *type_code* fm *source_MAC* nt *network ID*

Long Syntax: TKR.005 Unknown SNAP type *type_code* from *source_MAC* net *network ID*

Description: This message is generated when a frame with an unknown SNAP type (within organization code 000000) is received. The frame was addressed to the router.

Cause: Host sending packets for unknown Ethernet type using SNAP.

TKR.006

Level: P-TRACE

Short Syntax: TKR.006 unkn SNAP type *type_code* fm *source_MAC* nt *network ID*

Long Syntax: TKR.006 Unknown SNAP type *type_code* from *source_MAC* net *network ID*

Description: This message is generated when a frame with an unknown SNAP type (within organization code 000000) is received. The frame was a broadcast.

Cause: Host sending packets for unknown Ethernet type using SNAP.

TKR.007

Level: U-INFO

Short Syntax: TKR.007 unkn SAP *sap_number* fm *source_MAC* nt *network ID*

Long Syntax: TKR.007 Unknown SAP *sap_number* from *source_MAC* net *network ID*

Description: This message is generated when a frame with an unknown destination SAP is received. The message was addressed to the router.

Cause: Host sending packets for unknown protocol identifier (SAP).

TKR.008

Level: U-INFO

Short Syntax: TKR.008 unkn SAP *sap_number* fm *source_MAC* nt *network ID*

Long Syntax: TKR.008 Unknown SAP *sap_number* from *source_MAC* net *network ID*

Description: This message is generated when a frame with an unknown destination SAP is received. The message was a broadcast.

Cause: Host sending packets for unknown protocol identifier (SAP).

TKR.009

Level: U-INFO

Short Syntax: TKR.009 unexp U frm *LLC_control* fm *source_MAC* ssap *source_SAP* dsap *dest_SAP* nt *network ID*

Long Syntax: TKR.009 Unexpected U frame *LLC_control* from *source_MAC*, ssap *source_SAP*, dsap *dest_SAP*, net *network ID*

Description: This message is generated when an unexpected 802.2 LLC U (unnumbered) frame type is received. (Only UI, XID, and TEST are supported.) The frame was addressed to the router.

TKR.010

Level: P-TRACE

Short Syntax: TKR.010 unexp U frm *LLC_control* fm *source_MAC* ssap *source_SAP* dsap *dest_SAP* nt *network ID*

Long Syntax: TKR.010 Unexpected U frame *LLC_control* from *source_MAC*, ssap *source_SAP*, dsap *dest_SAP*, net *network ID*

Description: This message is generated when an unexpected 802.2 LLC U (unnumbered) frame type is received. (Only UI, XID, and TEST are supported.) The frame was a broadcast.

TKR.014

Level: UI-ERROR

Short Syntax: TKR.014 *selftest_phase* fld *error_condition* nt *network*

Long Syntax: TKR.014 *selftest_phase* failed: *error_condition*, network *network*

Description: The self-test for the 802.5 Token-Ring card has reported an error during self-test. The phases are ″Initial test″, ″Board reset″, ″Configuration″, ″Open″, ″Open: Lobe media test″, ″Open: Physical insertion″, ″Open: Address verification″, ″Open: Roll call poll″, ″Open: Request parameters″, ″Packet output″, and ″Packet receive″. See message TKR-45 for IBM Token-Ring self-test failures.

Cause: In the ″Initial test″ phase, the error is ″Buffer unavail″. This indicates that there is a severe packet buffer shortage in the router.

Action: Increase memory size, or decrease size of routing tables.

Cause: In the ″Board reset″ phase, the error can be one of: ″Initial test error″, ″Adaptor ROM CRC error″, ″Adaptor RAM error″, ″Instruction Test error″, ″Context/Interrupt Test error″, ″Protocol Handler Hardware Err″, or ″System Interface Register Err″. Any of these indicate internal problems within the adapter chipset.

Action: Probable hardware failure of interface. Replace.

Cause: In the ″Configuration″ phase, the error can be one of: ″Invalid init block″, ″Invalid options″, ″Invalid receive burst″, ″Invalid transmit burst″, ″Invalid DMA abort threshhold″, ″Invalid SCB″, ″Invalid SSB″, ″DIO Parity″, ″DMA timeout″, ″DMA parity″, ″DMA bus error″, ″DMA data error″, or ″Adaptor check″. These can indicate a hardware problem within the chipset, or a software problem.

Action: Probable hardware failure of interface. Replace.

Cause: In the ″Open″ phase, the error can be one of: ″Node address error″, ″List size error″, ″Buffer size error″, ″Expansion RAM error″, ″Transmit buffer count″, or ″Inavlid open option″. These can indicate a hardware problem within the chipset, or a software problem.

Action: Probable hardware failure of interface. Replace.

Cause: In the ″Open: Lobe media test″, ″Open: Physical insertion″, ″Open: Address verification″, ″Open: Participation in ring poll″, and ″Open: Request initialization″ phases, the error can be one of: ″Function failure″, ″Signal loss″, ″Timeout″, ″Ring failure″, ″Ring beaconing″, ″Duplicate node Address″, ″Request initialization″, ″Remove received″, or ″IMPL force received″. These are indications of failures in the process of the MAC algorithms for joining the ring. The

problem is probably in the ring or the cabling, not the interface.

Action: Investigate network problems in 802.5 ring that the interface is attempting to connect to.

Cause: In the ″Packet output″ phase, the error is ″Unknown″. The self-test packet that was sent by the node to itself did not have the address recognized bit set upon the completion of transmission.

Action: Investigate network problems, possible hardware problem.

Cause: In the ″Packet input″ phase, the error is ″Unkown″. The self-test packet that was sent by the node to itself was not received within half a second.

Action: Investigate network problems, possible hardware problem.

TKR.015

Level: UI-ERROR

Short Syntax: TKR.015 dwn sts cls nt *network*

Long Syntax: TKR.015 Down, ring status close indication, network *network*

Description: The interface has automatically removed itself form the ring due to some serious error condition. This may be one of ″Lobe wire fault″, ″Auto-removal error″, or ″Remove received″. The interface will attempt to join the ring again, and may come up again.

Cause: There is a hardware problem with the ring or the interface. The exact cause is not logged, but these errors are counted, and the counters in the +interface command should indicate what the problem is.

Action: Look at the interface counters. ″Lobe wire fault″ indicates a problem with the network. ″Auto-removal error″ indicates internal problems with the interface. ″Remove received″ indicates that a network management station has instructed this station to leave the ring.

TKR.016

Level: UI-ERROR

Short Syntax: TKR.016 dwn adap chk *adapter_check_code* nt *network*

Long Syntax: TKR.016 Down, adapter check *adapter_check_code*, network *network*

Description: The interface has been brought down because of an adaptor status check. The interface will not be self-tested, and will not come back up automatically. The adapter_check_code indicates which error occurred.

Cause: The adapter has detected a severe unrecoverable internal failure.

Action: If the problem persists, have the interface replaced.

TKR.017

Level: UI-ERROR

Short Syntax: TKR.017 pkt sz *configured_size* too big for 4 Mbps, limting to *maximum_size*, nt *network*

Long Syntax: TKR.017 Packet size *configured_size* too big for 4 Megabit/Second, limiting to *maximum_size*, network *network*

Description: The user has set the packet size for the 802.5 network larger than is allowed for a 4 Megabit/second network. The 8144, 11407, and 17800 byte sizes are only legal on a 16 Megabit/second network.

TKR.019

Level: UE-ERROR

Short Syntax: TKR.019 runt pkt (*length*) frm *source_address*, net *network*

Long Syntax: TKR.019 runt packet (*length* bytes) from node *source_address*, network *network*

Description: A packet has been received which is too short to contain the MAC and LLC headers.

Cause: External error.

TKR.020

Level: UE-ERROR

Short Syntax: TKR.020 DN bd ln *actual_length claimed_length source_MAC_address* -> *destination_MAC_address* nt *network*

Long Syntax: TKR.020 DECnet packet received with a bad length actual *actual_length* claimed *claimed_length* from *source_MAC_address* to *destination_MAC_address* network *network*

Description: A DECnet packet was received with a length field that was larger than the actual length of the packet.

TKR.021

Level: P-TRACE

Short Syntax: TKR.021 LOOP rcv *source_MAC_address* -> *destination_MAC_address*, nt *network*

Long Syntax: TKR.021 Loopback Protocol frame received from *source_MAC_address* to *destination_MAC_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet was received.

TKR.022

Level: UE-ERROR

Short Syntax: TKR.022 LOOP odd skp *count*, *source_MAC_address* -> *destination_MAC_address*, nt *network*

Long Syntax: TKR.022 Loopback Protocol, odd skipCount *count* from *source_MAC_address* to *destination_MAC_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet had an odd skipCount in the packet. It will be discarded.

Cause: Programming error on remote node.

TKR.023

Level: UE-ERROR

Short Syntax: TKR.023 LOOP bd skp *count*, *source_MAC_address* -> *destination_MAC_address*, nt *network*

Long Syntax: TKR.023 Loopback Protocol, bad skipCount *count* from *source_MAC_address* to *destination_MAC_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet had a skipCount in the packet that points to beyond the end of the packet. It will be discarded.

Cause: Programming error on remote node.

TKR.024

Level: P-TRACE

Short Syntax: TKR.024 LOOP func *function* not forw, *source_MAC_address* -> *destination_MAC_address*, nt *network*

Long Syntax: TKR.024 Loopback Protocol, function *function* not Forward Data from *source_MAC_address* to *destination_MAC_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet did not have a function code of forward (2). It will be discarded.

Cause: Function code was reply (1), because we were the ultimate destination of this packet.

Action: None.

TKR.025

Level: UE-ERROR

Short Syntax: TKR.025 LOOP mc fwd dst *forward_MAC_address*, *source_MAC_address* -> *destination_MAC_address*, nt *network*

Long Syntax: TKR.025 Loopback Protocol, multicast forward address *forward_MAC_address* from *source_MAC_address* to *destination_MAC_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet has a forward address that is a multicast. It will be discarded.

Cause: Programming error in remote node.

TKR.026

Level: P-TRACE

Short Syntax: TKR.026 LOOP fwd *source_MAC_address* -> *forward_MAC_address*, nt *network*

Long Syntax: TKR.026 Loopback Protocol, forwarding from *source_MAC_address* to *forward_MAC_address*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet is being forwarded to the specified next hop.

TKR.027

Level: UI-ERROR

Short Syntax: TKR.027 LOOP fwd to *forward_Ethernet_address* dsc, rsn *code*, nt *network*

Long Syntax: TKR.027 Loopback protocol, forward to *forward_Ethernet_address* discarded, for reason *code*, network *network*

Description: An Ethernet Loopback Protocol (Configuration Testing Protocol) packet could not be forwarded to the specified address, for the reason specified by code.

TKR.028

Level: UI-ERROR

Short Syntax: TKR.028 rif table corruption for nt *network*

Long Syntax: TKR.028 rif related functions failed because of rif table corruption on network *network*

Description: The rif table is corrupted.

TKR.029

Level: P_TRACE

Short Syntax: TKR.029 rif entry is being removed entry *hardware_address protocol_type* nt *network*

Long Syntax: TKR.029 rif aging function is removing entry *hardware_address protocol_type* network *network*

Description: The rif entry aging function is removing an entry from the rif table.

TKR.030

Level: UI_ERROR

Short Syntax: TKR.030 MAC frm typ *mac_frametype* unex from *hardware_address* nt *network*

Long Syntax: TKR.030 MAC frame type *mac_frametype* unexpected from *hardware_address* network *network*

Description: The handler received a frame with an unexpected frame type.

TKR.031

Level: P_TRACE

Short Syntax: TKR.031 Main rcd on nt *network*

Long Syntax: TKR.031 Maintenance packet received on net *network*

Description: The handler received a maintenance packet.

TKR.032

Level: P_TRACE

Short Syntax: TKR.032 test frm *mac_address*, src sap *source_sap*, nt *network*

Long Syntax: TKR.032 test packet from *mac_address*, source sap *source_sap*, net *network*

Description: The handler received a test message.

TKR.033

Level: P_TRACE

Short Syntax: TKR.033 xid frm *mac_address*, sap *source_sap*, nt *network*

Long Syntax: TKR.033 xid packet received from *mac_address*, source sap *source_sap*, net *network*

Description: The handler received an xid message.

TKR.035

Level: U-TRACE

Short Syntax: TKR.035 new RIF (*RIF*) for *MAC_address* nt *network ID*

Long Syntax: TKR.035 new RIF (*RIF*) for *MAC_address* net *network ID*

Description: This message is generated when a new RIF is added to the 802.5 MAC address to RIF translation cache.

TKR.036

Level: ALWAYS

Short Syntax: TKR.036 can't set 2nd grp addr *MAC_address*

Long Syntax: TKR.036 can't set 2nd group address *MAC_address*

Description: The Token-Ring hardware can only support one group address. A second address is being attempted to be installed.

TKR.037

Level: ALWAYS

Short Syntax: TKR.037 Net *network ID*, Unkn SRT Cmd Completion code - *SRT_Completion*. Being restarted

Long Syntax: TKR.037 Network *network ID*, Has Received an Unknown SRT Command Completion code - *SRT_Completion*. Interface being restarted

Description: The Token-Ring board has returned an unexpected SRT Completion Code. This will cause the interface to enter self-test.

TKR.038

Level: ALWAYS

Short Syntax: TKR.038 Net *network ID*, Cmnd to TKR failed - invld param(s). Being restarted

Long Syntax: TKR.038 Network *network ID*, Command to Token Ring Adapter failed - invalid parameter(s). Interface being restarted

Description: The Token-Ring board has returned an a illegal parameter status code indicating that one or more of the parameters passed to it were invalid. This will cause the interface to re-initialize.

TKR.039

Level: ALWAYS

Short Syntax: TKR.039 Net *network ID*, Unkn TKR Cmd Completion code - *Completion_Code*. Being restarted

Long Syntax: TKR.039 Network *network ID*, Unknown Command Completion code - *Completion_Code*. Interface being restarted

Description: The Token-Ring board has returned an unexpected Completion Code. This will cause the interface to re-initialize.

TKR.040

Level: ALWAYS

Short Syntax: TKR.040 Net *network ID*, Invld Command *Command* rcvd in tm_ioctl. Being restarted

Long Syntax: TKR.040 Network *network ID*,Invalid Command *Command* received by tm_ioctl from handler. Interface being restarted

Description: The tm_ioctl routine has received an invalid command from the device handler. This will cause the interface to re-initialize.

TKR.041

Level: ALWAYS

Short Syntax: TKR.041 Net *network ID*, Invld Interrupt rcvd *Interrupt* from TKR adapter. Being restarted

Long Syntax: TKR.041 Network *network ID*, Invalid Interrupt *Interrupt* received from the TKR adapter. Interface being restarted

Description: The interrupt service routine has received an invalid interrupt from the adapter card. This will cause the interface to re-initialize.

TKR.043

Level: UE-ERROR

Short Syntax: TKR.043 drop IPX pkt w/ *encap_seen* encaps - using *encap_used* encaps on int *intnum*

Long Syntax: TKR.043 dropped IPX pkt with encaps *encap_seen* using *encap_used* on interface *intnum*

Description: This message is generated when an IPX packet is recieved with an encapsulation other than that which has been selected for this interface

Cause: Normal for networks using multiple encapsulations on a single wire.

Action: None needed.

TKR.044

Level: UE-ERROR

Short Syntax: TKR.044 odd RIF len frm *MAC_address*; pkt drpd nt *network ID*

Long Syntax: TKR.044 odd RIF length from *MAC_address*; packet dropped on net *network ID*

Description: The length byte in the RIF header was odd, which is illegal. The packet was dropped.

TKR.045

Level: UI-ERROR

Short Syntax: TKR.045 *selftest_phase* fld *error_condition* nt *network*

Long Syntax: TKR.045 *selftest_phase* failed: *error_condition*, network *network*

Description: The self-test for the IBM 802.5 Token-Ring has reported an error during self-test. This message can often serve as a useful quick primitive diagnostic tool for the Token-Ring hardware. The phases are ″reset″, ″load loader (part 1)″, ″load loader (part 2)″, ″download microcode″, ″check downloaded microcode″, ″Configuration″, ″Read interesting pointers″, ″open: lobe media test″, ″open: physical insertion″, ″open: address verification″, ″open: participation in ring poll″, ″open: request initialization″, ″Set bridge params″, ″Set STE wanted″, ″Packet output″, ″Packet receive″, ″SRT Config″, ″Set func/group address″, ″Unknown Test″.

Cause: open: lobe media test: function failure.

Action: This is a basic cable problem. Check the cable. Check that router configuration has the correct media cable setting, that is, UTP or STP.

Cause: open: physical insertion fld ring beaconing. The Token-Ring is beaconing. This is usually due to one station having a misconfigured speed.

Action: Check that router configuration has the correct speed setting, that is, 4 Mbps or 16 Mbps. Check that all the stations in your ring are set to the same speed. Check for physical breaks in the Token-Ring.

Cause: open: address verification fld duplicate node address. The MAC address for this interface is a duplicate on the ring.

Action: Check that router configuration has the correct MAC address for this interface. Verify the other stations on your ring for a duplicate address.

Cause: Any of the ″reset″, ″load loader (part 1)″, ″load loader (part 2)″, ″download microcode″, ″check downloaded microcode″ phases.

Action: Probable hardware failure of interface. Replace.

Cause: In the ″Configuration″ phase, the error can be one of: ″initial test error″, ″microcode crc error″, ″adapter ram error″, ″instruction test error″, ″context/interrupt test error″, ″protocol handler hardware err″, ″system interface register err″, ″invalid parameter length″, ″invalid options″, ″invalid receive burst″, ″invalid transmit burst″, ″invalid dma abort threshhold″, ″invalid dma test address″, ″dio parity″, ″dma timeout″, ″dma parity″, ″dma bus error″, ″dma data error″, ″adapter check″.

Action: These are the failures from the diagnostics run

by the adapter. Probable hardware failure of interface. Replace if it persists.

Cause: In the ″Open″ phase, the error can be one of: ″Node address error″, ″List size error″, ″Buffer size error″, ″Expansion RAM error″, ″Transmit buffer count error″, or ″Inavlid open option″.

Action: Probable hardware failure of interface. Replace.

Cause: The ″open: lobe media test″, ″open: physical insertion″, ″open: address verification″, ″open: participation in ring poll″, ″open: request initialization″ phases. The open operation has failed.

Action: These are fixable a lot of the time. The usual failures have already been described above. Check cable configuration and speed again. Investigate network or cabling problems, possible hardware problem.

Cause: Phases ″Set bridge params″, ″Set STE wanted″, ″SRT Config″, ″Set func/group address″ are phases related to setting the token-ring for bridging, group address, functional addresses, etc.

Action: This is more likely to be a software problem since the Token-Ring is already up and running successfully.

Cause: Packet output fld unknown. The Token-Ring driver could not send a test packet. This is more likely to be a software problem, such as the buffers within the router are exhausted.

Action: Restart router if it persists.

Cause: Packet receive fld unknown. The Token-Ring driver was unable to send a test packet around the ring and receive it.

Action: Check for an unusally large amount of traffic on the ring.

TKR.046

Level: C-INFO

Short Syntax: TKR.046 FasTR frm drpd from *SRC_address* to *Dest_address*, RIF *RIF*, nt *network*

Long Syntax: TKR.046 Fast Token Ring Frame dropped from *SRC_address* to *Dest_address*, RIF *RIF*, net *network*

Description: A Fast Token Ring frame with a RIF was dropped.

TKR.047

Level: C-INFO

Short Syntax: TKR.047 FasTR frm drpd from *SRC_address* to *Dest_address*, nt *network*

Long Syntax: TKR.047 Fast Token Ring frame dropped from *SRC_address* to *Dest_address*, net *network*

Description: A Fast Token Ring frame without a RIF was dropped.

TKR.048

Level: C-INFO

Short Syntax: TKR.048 FasTR tst frm looped from *SRC_address* to *Dest_address*, nt *network* UP

Long Syntax: TKR.048 Fast Token Ring frame looped from *SRC_address* to *Dest_address*, net *network*

Description: A Fast Token Ring Test frame was looped back while the net was up.

TKR.049

Level: C-INFO

Short Syntax: TKR.049 FasTR tst frm looped from *SRC_address* to *Dest_address*, nt *network* not UP

Long Syntax: TKR.049 Fast Token Ring frame looped from *SRC_address* to *Dest_address*, net *network*

Description: A Fast Token Ring Test frame was looped back while the net was not up.

TKR.050

Level: C-INFO

Short Syntax: TKR.050 *flash_condition* flash state: *flash_state*, nt *network*

Long Syntax: TKR.050 *flash_condition*, flash state = *flash_state* network *network*

Description: Information notification for the Token Ring flash update process. The reasons are ″Microcode update terminated after 3 attempts″, ″Non-AMD flash update is not supported″, ″Good status from flash update″, ″Error status from flash update″, ″Flash update is in progress″.

Panic tkrMacTooManyReg

Short Syntax: tkr_regMacAddrUpCall: too many registered

Description: Internal problem.

Cause: Software bug.

Action: Inform customer service.

Panic tkrMacStsTooManyReg

Short Syntax: tkr_regStatusUpCall: too many registered

Description: Internal problem.

Cause: Software bug.

Action: Inform customer service.

Panic tkrMacXmitTooManyReg

Short Syntax: tkr_regXmitpCall: too many registered

Description: Internal problem.

Cause: Software bug.

Action: Inform customer service.

Chapter 119. Transmission Control Protocol (TCP)

This chapter describes Transmission Control Protocol (TCP) messages. For information on message content and how to use the message, refer to the Introduction.

TCP.001

Level: UI-ERROR

Short Syntax: TCP.001 pkt cksum fld pkt = *tcp_checksum* calc = *tcp_checksum*

Long Syntax: TCP.001 packet checksum failed received packet checksum is *tcp_checksum* and calculated checksum is *tcp_checksum*

Description: Checksum failed because received packet checksum is not equal to the calculated checksum

TCP.002

Level: UI-ERROR

Short Syntax: TCP.002 rcvd pkt *source_ip_address* -> *destination_ip_address* dst prt *tcp_port* no cnn

Long Syntax: TCP.002 received packet *source_ip_address* -> *destination_ip_address* with destination port *tcp_port* has no tcp connection

Description: TCP has received a packet with an invalid tcp port number.

TCP.003

Level: C-INFO

Short Syntax: TCP.003 Act opn sccfl dst prt *tcp_port*

Long Syntax: TCP.003 TCP Active open successful for port number *tcp_port*

Description: Active open was successful and we are notifying application of the open.

TCP.004

Level: UI-ERROR

Short Syntax: TCP.004 rcvd invld SYN in wndw *source_ip_address* -> *destination_ip_address* dst prt *tcp_port* kill cnn

Long Syntax: TCP.004 received invalid SYN packet *source_ip_address* -> *destination_ip_address* with destination port *tcp_port*, kill connection

Description: TCP has received an illegal SYN packet, so kill the connection.

TCP.007

Level: UI-ERROR

Short Syntax: TCP.007 drp seg *source_ip_address* -> *destination_ip_address* dst prt *tcp_port* rsn *reject_code* snd ACK

Long Syntax: TCP.007 dropped segment *source_ip_address* -> *destination_ip_address* with destination port *tcp_port*, reason *reject_code*, send a valid ACK in response

Description: TCP has rejected a segment. Reject codes are as follows: Reject codes: 1 - Seg len = 0, Rcv win > 0, seqnum < tcb_ack 2 - Seg len = 0, Rcv win = 0, seqnum != tcb_ack 3 - Seg len > 0, Rcv win > 0, winend < tcb_ack 4 - Seg len > 0, Rcv win = 0. 5 - Seg len = 0, Rcv win > 0, seqnum >= winend 6 - Seg len > 0, Rcv win > 0, seqnum >= winend Note: we only ACK if the segment received was a non RST segment.

TCP.009

Level: C-INFO

Short Syntax: TCP.009 state LISTEN: rcvd RST dst prt *tcp_port* seq num *seq_num*

Long Syntax: TCP.009 while in LISTEN state, received RST with destination port *tcp_port*, sequence number *seq_num*; drop segment

Description: TCP has received a RST while in LISTEN state; just ignore packet.

TCP.010

Level: C-INFO

Short Syntax: TCP.010 state SYN_RCVD: RST|TIMEOUT rtrn to LISTEN port *tcp_port*

Long Syntax: TCP.010 while in SYN_RECEIVED states, received RST or TIMEOUT with local port *tcp_port*. Return to LISTEN state

Description: A TCP passive connection attempt has failed due to our receiving a RESET from the active partner, or due to TIMEOUT after returning SYN|ACK

TCP.011

Level: C-INFO

Short Syntax: TCP.011 rcvd RST dst prt *tcp_port* seq num *seq_num*, abort

Long Syntax: TCP.011 received RST with destination port *tcp_port*, sequence number *seq_num*; drop segment and abort connection

Description: TCP has received a RST; abort connection.

TCP.012

Level: UI-ERROR

Short Syntax: TCP.012 drop seg dst prt *tcp_port* seq num *seq_num* no ACK present

Long Syntax: TCP.012 drop segment with destination port *tcp_port*, sequence number *seq_num* because no ACK is present

Description: TCP has stopped processing the packet because there is no ACK present in the packet.

TCP.013

Level: UI-ERROR

Short Syntax: TCP.013 drop seg dst prt *tcp_port* seq num *seq_num* ack num *ack_num* rcv invld ACK

Long Syntax: TCP.013 drop segment with destination port *tcp_port*, sequence number *seq_num*, acknowledge number *ack_num*, received invalid ACK

Description: Stop processing the segment because it contains acknowledgement for data not yet sent.

TCP.014

Level: C-INFO

Short Syntax: TCP.014 state ESTAB: rcvd FIN dst prt *tcp_port* seq num *seq_num*

Long Syntax: TCP.014 while in ESTABLISHED state, received FIN with destination port *tcp_port*, sequence number *seq_num*

Description: TCP has received a FIN while in ESTABLISHED state; when all data has been received, send FIN|ACK.

TCP.015

Level: C-INFO

Short Syntax: TCP.015 rcvd PSH dst prt *tcp_port* seq num *seq_num*

Long Syntax: TCP.015 received a segment with the PSH bit set with destination port *tcp_port*, sequence number *seq_num*

Description: TCP has received a segment with PSH bit set.

TCP.016

Level: C-INFO

Short Syntax: TCP.016 state SYNRCVD: rcvd vld seg dst prt *tcp_port* seq num *seq_num*, enter ESTAB

Long Syntax: TCP.016 while in SYNRCVD state, received valid segment with destination port *tcp_port*, sequence number *seq_num*, so enter ESTABLISHED state

Description: TCP has received a valid segment while in SYNRCVD state; enter ESTABLISHED state and notify application of the open.

TCP.017

Level: UI-ERROR

Short Syntax: TCP.017 rcvd FIN while in LISTEN dst prt *tcp_port* seq num *seq_num*, snd RST

Long Syntax: TCP.017 received FIN segment while in the LISTEN state, destination port *tcp_port*, sequence number *seq_num*, snd RST

Description: TCP has received a FIN while in the LISTEN state, so we send RST to the other side.

TCP.024

Level: UI-ERROR

Short Syntax: TCP.024 prcss FIN in invld state

Long Syntax: TCP.024 process a received FIN; current state is not SYNRCVD|ESTAB, so do nothing

Description: TCP processing FIN while not in SYNRCVD|ESTAB state.

TCP.025

Level: C-INFO

Short Syntax: TCP.025 prcss FIN in ESTAB|SYNRCVD state frgn hst *ip_address* lcl hst *ip_address* dprt *dst_port* sprt *src_port*

Long Syntax: TCP.025 process a received FIN; current state is SYNRCVD|ESTAB, foreign host *ip_address* local host *ip_address* destination port *dst_port* source port *src_port*

Description: TCP processing FIN while in SYNRCVD|ESTAB state.

TCP.026

Level: C-INFO

Short Syntax: TCP.026 app rcv tmout

Long Syntax: TCP.026 application posted receive timeout has fired

Description: Application posts a read specifying a timeout value. If not all the requested data has been received within a timeout period, a timer fires, and whatever is in the receive buffer is given to the application.

TCP.027

Level: UI-ERROR

Short Syntax: TCP.027 frgn prt illgl close of wndw frgn hst *ip_address* lcl hst *ip_address* dprt *dst_port* sprt *src_port*

Long Syntax: TCP.027 foreign port closed the advertised window illegally foreign host *ip_address* local host *ip_address* destination port *dst_port* source port *src_port*

Description: The other side has been deaf and mute, and the the foreign window seems to have been closed illegally; send a RST.

TCP.028

Level: C-INFO

Short Syntax: TCP.028 state trnstn to SYNRCVD

Long Syntax: TCP.028 state of TCP connection transitioned to SYN-RECEIVED state

Description: State of the connection has transitioned to SYN-RECEIVED state as a result of either an active open or a passive open.

TCP.029

Level: C-INFO

Short Syntax: TCP.029 state trnstn to ESTAB

Long Syntax: TCP.029 state of TCP connection transitioned to ESTABLISHED state

Description: State of the connection has transitioned to ESTABLISHED state as a result of either an active open or a passive open.

TCP.030

Level: P-TRACE

Short Syntax: TCP.030 rcvd TCP pkt *source_ip_address* -> *destination_ip_address* dst prt *tcp_port*

Long Syntax: TCP.030 received packet *source_ip_address* -> *destination_ip_address* with destination port *tcp_port*

Description: TCP has received a packet.

TCP.031

Level: P-TRACE

Short Syntax: TCP.031 seq num *seq_num* to *seq_num* given to app.

Long Syntax: TCP.031 data with sequence number *seq_num* through to *seq_num* given to application

Description: Valid data in receive buffer has been handed to the application for further processing.

TCP.032

Level: C-INFO

Short Syntax: TCP.032 excssv num rtries

Long Syntax: TCP.032 excessive number of retries has occurred

Description: We have retransmitted a frame an excessive number of times. If the application has closed the connection already, just abort. Else, notify the application that there is a problem.

TCP.033

Level: P-TRACE

Short Syntax: TCP.033 snd ctrl seg seq num *seq_num* ack num *ack_num* wndw *window*

Long Syntax: TCP.033 send control segment with sequence number *seq_num* and acknowledge number *ack_num* window *window*

Description: Send a control segment to either ack a segment or send special control segments like FIN or RST.

TCP.034

Level: C-INFO

Short Syntax: TCP.034 rxmt seq num *seq_num* to *seq_num*

Long Syntax: TCP.034 retransmit data with sequence number *seq_num* through to *seq_num*

Description: We have failed to receive a valid ACK for transmitted data, so retransmit the data.

TCP.035

Level: P-TRACE

Short Syntax: TCP.035 xmt seq num *seq_num* to *seq_num*

Long Syntax: TCP.035 transmit data with sequence number *seq_num* through to *seq_num*

Description: Transmit data.

TCP.036

Level: UI-ERROR

Short Syntax: TCP.036 unspt optn(*kind*) rcvd in SYN seg

Long Syntax: TCP.036 unsupported option(*kind*) received in SYN segment

Description: An unsupported option is present in the options field of a SYN packet.

TCP.038

Level: UI-ERROR

Short Syntax: TCP.038 rjct seg dst prt *tcp_port* seq num *seq_num* bad ACK in SYNRCVD, snd RST

Long Syntax: TCP.038 reject segment with destination port *tcp_port* and sequence number *seq_num*, bad ACK in segment while in SYNRCVD state

Description: Reject the segment, and send a RST to the other side for receiving a segment with the incorrect acknowledgement while in the SYNRCVD state. Until a correct acknowledgement is received, we cannot progress into the ESTABLISHED state.

TCP.039

Level: UI-ERROR

Short Syntax: TCP.039 rcvd ACK seg with dst prt *tcp_port* seq num *seq_num* in LISTEN, snd RST

Long Syntax: TCP.039 received ACK segment with destination port *tcp_port*, sequence number *seq_num* while in the LISTEN state, send RST

Description: TCP has received an ACK while in the LISTEN state; this does not make any sense because we have not yet sent any data, so nothing should be ACKed. As a result, we send a RST.

TCP.040

Level: UI-ERROR

Short Syntax: TCP.040 TCP snd rst to hst *source_ip_address*

Long Syntax: TCP.040 TCP sending RESET to host *source_ip_address*

Description: TCP is sending a RESET segment to the other side.

TCP.041

Level: C-INFO

Short Syntax: TCP.041 TCP cnn clsd frgn hst *foreign_ip_address* lcl hst *local_ip_address*

Long Syntax: TCP.041 TCP connection closed, foreign host *foreign_ip_address*, local hst *local_ip_address*

Description: TCP connection is closed - notifying the application.

TCP.042

Level: C-INFO

Short Syntax: TCP.042 Frng TCB for frgn hst *foreign_ip_address* lcl hst *local_ip_address*

Long Syntax: TCP.042 Freeing TCB block for connection between *foreign_ip_address* and *local_ip_address*

Description: Freeing the TCB block associated with the TCP connection that has closed.

TCP.044

Level: C-INFO

Short Syntax: TCP.044 Idle tmr fires frgn hst *foreign_ip_address* lcl hst *local_ip_address*

Long Syntax: TCP.044 Idle timer fires for connection between *foreign_ip_address* and *local_ip_address*

Description: Idle timer fires for TCP connection.

TCP.045

Level: C-INFO

Short Syntax: TCP.045 Rxmt tmr fires frgn hst *foreign_ip_address* lcl hst *local_ip_address*

Long Syntax: TCP.045 Retransmit timer fires for connection between *foreign_ip_address* and *local_ip_address*

Description: Retransmit timer fires for TCP connection.

TCP.046

Level: C-INFO

Short Syntax: TCP.046 State trnstn frm ESTAB to FINWAIT *source_ip_address* -> *destination_ip_address* dst prt *tcp_src_port* src prt *tcp_dst_port*

Long Syntax: TCP.046 State transitioned from ESTABLISHED to FINWAIT *source_ip_address* -> *destination_ip_address* dst prt *tcp_src_port* src prt *tcp_dst_port*

Description: State of tcp connection has transitioned from ESTABLISHED to FINWAIT - send FIN, and now

waiting for FIN-ACK to arrive.

TCP.047

Level: C-INFO

Short Syntax: TCP.047 State trnstn to CLOSED *source_ip_address* -> *destination_ip_address* dst prt *tcp_src_port* src prt *tcp_dst_port*

Long Syntax: TCP.047 State transitioned to CLOSED *source_ip_address* -> *destination_ip_address* dst prt *tcp_src_port* src prt *tcp_dst_port*

Description: State of tcp connection has transitioned to CLOSED.

TCP.048

Level: C-INFO

Short Syntax: TCP.048 Rcvd data after CLOSE issued and zero wndw, snd RST *source_ip_address* -> *destination_ip_address* dst prt *tcp_src_port* src prt *tcp_dst_port*

Long Syntax: TCP.048 Received data after CLOSE was issued, and window is zero, send RESET *source_ip_address* -> *destination_ip_address* dst prt *tcp_src_port* src prt *tcp_dst_port*

Description: TCP connection is CLOSING due to application requesting a CLOSE. After the window shrinks to zero, discard any packets received. This is based on the half-duplex TCP close sequence.

TCP.049

Level: C-INFO

Short Syntax: TCP.049 Rcvd NACK

Long Syntax: TCP.049 Received NACK

Description: The other side has send an old ACK with zero data length - we treat this as a NACK.

TCP.050

Level: C-INFO

Short Syntax: TCP.050 Rcvd ACK for Keep Alive

Long Syntax: TCP.050 Received Acknowledge for the keep alive packet sent

Description: The other side has acknowledged the keep alive packet. The keep alive packet is sent if keep alive is enabled on this tcp connection, and the connection has been idle.

TCP.051

Level: C-INFO

Short Syntax: TCP.051 Lcl wndw zero

Long Syntax: TCP.051 Local window zero

Description: The local window advertised is zero. The application is not draining the tcp receive buffer fast enough.

TCP.052

Level: C-INFO

Short Syntax: TCP.052 snd FIN seq *seq_num*, ack *ack_num*

Long Syntax: TCP.052 send FIN sequence number *seq_num*, acknowledge number *ack_num*

Description: The TCP connection is closing, and we sent a FIN.

TCP.053

Level: C-INFO

Short Syntax: TCP.053 get buf fld - cannot snd pkt

Long Syntax: TCP.053 get buf failed - cannot send packet

Description: The router is running out of iorbs, getbuf failed, so we cannot send a packet.

TCP.054

Level: C-INFO

Short Syntax: TCP.054 xmit buf too large (*requested_amount*), clipped to *clipped_amount*

Long Syntax: TCP.054 transmit buffer too large for listen/open (*requested_amount*), clipped to (*clipped_amount*)

Description: The transmit buffer size requested for a TCP connection is too large to be allocated by the system. TCP has selected in its place the largest chunk size available in the system.

TCP.055

Level: C-INFO

Short Syntax: TCP.055 recv buf too large (*requested_amount*), clipped to *clipped_amount*

Long Syntax: TCP.055 receive buffer too large for listen/open (*requested_amount*), clipped to (*clipped_amount*)

Description: The receive buffer size requested for a TCP connection is too large to be allocated by the system. TCP has selected in its place the largest chunk size available in the system.

TCP.056

Level: UE-ERROR

Short Syntax: TCP.056 6 Duplicate acks with seqnum *seq_num* ack num *ack_num* wndw *window*

Long Syntax: TCP.056 6 ACKs seen with with sequence number *seq_num* and acknowledge number *ack_num* window *window*

Description: Fast Retransmit has sent the missing segment. New data should have been ACKed. Other end might be down or congested.

TCP.057

Level: UE-ERROR

Short Syntax: TCP.057 New data (*tcp_ack*) ACKed after *tcp_dupack* dups

Long Syntax: TCP.057 Sequence number *tcp_ack* ACKnowledged after processing *tcp_dupack* duplicate ACKs

Description: TCP counts ACKs which acknowledge data which was previously acknowledged. After 3 exactly duplicate ACKs are received, the apparently lost data segment is retransmitted. Whenever new data is acknowledged, this message is printed (with the total number of exactly duplicate ACKs) and the counter is cleared.

TCP.058

Level: U-INFO

Short Syntax: TCP.058 Echo *foreign_ip_address*(*foreign_port_number*) -> *local_ip_address*(*local_port_number*)

Long Syntax: TCP.058 Connection to Echo from *foreign_ip_address* port *foreign_port_number* to *local_ip_address* port *local_port_number*

Description: A connection has been established to Echo. Echo will return the data it receives to the sender.

TCP.059

Level: UI-ERROR

Short Syntax: TCP.059 rcvd pkt *source_ip_address* -> *destination_ip_address* dst prt *tcp_port* no cnn

Long Syntax: TCP.059 received packet *source_ip_address* -> *destination_ip_address* with destination port *tcp_port* has no tcp connection

Description: TCP has received a packet with an invalid tcp port number.

TCP.060

Level: UI-ERROR

Short Syntax: TCP.060 rcvd invld SYN in wndw *source_ip_address* -> *destination_ip_address* dst prt *tcp_port* kill cnn

Long Syntax: TCP.060 received invalid SYN packet *source_ip_address* -> *destination_ip_address* with destination port *tcp_port*, kill connection

Description: TCP has received an illegal SYN packet, so kill the connection.

TCP.063

Level: UI-ERROR

Short Syntax: TCP.063 drp seg *source_ip_address* -> *destination_ip_address* dst prt *tcp_port* rsn *reject_code* snd ACK

Long Syntax: TCP.063 dropped segment *source_ip_address* -> *destination_ip_address* with destination port *tcp_port*, reason *reject_code*, send a valid ACK in response

Description: TCP has rejected a segment. Reject codes are as follows: Reject codes: 1 - Seg len = 0, Rcv win > 0, seqnum < tcb_ack 2 - Seg len = 0, Rcv win = 0, seqnum != tcb_ack 3 - Seg len > 0, Rcv win > 0, winend < tcb_ack 4 - Seg len > 0, Rcv win = 0. 5 - Seg len = 0, Rcv win > 0, seqnum >= winend 6 - Seg len > 0, Rcv win > 0, seqnum >= winend Note: we only ACK if the segment received was a non RST segment.

TCP.064

Level: C-INFO

Short Syntax: TCP.064 prcss FIN in ESTAB|SYNRCVD state frgn hst *ip_address* lcl hst *ip_address* dprt *dst_port* sprt *src_port*

Long Syntax: TCP.064 process a received FIN; current state is SYNRCVD|ESTAB, foreign host *ip_address* local host *ip_address* destination port *dst_port* source port *src_port*

Description: TCP processing FIN while in SYNRCVD|ESTAB state.

TCP.065

Level: UI-ERROR

Short Syntax: TCP.065 frgn prt illgl close of wndw frgn hst *ip_address* lcl hst *ip_address* dprt *dst_port* sprt *src_port*

Long Syntax: TCP.065 foreign port closed the advertised window illegally foreign host *ip_address* local host *ip_address* destination port *dst_port* source port *src_port*

Description: The other side has been deaf and mute, and the the foreign window seems to have been closed illegally; send a RST.

TCP.066

Level: P-TRACE

Short Syntax: TCP.066 rcvd TCP pkt *source_ip_address* -> *destination_ip_address* dst prt *tcp_port*

Long Syntax: TCP.066 received packet *source_ip_address* -> *destination_ip_address* with destination port *tcp_port*

Description: TCP has received a packet.

TCP.067

Level: UI-ERROR

Short Syntax: TCP.067 TCP snd rst to hst *source_ip_address*

Long Syntax: TCP.067 TCP sending RESET to host *source_ip_address*

Description: TCP is sending a RESET segment to the other side.

TCP.068

Level: C-INFO

Short Syntax: TCP.068 TCP cnn clsd frgn hst *foreign_ip_address* lcl hst *local_ip_address*

Long Syntax: TCP.068 TCP connection closed, foreign host *foreign_ip_address*, local hst *local_ip_address*

Description: TCP connection is closed - notifying the application.

TCP.069

Level: C-INFO

Short Syntax: TCP.069 Frng TCB for frgn hst *foreign_ip_address* lcl hst *local_ip_address*

Long Syntax: TCP.069 Freeing TCB block for connection between *foreign_ip_address* and *local_ip_address*

Description: Freeing the TCB block associated with the TCP connection that has closed.

TCP.071

Level: C-INFO

Short Syntax: TCP.071 Rxmt tmr fires frgn hst *foreign_ip_address* lcl hst *local_ip_address*

Long Syntax: TCP.071 Retransmit timer fires for connection between *foreign_ip_address* and *local_ip_address*

Description: Retransmit timer fires for TCP connection.

TCP.072

Level: C-INFO

Short Syntax: TCP.072 State trnstn frm ESTAB to FINWAIT *source_ip_address* -> *destination_ip_address* dst prt *tcp_src_port* src prt *tcp_dst_port*

Long Syntax: TCP.072 State transitioned from ESTABLISHED to FINWAIT *source_ip_address* -> *destination_ip_address* dst prt *tcp_src_port* src prt *tcp_dst_port*

Description: State of tcp connection has transitioned from ESTABLISHED to FINWAIT - send FIN, and now waiting for FIN-ACK to arrive.

TCP.073

Level: C-INFO

Short Syntax: TCP.073 State trnstn to CLOSED *source_ip_address* -> *destination_ip_address* dst prt *tcp_src_port* src prt *tcp_dst_port*

Long Syntax: TCP.073 State transitioned to CLOSED *source_ip_address* -> *destination_ip_address* dst prt *tcp_src_port* src prt *tcp_dst_port*

Description: State of tcp connection has transitioned to CLOSED.

TCP.074

Level: C-INFO

Short Syntax: TCP.074 Rcvd data after CLOSE issued and zero wndw, snd RST *source_ip_address* -> *destination_ip_address* dst prt *tcp_src_port* src prt *tcp_dst_port*

Long Syntax: TCP.074 Received data after CLOSE was issued, and window is zero, send RESET *source_ip_address* -> *destination_ip_address* dst prt *tcp_src_port* src prt *tcp_dst_port*

Description: TCP connection is CLOSING due to application requesting a CLOSE. After the window shrinks to zero, discard any packets received. This is based on the half-duplex TCP close sequence.

TCP.075

Level: C-INFO

Short Syntax: TCP.075 PMTU chg, fhost *destination_ip_address*, PMTU *pmtu*, old MSS *old_mss* new MSS *new_mss*

Long Syntax: TCP.075 Path MTU changed to foreign host *destination_ip_address*, PMTU = *pmtu*, old MSS = *old_mss* new MSS = *new_mss*

Description: A Packet Too Big ICMP was received indicating the local host sent a TCP segment that was too big for one of the links in the path to this foreign host.

TCP.076

Level: C-INFO

Short Syntax: TCP.076 PMTU chg, rxmt *source_ip_address* -> *destination_ip_address*, dst prt *tcp_src_port* src prt *tcp_dest_port*

Long Syntax: TCP.076 PMTU change, retransmit *source_ip_address* -> *destination_ip_address*, dst prt *tcp_src_port* src prt *tcp_dest_port*

Description: Retransmit the segment that was dropped as indicated by a Packet Too Big ICMP.

TCP.077

Level: C-INFO

Short Syntax: TCP.077 xmt *source_ip_address* -> *destination_ip_address*, dst prt *tcp_src_port* src prt *tcp_dest_port*

Long Syntax: TCP.077 transmit data *source_ip_address* -> *destination_ip_address*, dst prt *tcp_src_port* src prt *tcp_dest_port*

Description: Transmit data to Link Local Address.

TCP.078

Level: UI-ERROR

Short Syntax: TCP.078 xmt failed *source_ip_address* -> *destination_ip_address*, dst prt *tcp_src_port* src prt *tcp_dest_port*

Long Syntax: TCP.078 transmit failed *source_ip_address* -> *destination_ip_address*, dst prt *tcp_src_port* src prt *tcp_dest_port*

Description: Transmit failed for Link Local Address.

TCP.079

Level: U-INFO

Short Syntax: TCP.079 Echo *foreign_ip_address*(*foreign_port_number*) -> *local_ip_address*(*local_port_number*)

Long Syntax: TCP.079 Connection to Echo from *foreign_ip_address* port *foreign_port_number* to *local_ip_address* port *local_port_number*

Description: A connection has been established to Echo. Echo will return the data it receives to the sender.

TCP.080

Level: UE-ERROR

Short Syntax: TCP.080 Persistent Listen *local_ip_address*(*local_port_number*) denied. *total_current_open* sockets open

Long Syntax: TCP.080 SYN pkt has been received for *local_ip_address*(*local_port_number*). *total_current_open* bytes are in SYNRCVD state

Description: A connection is attempted to %I(%d). TCP should clone a LISTEN, but we have reached our connection limit. This may indicate that the router has insufficient resources for the clients using it. It may indicate that it is experiencing a flood of SYN packets as a denial of service attack.

TCP.081

Level: UE-ERROR

Short Syntax: TCP.081 Persistent Listen *local_ip_address*(*local_port_number*) replaced

Long Syntax: TCP.081 A connection (*local_ip_address*(*local_port_number*)) has completed. A LISTEN was replaced

Description: A previous attempt to replace this LISTEN was denied due to too many existing persistent LISTENs. A connection for that server has been established, and there was sufficient memory to replace the LISTEN.

TCP.082

Level: C-INFO

Short Syntax: TCP.082 rcvd pkt *source_ip_address* -> *destination_ip_address* dst prt *tcp_port* sending to OS

Long Syntax: TCP.082 received packet *source_ip_address* -> *destination_ip_address* with destination port *tcp_port*. Router passing packet to OS

Description: TCP has received a packet which it is passing to OS. The operating system might have this connection. Otherwise the operating system should return a reset.

TCP.083

Level: C-INFO

Short Syntax: TCP.083 TCP cnn request *source_ip_address* -> *destination_ip_address*(*tcp_port*) refused. Router is at cnn max(*tcp_max_cnn*)

Long Syntax: TCP.083 received packet *source_ip_address* -> *destination_ip_address*(dst port *tcp_port*). Router supports max of *tcp_max_cnn* concurrent connections

Description: TCP has received a connection request

when the router already has as many TCP connections as it can support. This limit protects other router components from memory exhaustion.

TCP.084

Level: UI-ERROR

Short Syntax: TCP.084 TCP retransmiting *data_size* bytes *local_host*(*local_port*) -> *foreign_host*(*foreign_port*) with no buffer.

Long Syntax: TCP.084 TCP retransmiting *data_size* bytes *local_host*(*local_port*) -> *foreign_host*(*foreign_port*) with no buffer.

Description: TCP was in idle mode when the application called close. There is no transmit buffer, and we need to only set some control bit in the frame

---

TCP.085

Level: UI-ERROR

Short Syntax: TCP.085 TCP reset connection *local_host*(*local_port*) -> *foreign_host*(*foreign_port*). No xbuf for *data_size* data bytes

Long Syntax: TCP.085 TCP reset connection *local_host*(*local_port*) -> *foreign_host*(*foreign_port*). No xbuf for *data_size* data bytes

Description: There is no transmit buffer, and we need to retransmit multiple bytes. It's not clear how this happens, and it's occurrance should be reported to IBM service.

TCP.086

Level: UI-ERROR

Short Syntax: TCP.086 New cnn *foreign_ip_addr*(*foreign_ip_port*)-> *local_ip_addr*(*local_ip_port*) has no master socket

Long Syntax: TCP.086 New TCP handshake has completed from *foreign_ip_addr*(*foreign_ip_port*) to *local_ip_addr*(*local_ip_port*), but the master socket is no longer available

Description: The master socket which this TCB points to has not been passed to listen(). So, either the master socket has been closed, or an application is using an invalid socket.

Chapter 120. Trivial File Transfer Protocol (TFTP)

This chapter describes Trivial File Transfer Protocol (TFTP) messages. For information on message content and how to use the message, refer to the Introduction.

TFTP.001

Level: UI-ERROR

Short Syntax: TFTP.001 xfer max exceeded

Long Syntax: TFTP.001 simultaneous transfer maximum exceeded

Description: There is a maximum number of simultaneous TFTP transfers supported; a request (either local or remote) was made while this maximum number of TFTP transfers were already in progress.

TFTP.002

Level: UI-ERROR

Short Syntax: TFTP.002 unknwn rqst opcode: *opcode*

Long Syntax: TFTP.002 unknown TFTP request opcode: *opcode*

Description: Unknown TFTP request opcode was received.

TFTP.003

Level: UI-ERROR

Short Syntax: TFTP.003 accs viol fn: *filename_requested*

Long Syntax: TFTP.003 access violation filename: *filename_requested*

Description: A TFTP file transfer request (either local or remote) failed because of a TFTP access control violation.

TFTP.004

Level: UI-ERROR

Short Syntax: TFTP.004 no UDP port avail

Long Syntax: TFTP.004 no UDP port available

Description: A TFTP file transfer request (either local or remote) failed because no UDP port was available.

TFTP.005

Level: UI-ERROR

Short Syntax: TFTP.005 no bfr avail

Long Syntax: TFTP.005 no buffer available

Description: A TFTP request failed for lack of buffers.

TFTP.006

Level: CI-ERROR

Short Syntax: TFTP.006 2nd srvr regd

Long Syntax: TFTP.006 second TFTP server registered

Description: Only one TFTP server can be active at any one time; a second server has been registered by software and the previous server has been deactivate.

TFTP.007

Level: UE-ERROR

Short Syntax: TFTP.007 unexp data pkt rcv

Long Syntax: TFTP.007 unexpected TFTP data packet received

Description: A TFTP packet on an inactive connection was received.

TFTP.008

Level: UE-ERROR

Short Syntax: TFTP.008 unexp xfer term: *reason_code*, tid *transfer_id*

Long Syntax: TFTP.008 TFTP transfer unexpected termination: *reason_code*, transfer id *transfer_id*

Description: A TFTP transfer has terminated prematurely; reason code provided.

TFTP.009

Level: C-INFO

Short Syntax: TFTP.009 normal xfer cmpltd, tid *transfer_id*

Long Syntax: TFTP.009 TFTP transfer completed normally, transfer id *transfer_id*

Description: A TFTP transfer has completed normally.

TFTP.010

Level: CE-ERROR

Short Syntax: TFTP.010 sorc appren avrtd, blk *block* exp *expected_block* tid *transfer_id*

Long Syntax: TFTP.010 sorcerer's apprentice bug avoided, block *block* expected *expected_block* transfer id *transfer_id*

Description: The fix to a bug called the sorcerer's apprentice is to not retransmit old TFTP data packets in response to out-of-sequence TFTP acks; this has just occurred. The block number of the ack received and of the ack expected are displayed.

TFTP.011

Level: UE-ERROR

Short Syntax: TFTP.011 xfer timeout, tid *transfer_id*

Long Syntax: TFTP.011 TFTP transfer network timeout, transfer id *transfer_id*

Description: TFTP transfer failed due to timeout on the network.

TFTP.012

Level: U-INFO

Short Syntax: TFTP.012 ack pkt retrns, blk *block* tid *transfer_id*

Long Syntax: TFTP.012 TFTP ack packet retransmission, block number *block* transfer id *transfer_id*

Description: A TFTP ack packet was retransmitted in response to an out-of-sequence data packet received.

TFTP.013

Level: U-INFO

Short Syntax: TFTP.013 data pkt retrns, blk *block* tid *transfer_id*

Long Syntax: TFTP.013 TFTP data packet retransmission, block number *block* transfer id *transfer_id*

Description: A TFTP packet was retransmitted on expiration of a timer.

TFTP.014

Level: C-INFO

Short Syntax: TFTP.014 rmt *type* req accptd, tid *transfer_id*

Long Syntax: TFTP.014 remote TFTP *type* request accepted, transfer id *transfer_id*

Description: A remote TFTP transfer request has been accepted.

TFTP.015

Level: C-INFO

Short Syntax: TFTP.015 data pkt sent, blk *block* tid *transfer_id*

Long Syntax: TFTP.015 data packet sent, block number *block* transfer id *transfer_id*

Description: A TFTP data packet has been sent.

TFTP.016

Level: C-INFO

Short Syntax: TFTP.016 ack pkt sent, blk *block* tid *transfer_id*

Long Syntax: TFTP.016 ack packet sent, block number *block* transfer id *transfer_id*

Description: A TFTP ack packet has been sent.

TFTP.017

Level: U-INFO

Short Syntax: TFTP.017 req pkt retrns, tid *transfer_id*

Long Syntax: TFTP.017 request packet retransmitted, transfer id *transfer_id*

Description: A TFTP request packet has been retransmitted

TFTP.018

Level: UE-ERROR

Short Syntax: TFTP.018 remt req rej'd: *reason optional_details*

Long Syntax: TFTP.018 remote request rejected: *reason optional_details*

Description: A remote TFTP request was rejected for the reason shown. An optional second parameter provides further details.

TFTP.019

Level: C-INFO

Short Syntax: TFTP.019 *type* req sent, tid *transfer_id*

Long Syntax: TFTP.019 locally originated *type* request sent, transfer id *transfer_id*

Description: A locally originated TFTP request has been sent.

TFTP.020

Level: C-INFO

Short Syntax: TFTP.020 xfer abrted by usr

Long Syntax: TFTP.020 locally originated TFTP transfer aborted at the console

Description: Locally originated TFTP transfer was aborted at the console.

TFTP.021

Level: C-INFO

Short Syntax: TFTP.021 ack pkt rcvd blk *block* tid *trans_id*

Long Syntax: TFTP.021 ack packet received, block *block* transfer id *trans_id*

Description: A TFTP ack packet has been received.

TFTP.022

Level: C-INFO

Short Syntax: TFTP.022 data pkt rcvd blk *block* tid *trans_id*

Long Syntax: TFTP.022 data packet received, block *block* transfer id *trans_id*

Description: A TFTP data packet has been received.

TFTP.023

Level: C-INFO

Short Syntax: TFTP.023 unexp err pkt rcvd code *errcode colon_and_openquote errmsg closequote*

Long Syntax: TFTP.023 unexpected error packet received, code *errcode colon_and_openquote errmsg closequote*

Description: A unexpected TFTP error packet has been received.

TFTP.024

Level: UE-ERROR

Short Syntax: TFTP.024 lcl dev err *errmsg*

Long Syntax: TFTP.024 local device error, *errmsg*

Description: Error accessing one of the local device. Errmsg describes the actual device and the type of error.

TFTP.025

Level: ALWAYS

Short Syntax: TFTP.025 Starting tftp of file *configFile* from *serverIpAddr*

Long Syntax: TFTP.025 Starting tftp of file *configFile* from *serverIpAddr*

Description: EasyStart is trying to download a specified file from a specified host.

TFTP.026

Level: ALWAYS

Short Syntax: TFTP.026 Open failed.

Long Syntax: TFTP.026 Open failed.

Description: Open failed.

TFTP.027

Level: ALWAYS

Short Syntax: TFTP.027 Transfer completed successfully. Writing to NVRAM.

Long Syntax: TFTP.027 Transfer completed successfully. Writing to NVRAM.

Description: Transfer completed successfully. Writing to NVRAM.

TFTP.028

Level: ALWAYS

Short Syntax: TFTP.028 Writing to NVRAM completed.

Long Syntax: TFTP.028 Writing to NVRAM completed.

Description: Writing to NVRAM completed.

TFTP.029

Level: ALWAYS

Short Syntax: TFTP.029 Transfer stopped due to a failure.

Long Syntax: TFTP.029 Transfer stopped due to a failure.

Description: Transfer stopped due to a failure.

TFTP.030

Level: U-INFO

Short Syntax: TFTP.030 ack blk *block* to duplicate data blk *duplicate*, tid *transfer_id*

Long Syntax: TFTP.030 TFTP ack block *block* sent as response to duplicate block *duplicate* - tid *transfer_id*

Description: A TFTP ack packet was sent in response to a duplicate data packet.

TFTP.031

Level: UE-ERROR

Short Syntax: TFTP.031 blk wrapped. max rcvd blk *block* tid *trans_id*

Long Syntax: TFTP.031 block wrapped. maximum received block was *block* transfer id *trans_id*

Description: The file to be transferred is very large. More than 0xffff or 0x7fff blocks of 512 bytes are required to transfer the file. The remote server is unable to handle the wrapping of the block number back to 0. A new write request will be started to continue to transfer the remainder of the file.

TFTP.032

Level: UI-ERROR

Short Syntax: TFTP.032 no mem avail

Long Syntax: TFTP.032 no memory available

Description: A TFTP request failed for lack of memory.

TFTP.033

Level: C-INFO

Short Syntax: TFTP.033 *req* req sent for fn *fn*, tid *transfer_id*

Long Syntax: TFTP.033 *req* request sent for file name *fn*, transfer id *transfer_id*

Description: TFTP request has been sent for the specified file.

TFTP.034

Level: UI-ERROR

Short Syntax: TFTP.034 No known path to destination.

Long Syntax: TFTP.034 No known path to destination.

Description: Transfer could not be started because there was no known path to the destination of the transfer request.

Chapter 121. User Datagram Protocol (UDP)

This chapter describes User Datagram Protocol (UDP) messages. For information on message content and how to use the message, refer to the Introduction.

UDP.003

Level: UE-ERROR

Short Syntax: UDP.003 dsc pkt frm *source_ip_address* bd len *length*

Long Syntax: UDP.003 Discarded packet from *source_ip_address*, bad length *length*

Description: This message is generated when a packet is discarded because it had a UDP length greater than its IP length.

UDP.004

Level: UE-ERROR

Short Syntax: UDP.004 bd cksm clc *checksum* rcv *checksum*

Long Syntax: UDP.004 Bad checksum - calculated *checksum*, received *checksum*

Description: This message is generated when a packet is discarded because it had a bad checksum.

UDP.005

Level: U-TRACE

Short Syntax: UDP.005 rcvd pkt frm (*source_IP_address*, prt *udp_port_number*, nt *Network ID*)

Long Syntax: UDP.005 received packet from (*source_IP_address*, port *udp_port_number*, net *Network ID*)

Description: A UDP datagram has been received on a particular interface. The port number is included in the message.

UDP.006

Level: U-TRACE

Short Syntax: UDP.006 fwd pkt to *destination_IP_address* on prt *udp_port_number*

Long Syntax: UDP.006 Forwarding packet to *destination_IP_address* on udp port *udp_port_number*

Description: A UDP datagram is being forwarded to a particular destination. The port number is included in the message.

UDP.007

Level: U-INFO

Short Syntax: UDP.007 echo *source_ip_address*(*source_port_number*) -> *destination_ip_address*(*destination_port_number*)

Long Syntax: UDP.007 UDP Echo received datagram from *source_ip_address* port *source_port_number* to *destination_ip_address* port *destination_port_number*

Description: UDP Echo received a datagram. It will return the datagram to the sender.

Chapter 122. User Datagram Protocol for IPv6 (UDP6)

This chapter describes User Datagram Protocol for IPv6 (UDP6) messages. For information on message content and how to use the message, refer to the Introduction.

UDP6.001

Level: U-INFO

Short Syntax: UDP6.001 pkt *source_ip_address* -> *destination_ip_address* port *port_number* no srvr

Long Syntax: UDP6.001 Packet from *source_ip_address* for *destination_ip_address* port *port_number*, no server

Description: This message is generated when a packet is discarded because either UDP6 is not installed in the router or UDP6 is installed but the packet was for the multicast address and no application supports the UDP destination port.

UDP6.002

Level: U-INFO

Short Syntax: UDP6.002 pkt *source_ip_address* -> *destination_ip_address* port *port_number* no srvr

Long Syntax: UDP6.002 Packet from *source_ip_address* for *destination_ip_address* port *port_number*, no server

Description: This message is generated when a packet is discarded because either UDP6 is not installed in the router or UDP6 is installed but no UDP6 application supports the UDP6 destination port. The packet was addressed to the router.

UDP6.003

Level: UE-ERROR

Short Syntax: UDP6.003 discarded pkt from *source_ip_address* bad len *length*

Long Syntax: UDP6.003 Discarded packet from *source_ip_address*, bad length *length*

Description: This message is generated when a packet is discarded because it had a UDP length greater than its IP length.

Cause: There are many possible causes of problem. The problem may originate from the sending node of this UDP6 packet, or the packet may be corrupted either during the tranmission or at this router.

Action: Trace the packet right after the sending node and right before the packet arrives at this router.

UDP6.004

Level: UE-ERROR

Short Syntax: UDP6.004 bad checksum clc *checksum* rcv *checksum*

Long Syntax: UDP6.004 Bad checksum - calculated *checksum*, received *checksum*

Description: This message is generated when a packet is discarded because it had a bad checksum.

Cause: Bad checksum. Either the sending node did not perform UDP6 checksum correctly or it is possible that the UDP6 portion of this packet is corrupted.

Action: Trace the packet right after the sending node and right before the packet arrives at this router. Key on the UDP6 checksum field.

UDP6.005

Level: C-TRACE

Short Syntax: UDP6.005 received pkt from (*source_IP_address*, prt *udp_port_number*, nt *Network ID*)

Long Syntax: UDP6.005 received packet from (*source_IP_address*, port *udp_port_number*, net *Network ID*)

Description: A UDP6 datagram has been received on a particular interface. The port number is included in the message.

UDP6.006

Level: C-TRACE

Short Syntax: UDP6.006 forwarding pkt to *destination_IP_address* on prt *udp_port_number*

Long Syntax: UDP6.006 Forwarding packet to *destination_IP_address* on udp port *udp_port_number*

Description: A UDP6 datagram is being forwarded to a particular destination. The port number is included in the message.

UDP6.007

Level: UE-ERROR

Short Syntax: UDP6.007 zero checksum rcv *checksum*

Long Syntax: UDP6.007 zero checksum - received *checksum*

Description: This message is generated when a packet is discarded because it had a zero checksum. (Note: UDP6 checksum is not optional like in UDP in IPV4).

Cause: Either the sending node did not perform UDP6 checksum or the checksum field in UDP6 header field is corrupted.

Action: Trace the packet right after the sending node and right before the packet arrives at this router. Key on the UDP6 checksum field.

Chapter 123. VCRM

This chapter describes VCRM messages. For information on message content and how to use the message, refer to the Introduction.

VCRM.001

Level: UI-ERROR

Short Syntax: VCRM.001 VCRM No Rsrcs *Failure_type Routine*

Long Syntax: VCRM.001 VCRM Insufficent resources of type *Failure_type* in *Routine*

Description: VCRM Could not obtain needed resources (memory)

VCRM.002

Level: UI-ERROR

Short Syntax: VCRM.002 VCRM *Correlator_type* Crrltr *Correlator* Msmtch *Routine*

Long Syntax: VCRM.002 VCRM *Correlator_type* Correlator *Correlator* Mismatch in routine *Routine*

Description: VCRM Unknown correlator of specified type encountered

VCRM.003

Level: UI-ERROR

Short Syntax: VCRM.003 VCRM Unxpctd *Condition Code* from *Routine*

Long Syntax: VCRM.003 VCRM Unexpected *Condition Code* from *Routine*

Description: VCRM Unexpected condition occured

VCRM.004

Level: P-TRACE

Short Syntax: VCRM.004 VCRM *Event_type* Evnt *Added_info1 Added_info2 Added_info3* in *Routine*

Long Syntax: VCRM.004 VCRM *Event_type* event occured *Added_info1 Added_info2 Added_info3* in routine *Routine*

Description: VCRM A trace event in VCRM occured

VCRM.005

Level: U-TRACE

Short Syntax: VCRM.005 *var1= val1*

Long Syntax: VCRM.005 component var1 *var1= val1*

Description: generic dump of intServ spec content (one floating point field)

VCRM.006

Level: U-TRACE

Short Syntax: VCRM.006 *var1= val1*

Long Syntax: VCRM.006 component var1 *var1= val1*

Description: generic dump of intServ spec content (one integer field)

VCRM.007

Level: P-TRACE

Short Syntax: VCRM.007 VCRM *event* avg= *avg* peak= *peak* burst= *burst* minpkt= *minmtu* maxpkt= *maxmtu* on i/f *oif* to *gateway*

Long Syntax: VCRM.007 VCRM *event* evnt; avgrate= *avg* peak= *peak* burst= *burst* minTU= *minmtu* maxTU= *maxmtu* on out i/f *oif* to nxt-hop *gateway*

Description: dump out the VCRM add/delete resveration QoS parameters, output i/f, and next-hop gateway.

VCRM.008

Level: U-TRACE

Short Syntax: VCRM.008 *var1= v1*; *var2= v2*; *var3= v3*; *var4= v4*

Long Syntax: VCRM.008 component var1 *var1= v1*; var2 *var2= v2*; var3 *var3= v3*; var4 *var4= v4*

Description: A generic trace for unusual events; var1 shows module name and first trace variable name, v1 shows the first trace variable value; var2 show second trace variable name, v2 shows second trace variable value, and so on (in hex values).

VCRM.009

Level: UI-ERROR

Short Syntax: VCRM.009 VCRM Unxpctd *Condition Code ErrorMsg* from *Routine*

Long Syntax: VCRM.009 VCRM Unexpected *Condition Code ErrorMsg* from *Routine*

Description: VCRM Unexpected condition occured

VCRM.010

Level: U-TRACE

Short Syntax: VCRM.010 *var1= v1*; *var2= v2*; *var3= v3*; *var4= v4*

Long Syntax: VCRM.010 component var1 *var1= v1*; var2 *var2= v2*; var3 *var3= v3*; var4 *var4= v4*

Description: A generic trace for unusual events; var1 shows module name and first trace variable name, v1 shows the first trace variable value; var2 show second trace variable name, v2 shows second trace variable value, and so on (in decimal values).

VCRM.011

Level: P-TRACE

Short Syntax: VCRM.011 DSRM strm *streamId action* chng= *deltaBw* sum= *sumBw Module*

Long Syntax: VCRM.011 DSRM stream id *streamId action deltaBw* total amount *sumBw Module*

Description: A DS stream used by RSVP changed its reservation amount.

Chapter 124. Virtual Lan (VLAN) ELS

This chapter describes Virtual Lan (VLAN) ELS messages. For information on message content and how to use the message, refer to the Introduction.

VLAN.001

Level: C-TRACE

Short Syntax: VLAN.001 TR IP arp

Long Syntax: VLAN.001 Received a token ring IP arp frame

Description: A token ring IP arp frame was received

VLAN.002

Level: C-TRACE

Short Syntax: VLAN.002 ENET IP arp DX

Long Syntax: VLAN.002 Received an ethernet IP arp frame in DIX encapsulation

Description: An ethernet IP arp frame in DIX encapsulation was received

VLAN.003

Level: C-TRACE

Short Syntax: VLAN.003 ENET IP arp SNAP

Long Syntax: VLAN.003 Received an ethernet IP arp frame in SNAP encapsulation

Description: An ethernet IP arp frame in SNAP encapsulation was received

VLAN.004

Level: C-TRACE

Short Syntax: VLAN.004 TR IPX 802.2

Long Syntax: VLAN.004 Received a token ring IPX frame in 802.2 encapsulation

Description: A token ring IPX frame in 802.2 encapsulation was received

VLAN.005

Level: C-TRACE

Short Syntax: VLAN.005 TR IPX SNAP

Long Syntax: VLAN.005 Received a token ring IPX frame in SNAP encapsulation

Description: A token ring IPX frame in SNAP encapsulation was received

VLAN.006

Level: C-TRACE

Short Syntax: VLAN.006 ENET IPX DIX

Long Syntax: VLAN.006 Received an ethernet IPX frame in DIX encapsulation

Description: An ethernet IPX frame in DIX encapsulation was received

VLAN.007

Level: C-TRACE

Short Syntax: VLAN.007 ENET IPX raw

Long Syntax: VLAN.007 Received an ethernet IPX frame in raw encapsulation

Description: An ethernet IPX frame in raw encapsulation was received

VLAN.008

Level: C-TRACE

Short Syntax: VLAN.008 ENET IPX 802.2

Long Syntax: VLAN.008 Received an ethernet IPX frame in 802.2 encapsulation

Description: An ethernet IPX frame in 802.2 encapsulation was received

VLAN.009

Level: C-TRACE

Short Syntax: VLAN.009 ENET IPX SNAP

Long Syntax: VLAN.009 Received an ethernet IPX frame in SNAP encapsulation

Description: An ethernet IPX frame in SNAP encapsulation was received

VLAN.010

Level: C-TRACE

Short Syntax: VLAN.010 TR NTBS

Long Syntax: VLAN.010 Received a token ring netbios frame

Description: A token ring netbios frame was received from

VLAN.011

Level: C-TRACE

Short Syntax: VLAN.011 ENET NTBS LLC

Long Syntax: VLAN.011 Received an ethernet netbios LLC frame

Description: An ethernet netbios LLC frame was received from

VLAN.012

Level: C-TRACE

Short Syntax: VLAN.012 ENET NTBS

Long Syntax: VLAN.012 Received an ethernet netbios frame

Description: An ethernet netbios frame was received

VLAN.013

Level: C-TRACE

Short Syntax: VLAN.013 Discard packet source MAC *sourceMac[0]*

Long Syntax: VLAN.013 The packet from the indicated MAC *sourceMac[0]* was discarded. The port map is set to zero

Description: The packet was discarded. The port map is set to zero. This could be due to IP Cut-Thru being disabled or no matching IPX encapsulation found.

Action: None.

VLAN.014

Level: C-TRACE

Short Syntax: VLAN.014 Discard packet (port excluded) prt *port_num* ifc *ifc_num* MAC *sourceMac[0]*

Long Syntax: VLAN.014 Discard packet (port excluded) prt *port_num* ifc *ifc_num* MAC *sourceMac[0]*

Description: A packet received on the indicated port and interface from the indicated MAC address was discarded due to port exclusion being set in a vlan. The port map is set to zero.

Action: None.

VLAN.015

Level: C-TRACE

Short Syntax: VLAN.015 Flood packet from MAC *sourceMac[0]*

Long Syntax: VLAN.015 The packet from the indicated MAC address *sourceMac[0]* will be flooded.

Description: The packet from the indicated MAC address will be flooded. The port map is unchanged.

Action: None.

VLAN.016

Level: C-TRACE

Short Syntax: VLAN.016 IP prt *port_num* ifc *ifc_num* MAC *sourceMac[0] sourceNet -> protocolOption*

Long Syntax: VLAN.016 Received an IP packet on port *port_num* interface *ifc_num* MAC *sourceMac[0]* source *sourceNet* -> destination *protocolOption*

Description: An IP packet was received on the indicated port and interface going from the source to the destination, from the indicated MAC address.

Action: None.

VLAN.017

Level: C-TRACE

Short Syntax: VLAN.017 IPX prt *port_num* ifc *ifc_num* MAC *sourceMac[0]* nt *sourceNet*

Long Syntax: VLAN.017 Received an IPX packet on port *port_num* interface *ifc_num* MAC address *sourceMac[0]* network *sourceNet*

Description: An IPX packet was received on the indicated port,interface, and network from the indicated MAC address.

Action: None.

VLAN.018

Level: C-TRACE

Short Syntax: VLAN.018 NTBS prt *port_num* ifc *ifc_num* MAC *sourceMac[0]*

Long Syntax: VLAN.018 Received a NETBIOS packet on port *port_num* interface *ifc_num* MAC address *sourceMac[0]*

Description: A NETBIOS packet was received on the indicated port and interface from the indicated MAC address.

Action: None.

VLAN.019

Level: C-TRACE

Short Syntax: VLAN.019 SLDW prt *port_num* ifc *ifc_num* MAC *sourceMac[0]*

Long Syntax: VLAN.019 Received a packet on port *port_num* interface *ifc_num* MAC address *sourceMac[0]* sldw fltr.

Description: A packet was received on the indicated port and interface from the indicated MAC addres.

Sliding window filters are defined and will be checked.

Action: None.

VLAN.020

Level: C-TRACE

Short Syntax: VLAN.020 fwd PMP[0-3] *vlanPmap[0] vlanPmap[1] vlanPmap[2] vlanPmap[3]*

Long Syntax: VLAN.020 forwarding port map [0][1][2][3] *vlanPmap[0] vlanPmap[1] vlanPmap[2] vlanPmap[3]*

Description: The forwarding port map where the packet will be sent out on.

Action: None.

VLAN.021

Level: C-TRACE

Short Syntax: VLAN.021 fwd PMP[4-7] *vlanPmap[0] vlanPmap[1] vlanPmap[2] vlanPmap[3]*

Long Syntax: VLAN.021 forwarding port map [4][5][6][7] *vlanPmap[0] vlanPmap[1] vlanPmap[2] vlanPmap[3]*

Description: The forwarding port map where the packet will be sent out on.

Action: None.

VLAN.022

Level: C-TRACE

Short Syntax: VLAN.022 agt *network* hndl *handle* PMP[0-3] *ageoutPmap[0] ageoutPmap[1] ageoutPmap[2] ageoutPmap[3]*

Long Syntax: VLAN.022 ageout *network* handle *handle* port map[4-7] *ageoutPmap[0] ageoutPmap[1] ageoutPmap[2] ageoutPmap[3]*

Description: Exclusive or of the active and forwarding port maps when a timer expires on the indicated network. The handle indicates which vlan it is.

Action: None.

VLAN.023

Level: C-TRACE

Short Syntax: VLAN.023 agt *network* hndl *handle* PMP[4-7] *ageoutPmap[4] ageoutPmap[5] ageoutPmap[6] ageoutPmap[7]*

Long Syntax: VLAN.023 ageout *network* handle *handle* port map[4-7] *ageoutPmap[4] ageoutPmap[5] ageoutPmap[6] ageoutPmap[7]*

Description: Exclusive or of the active and forwarding port maps when a timer IP,IPX, or NBS (NetBios). The handle indicates which vlan it is in the respective vlan.

Action: None.

VLAN.024

Level: C-TRACE

Short Syntax: VLAN.024 sld mtch prt *port_num* ifc *ifc_num* MAC *sourceMac[0]* strt *offsetType* offst *offset* cmpln *sldwCmpLen* vl *framePtr framePtr+4 framePtr+8*

Long Syntax: VLAN.024 Match on a sliding window filter occured on port *port_num* interface *ifc_num* MAC *sourceMac[0]* starting field *offsetType* offset *offset* compare length *sldwCmpLen* value *framePtr framePtr+4 framePtr+8*

Description: A match occured on a sliding window vlan with a packet received on the indicated port and interface from the MAC address. The match occurred at the indicated offset for the indicated length. Ten bytes of data are displayed.

Action: None.

VLAN.025

Level: C-TRACE

Short Syntax: VLAN.025 mac match prt *port_num* ifc *ifc_num* MAC *sourceMac[0]*

Long Syntax: VLAN.025 Match on a Mac Address filter occured on port *port_num* interface *ifc_num* MAC *sourceMac[0]*

Description: A match occured on a Mac Address vlan with a packet received on the indicated port and interface from the source MAC address.

Action: None.

VLAN.026

Level: C-TRACE

Short Syntax: VLAN.026 port match prt *port_num* ifc *ifc_num* MAC *sourceMac[0]*

Long Syntax: VLAN.026 Match on a Port-based filter occured on port *port_num* interface *ifc_num* MAC Address *sourceMac[0]*

Description: A match occurred on a Port-based vlan with a packet received on the indicated port and interface from the MAC address.

Action: None.

VLAN.027

Level: C-TRACE

Short Syntax: VLAN.027 IGMP Report prt *port_num* ifc *ifc_num* MAC *sourceMac[0]* Group *ipGroupAddress*

Long Syntax: VLAN.027 Received an IGMP Report on

port *port_num* interface *ifc_num* MAC *sourceMac[0]* Group *ipGroupAddress*

Description: An IGMP Report frame was received on the indicated port and interface for the indicated IP Multicast group, from the indicated MAC address.

Action: None.

VLAN.028

Level: C-TRACE

Short Syntax: VLAN.028 IP Mcast port *port_num* ifc *ifc_num* MAC *sourceMac[0]* Group *ipGroupAddress*

Long Syntax: VLAN.028 Received matching IP Multicast frame on port *port_num* interface *ifc_num* MAC *sourceMac[0]* to Group *ipGroupAddress*

Description: An IP Multicast frame that matched an enabled IP Multicast VLAN was received on the indicated port and interface to the indicated IP Multicast group, from the indicated MAC address.

Action: None.

VLAN.029

Level: C-TRACE

Short Syntax: VLAN.029 OSPF Hello prt *port_num* ifc *ifc_num* MAC *sourceMac[0]*

Long Syntax: VLAN.029 Received an OSPF Hello on port *port_num* interface *ifc_num* MAC *sourceMac[0]*

Description: An OSPF Hello frame was received on the indicated port and interface from the indicated MAC address.

Action: None.

VLAN.030

Level: C-TRACE

Short Syntax: VLAN.030 DVMRP Probe prt *port_num* ifc *ifc_num* MAC *sourceMac[0]*

Long Syntax: VLAN.030 Received a DVMRP Probe on port *port_num* interface *ifc_num* MAC *sourceMac[0]*

Description: An DVMRP Probe frame was received on the indicated port and interface from the indicated MAC address.

Action: None.

VLAN.031

Level: UI-ERROR

Short Syntax: VLAN.031 Invld brid *brid*

Long Syntax: VLAN.031 Invalid bridge ID *brid*

Description: The bridge ID does not exit in the VLAN bridge table.

Action: None.

VLAN.032

Level: C-INFO

Short Syntax: VLAN.032 BcstFwd Info brid *brid* bcsts_fwd *brInstTable->bcasts_forward* crrnt_bcsts_fwd *brInstTable->current_bcasts_forward*

Long Syntax: VLAN.032 Broadcast forwarding state requested bridge ID *brid* brInstTable->bcasts_forward *brInstTable->bcasts_forward* brInstTable->current_bcasts_forward *brInstTable->current_bcasts_forward*

Description: The current state of the broadcast forwarding information was requested for the indicated bridge ID. The state of the broadcasts forward flag is displayed, along with the last state of the current forwarding flag.

Action: None.

VLAN.033

Level: UI-ERROR

Short Syntax: VLAN.033 fddl_brdg_intCnfgStReq state *portState*

Long Syntax: VLAN.033 fddl_bridge_interfaceConfigSetRequest state *portState*

Description: On a call to fddl_bridge_interfaceConfigSetRequest, the state passed in is not SRT_PBLOCKED, SRT_PLISTENING, SRT_PLEARNING or SRT_PFORWARDING.

Action: None.

VLAN.034

Level: C-INFO

Short Syntax: VLAN.034 CfgReq brghndl *brgFddlHandle* ifc *cfgInfo.interfaceNum* cfg.state *cfgInfo.interfaceState* cfg.bcf *cfgInfo.broadcastsForwardedByCpu*

Long Syntax: VLAN.034 Sending CFG interface bridge handle *brgFddlHandle* number *cfgInfo.interfaceNum* interface state *cfgInfo.interfaceState* broadcasts forwarded *cfgInfo.broadcastsForwardedByCpu*

Description: Sending an interface configuration set request for the indicated interface. The interface fddl bridge handle, interface number, broadcast state and and broadcast forwarding flag are displayed.

Action: None.

VLAN.035

Level: UI-ERROR

Short Syntax: VLAN.035 Err fddl_brdg_intCnfgStReq rc *rc*

Long Syntax: VLAN.035 Error calling fddl_bridge_interfaceConfigSetRequest return code *rc*

Description: An error was returned on calling fddl_bridge_interfaceConfigSetRequest. The return code from the call is displayed.

Action: None.

VLAN.036

Level: C-INFO

Short Syntax: VLAN.036 mcastAddReq brghndl *brgFddlHandle* MAC *&mcAddReq.lanDest.macAddr.octets[0]* fwd msk *mcAddReq.iMask[0-7] &mcAddReq.eMask[0-7]* exc msk

Long Syntax: VLAN.036 Sending multicast add request bridge handle *brgFddlHandle* MAC address *&mcAddReq.lanDest.macAddr.octets[0]* include mask *mcAddReq.iMask[0-7] &mcAddReq.eMask[0-7]* exclude map

Description: Sending a fddl_bridge_multicastAddRequest. The fddl bridge handle, MAC address include and exclude mask are displayed.

Action: None.

VLAN.037

Level: UI-ERROR

Short Syntax: VLAN.037 Err fddl_brdg_mcastAddReq *rc*

Long Syntax: VLAN.037 Error calling fddl_bridge_interfaceConfigSetRequest return code *rc*

Description: An error was returned on calling fddl_bridge_multicastAddRequest. The return code from the call is displayed.

Action: None.

VLAN.038

Level: C-INFO

Short Syntax: VLAN.038 macAddrDelReq brghndl *brgFddlHandle* MAC *mcAddReq.lanDest.macAddr.octets[0]*

Long Syntax: VLAN.038 Sending mac address delete request bridge handle *brgFddlHandle* MAC address *mcAddReq.lanDest.macAddr.octets[0]*

Description: Sending a fddl_bridge_macAddreDelteRequest. The fddl bridge handle and MAC address are displayed.

Action: None.

VLAN.039

Level: UI-ERROR

Short Syntax: VLAN.039 Err fddl_brdg_macAddrDelReq *rc*

Long Syntax: VLAN.039 Error calling fddl_bridge_macAddrDeleteRequest return code *rc*

Description: An error was returned on calling fddl_bridge_macAddrDeleteRequest. The return code from the call is displayed.

Action: None.

VLAN.040

Level: C-INFO

Short Syntax: VLAN.040 mcastStatsGetReq brghndl *brgFddlHandle* MAC *&mcAddReq.lanDest.macAddr.octets[0]*

Long Syntax: VLAN.040 Sending multicast get statistics request bridge handle *brgFddlHandle* MAC address *&mcAddReq.lanDest.macAddr.octets[0]*

Description: Sending a fddl_bridge_multicastStatsGetRequest. The fddl bridge handle and MAC address are displayed.

Action: None.

VLAN.041

Level: UI-ERROR

Short Syntax: VLAN.041 Err fddl_brdg_mcastStatsGetReq *rc*

Long Syntax: VLAN.041 Error calling fddl_bridge_multicastStatsGetRequest return code *rc*

Description: An error was returned on calling fddl_bridge_multicastStatsGetRequest. The return code from the call is displayed.

Action: None.

VLAN.042

Level: C-INFO

Short Syntax: VLAN.042 mcastStatsGetReply MAC *mcAddReq.lanDest.macAddr.octets[0]* fwd *stats.framesForwarded* fltrd *stats.framesFiltered*

Long Syntax: VLAN.042 Received multicast statistics reply MAC address *mcAddReq.lanDest.macAddr.octets[0]* frames forwarded *stats.framesForwarded* frames filtered *stats.framesFiltered*

Description: Received fddl_bridge_multicastStatsGetReply. The MAC address,

frames forwarded and frames filtered values are displayed.

Action: None.

VLAN.043

Level: UI-ERROR

Short Syntax: VLAN.043 mcastStatsGetReply address not found MAC *lanDest.macAddr.octets[0]*

Long Syntax: VLAN.043 Received multicast statistics reply MAC address *lanDest.macAddr.octets[0]* not found

Description: Received a multicast statistics reply but the MAC address was not found.

Action: None.

VLAN.044

Level: C-INFO

Short Syntax: VLAN.044 IGMP Query in prt *port_num* ifc *ifc_num* MAC *sourceMac[0]* Group *ipGroupAddress*

Long Syntax: VLAN.044 Received an IGMP Query on port *port_num* interface *ifc_num* MAC *sourceMac[0]* Group *ipGroupAddress*

Description: An IGMP Query frame was received on the indicated port and interface for the indicated IP Multicast group, from the indicated MAC address. If the group is zero, the frame is a general Query to all groups.

Action: None.

VLAN.045

Level: C-INFO

Short Syntax: VLAN.045 Tx IGMP General Query out port *port_num* ifc *ifc_num*

Long Syntax: VLAN.045 Transmit IGMP Query out port *port_num* interface *ifc_num*

Description: An IGMP General Query frame was transmitted out the indicated port and interface.

Action: None.

VLAN.046

Level: C-INFO

Short Syntax: VLAN.046 Tx IGMP Report prt *port_num* ifc *ifc_num* srcIp *ipSourceAddress* Group *ipGroupAddress*

Long Syntax: VLAN.046 Transmit an IGMP Report out port *port_num* interface *ifc_num* srcIp *ipSourceAddress* Group *ipGroupAddress*

Description: An IGMP Report frame was transmitted out the indicated port and interface for the indicated IP Multicast group, with the indicated source IP address.

Action: None.

VLAN.047

Level: C-INFO

Short Syntax: VLAN.047 BOOTP received on prt *port_num* ifc *ifc_num* from MAC *sourceMac[0]*

Long Syntax: VLAN.047 Received a BOOTP frame on port *port_num* interface *ifc_num* from MAC *sourceMac[0]*

Description: A BOOTP frame was received on the indicated port and interface from the indicated MAC address.

Action: None.

Chapter 125. Virtual Router Redundancy Protocol (VRRP)

This chapter describes Virtual Router Redundancy Protocol (VRRP) messages. For information on message content and how to use the message, refer to the Introduction.

VRRP.001

Level: C-INFO

Short Syntax: VRRP.001 VRID *Interface_address/ vrid* init success net *network_number* ifc *network_name_number/*

Long Syntax: VRRP.001 VRID *Interface_address/ vrid* initialization successful for net *network_number* and interface *network_name_number/* .

Description: A VRID (Virtual Router) was successfully initialized. This VRID will participate in the VRRP protocol.

VRRP.002

Level: UE-ERROR

Short Syntax: VRRP.002 VRID *Interface_address/ vrid* init failed: *reason_code*

Long Syntax: VRRP.002 VRID *Interface_address/ vrid* initialization failed due to *reason_code*.

Description: A VRID (Virtual Router) was not initialized. The reason code indicates the type of failure: 1 - Interface IP address not found. 2 - Net for IP address not found. 3 - Net for IP address not supported. 4 - Unsupported token ring functional address. 5 - Memory error allocating VRID control block. 6 - VRID would not have any virtual addresses. 7 - Multicast VRID on Bridge net not allowed.

VRRP.003

Level: C-INFO

Short Syntax: VRRP.003 Net *network_number* ifc *network_name_number/ source_mac destination_mac-> protocol* proto

Long Syntax: VRRP.003 Net *network_number* interface *network_name_number/ source_mac* MAC level send *destination_mac-> protocol* protocol successful.

Description: The MAC level frame was sent on the interface using the network n_fsend function.

VRRP.004

Level: UE-ERROR

Short Syntax: VRRP.004 Net *network_number* ifc *network_name_number/ source_mac destination_mac-> protocol* proto *reason_code* failed:

Long Syntax: VRRP.004 Net *network_number* interface *network_name_number/ source_mac* MAC level send *destination_mac-> protocol* protocol *reason_code* failed due to .

Description: The MAC level frame send failed. The reason indicates the origin of the failure. Others - n_fsend() return code 254 - Link layer header allocation failure.

VRRP.005

Level: C-INFO

Short Syntax: VRRP.005 VRID *Interface_address/ vrid* adv net *interface_name* ifc *network_number/ network_name_number*

Long Syntax: VRRP.005 VRID *Interface_address/ vrid* advertisement sent on net *interface_name* interface *network_number/ network_name_number*.

Description: The VRID advertisement was sent on the interface.

VRRP.006

Level: UE-ERROR

Short Syntax: VRRP.006 VRID *Interface_address/ vrid* adv net *interface_name* ifc *network_number/ network_name_name* failed:

Long Syntax: VRRP.006 VRID *Interface_address/ vrid* advertisement on net *interface_name* interface *network_number/ network_name_name* failed due to reason .

Description: The VRID advertisement was not sent on the interface due to reason: 1 - I/O buffer allocation failure 2 - vrrp_mac_send() failure.

VRRP.007

Level: C-INFO

Short Syntax: VRRP.007 VRID *Interface_address/ vrid* state *old_state_name-> new_state_name*: *event_name*

Long Syntax: VRRP.007 VRID *Interface_address/ vrid* state change from *old_state_name* to *new_state_name* due to event *event_name*.

Description: The VRID went through a state transistion.

VRRP.008

Level: C-INFO

Short Syntax: VRRP.008 VRID *Interface_address/ vrid* adv rcv src *source_ip* net *interface_name* ifc *network_number/ network_name_number*

Long Syntax: VRRP.008 VRID *Interface_address/ vrid* advertisement received on from *source_ip* on net *interface_name* interface *network_number/ network_name_number*.

Description: The VRID advertisement was received from the sender on the interface.

VRRP.009

Level: UE-ERROR

Short Syntax: VRRP.009 VRID *Interface_address/ vrid* adv rej src *source_ip* net *interface_name* ifc *network_number/ network_name_number*: *reason_code*

Long Syntax: VRRP.009 VRID *Interface_address/ vrid* advertisement received on from *source_ip* on net *interface_name* interface *network_number/ network_name_number* due to reason *reason_code*.

Description: The VRID advertisement was rejected due to the specified reason. Reason codes include: 1 - Bad IP TTL 2 - Bad IP Length 3 - Bad VRRP Version/Type 4 - Received on wrong Net 5 - Bad checksum 6 - Authentication error

---

VRRP.010

Level: C-INFO

Short Syntax: VRRP.010 VRID *Interface_address/ vrid* adv src *source_ip* net *vrid_interval* ifc *received_interval/* int mm vs

Long Syntax: VRRP.010 VRID *Interface_address/ vrid* advertisement received from *source_ip* net *vrid_interval* interface *received_interval/* has interval mismatch versus .

Description: The received VRID advertisement had an advertisement interval different than the configured VRID advertisement. Nevertheless, it was accepted.

VRRP.011

Level: C-INFO

Short Syntax: VRRP.011 VRID *Interface_address/ vrid* adv src *source_ip* net ifc / addr mm

Long Syntax: VRRP.011 VRID *Interface_address/ vrid* advertisement received on from *source_ip* net interface / had address list mismatch.

Description: The received VRID advertisement had an address list different from the configured VRID address list. Nevertheless, it was accepted.

VRRP.012

Level: UI-ERROR

Short Syntax: VRRP.012 Dup VR addr *ip_address* in VRIDs *interface_address/ vrid* and *interface_address/ vrid*

Long Syntax: VRRP.012 Duplicate Virtual Router IP address *ip_address* in VRIDs *interface_address/ vrid* and *interface_address/ vrid*.

Description: The specified Virtual Router IP address is configured in multiple VRIDs or multiple times in a single VRID.

Action: Delete all but one instance of the specified Virtual Router IP address.

Chapter 126. Voice Feature (VOICE)

This chapter describes Voice Feature (VOICE) messages. For information on message content and how to use the message, refer to the Introduction.

VOIC.001

Level: C-INFO

Short Syntax: VOIC.001 Call *message*7s recv nt *network ID*, trans nt *network ID* DLCI *dlci* sub *subchannel* cir *circuit* NID *nid infolabel info*

Long Syntax: VOIC.001 Call *message*7s packet received on network *network ID*, transmitted on network *network ID* DLCI = *dlci* subchannel = *subchannel* circuit = *circuit* Sender NID = *nid infolabel = info*

Description: A Call Processing Packet has been received and transmitted on the indicated interfaces.

VOIC.002

Level: C-INFO

Short Syntax: VOIC.002 Call *message*7s trans nt *network ID* DLCI *dlci* sub *subchannel* cir *circuit* NID *nid infolabel info*

Long Syntax: VOIC.002 Call *message*7s packet transmitted on network *network ID* DLCI = *dlci* subchannel = *subchannel* circuit = *circuit* Sender NID = *nid infolabel = info*

Description: A Locally Generated Call Processing Packet has been transmitted on the indicated interface.

VOIC.003

Level: C-INFO

Short Syntax: VOIC.003 Call *message*7s recv nt *network ID*, disc unroutable DLCI *dlci* sub *subchannel* cir *circuit* NID *nid infolabel info*

Long Syntax: VOIC.003 Call *message*7s packet received on network *network ID*, discarded as unroutable DLCI = *dlci* subchannel = *subchannel* circuit = *circuit* Sender NID = *nid infolabel = info*

Description: A Call Processing Packet has been received on the indicated interface and discarded as unroutable.

VOIC.004

Level: ALWAYS

Short Syntax: VOIC.004 Cfg Warning: *warning* nt *network ID*

Long Syntax: VOIC.004 Configuration Warning: *warning* network *network ID*

Description: A possible configuration error affecting voice operation has been detected for the specified interface.

Chapter 127. V.25bis Dialing (V25B)

This chapter describes V.25bis Dialing (V25B) messages. For information on message content and how to use the message, refer to the Introduction.

V25B.001

Level: CE-ERROR

Short Syntax: V25B.001 I_ERR (0x *status*) len(*msglen*) on rcv nt *network ID*

Long Syntax: V25B.001 Frame received with I_ERR set (status = 0x *status*) or bad length(*msglen*), on network *network ID*

Description: V.25bis: v25b_rx() received a buffer from the driver with the error flag set or with a length less than the minimum.

Action: Report this event to customer service.

V25B.002

Level: UE-ERROR

Short Syntax: V25B.002 Rx bad type (*type*) st *state* on nt *network ID*

Long Syntax: V25B.002 Received an unrecognized frame type (*type*) in state *state*, on network *network ID*

Description: V.25bis: v25b_rx() received a frame from the DCE other than a normal V.25bis indication in a state other than ″connected″.

Action: Report this event to customer service.

V25B.003

Level: U-INFO

Short Syntax: V25B.003 Cll to *address* failed T = *secs*. *ms* secs on nt *network ID*

Long Syntax: V25B.003 Call to *address* failed after *secs*. *ms* seconds on network *network ID*

Description: A connection attempt failed. Ref V25B.016 for posssible reasons.

V25B.006

Level: C-INFO

Short Syntax: V25B.006 FSM st *state1* ev *event* -> st *state2* nt *network ID*

Long Syntax: V25B.006 FSM transition occurred: old state *state1*, event *event*, new state *state2* on network *network ID*.

Description: The handler received an event which triggered a state change. If this occurred as a result of a modem signal change, the preceding log message (if enabled) should indicate the new signals.

V25B.007

Level: C-INFO

Short Syntax: V25B.007 Mdm Chg 0x *modem1* -> 0x *modem2* (DSR/CTS/CD/CI) nt *network ID*

Long Syntax: V25B.007 A modem signal change was detected (0x *modem1* -> 0x *modem2* DSR/CTS/CD/CI) network *network ID*.

Description: A change in the modem signals from the DCE was detected; this may or may not precipitate an FSM transition (follows).

V25B.008

Level: UE-ERROR

Short Syntax: V25B.008 Dead DCE nt *network ID*

Long Syntax: V25B.008 DCE not responding to the handler on network *network ID*.

Description: The V.25bis handler attempts to raise the modem (or CU/DSU) on self-test. If it doesn't respond (by raising CTS), the handler assumes it is dead or non-compliant.

Cause: DCE not connected, powered-off, inoperable, or non-V.25bis compliant.

Action: Attach the cable, turn it on, fix it, or get a compliant one.

V25B.009

Level: P-TRACE

Short Syntax: V25B.009 RxD Pkt ln *msglen* nt *network ID*

Long Syntax: V25B.009 Received a frame of length (*msglen*) from network *network ID*.

Description: The V.25bis handler received a data frame, which it is forwarding to its client encapsulator.

V25B.010

Level: P-TRACE

Short Syntax: V25B.010 TxD Pkt ln *msglen* nt *network ID*

Long Syntax: V25B.010 Transmitted a frame of length (*msglen*) over network *network ID*.

Description: The V.25bis handler has transmitted a data frame on behalf of its client encapsulator.

V25B.012

Level: UE-ERROR

Short Syntax: V25B.012 No heap on *function* nt *network ID*

Long Syntax: V25B.012 Insufficient heap memory to support this function (*function*) on network *network ID*.

Description: The V.25bis handler requires a certain amount of heap memory to operate, and it couldn't get it.

Cause: Either the load image, or the protocol tables are too large.

Action: Get a smaller load image, or reduce the size of the forwarder tables.

V25B.013

Level: UE-ERROR

Short Syntax: V25B.013 Bd cfg (*function*) nt *network ID*

Long Syntax: V25B.013 Incomplete configuration (*function*) for network *network ID*.

Description: The V.25bis handler requires a minimal configuration to work, and that information was not specified.

Action: Verify that the V25B configuration for this interface includes at least the Local Address.

V25B.014

Level: UE-ERROR

Short Syntax: V25B.014 Bd ConnID (0x *ConnID* 0x *RegP* 0x *PortP*)

Long Syntax: V25B.014 V.25bis function invoked with an invalid Connection Identifier (0x *ConnID* 0x *RegP* 0x *PortP*).

Description: The V.25bis handler interfaces to the encapsulators via a Connection Identifier for its connection-related functions. It has been invoked with an invalid Connection Identifier.

V25B.015

Level: U-TRACE

Short Syntax: V25B.015 Drp RxD Pkt ln *msglen* st *state* nt *network ID*

Long Syntax: V25B.015 Dropping a received Data frame of length (*msglen*) in state *state* from network *network ID*.

Description: The V.25bis handler received a data frame, in a state where it doesn't expect one, so it dropped it.

V25B.016

Level: U-TRACE

Short Syntax: V25B.016 *indtype* Ind rsn *reason* st *state* nt *network ID*

Long Syntax: V25B.016 DCE indication *indtype*, reason *reason* in state *state* on network *network ID*.

Description: The DCE has sent the specified indication. This may indicate that a connect attempt, initiated by the V.25bis handler has failed (INV or CFI) for the reason specified (see the calling unit user's manual for a description of the reason code, if any accompanies this message). Alternatively, this may just be a redundant incoming call indicatation (INC), which had already been signalled by the CI Circuit 125.

Cause: Call aborted: router timed out, or modem user interface command.

Action: Extend the call establishment period or don't interrupt the call.

Cause: Local DCE Busy: the user interfered through the calling unit user interface.

Action: Do not interfere.

Cause: Engaged Tone: the remote end is busy.

Action: Try again later (the router should automatically).

Cause: No Dial tone: the telephone network isn't responding.

Action: Fix the link, contact service provider.

Cause: Number not stored.

Action: Call customer service: we don't use the corresponding command.

Cause: No Answer Tone detected: remote unit did not respond with answer tone.

Action: Check called number, verify that remote unit is on-line.

Cause: Ring Tone (but no answer).

Action: Check called number, verify that remote unit is on-line.

V25B.017

Level: C-INFO

Short Syntax: V25B.017 Indctn *Message* st *state* nt *network ID*

Long Syntax: V25B.017 DCE sent *Message* in state *state*, on network *network ID*.

Description: The calling unit has either accepted the router's request (INC), or is connecting the call (CNX or ONL). This is a normal event -- albeit perhaps not always reported by a given DCE/CU.

V25B.018

Level: UE-ERROR

Short Syntax: V25B.018 Dlyd Cll ind *delaytime* minutes nt *network ID*

Long Syntax: V25B.018 DCE indicates Call Delayed for *delaytime* minutes on network *network ID*.

Description: The calling unit (DCE) has indicated that it will not attempt additional outgoing calls for at least the indicated period. This is an optional feature of some DCEs in some administrations, which inhibits high frequencies of calls over a short period. Examine the previous log entries to determine why so many calls are being made.

Cause: Connections to a particular destination(s) are continually being cleared.

Action: Check the GateWay messages, to determine if the calls are being IDLE-d out (increase the idle period), or if the verification procedure is failing (check the calling number at both ends).

Cause: Non-responding remote DCE.

Action: Check the called number and verify that the remote DCE is on-line.

Cause: Busy remote.

Action: Increase the Call Retries timeout for that destination.

V25B.019

Level: UE-ERROR

Short Syntax: V25B.019 No Bf Cll nt *network ID*

Long Syntax: V25B.019 Buffer unavailable for connection request on network *network ID*.

Description: The handler needs a buffer to send the "connection request" to the DCE, and couldn't obtain one. The call fails. The router should re-initiate the call at a later time.

V25B.020

Level: UE-ERROR

Short Syntax: V25B.020 Bd Sts CRN Tx 0x *status* nt *network ID*

Long Syntax: V25B.020 Bad transmit status (0x *status*) for CRN network *network ID*.

Description: The driver reports a bad transmit status when trying to send the Call Request (CRN).

V25B.021

Level: C-INFO

Short Syntax: V25B.021 Set DSS *DSS* nt *network ID*

Long Syntax: V25B.021 Set output signals *DSS* on network *network ID*

Description: The router is changing its output dataset signals in response to the preceding event. (DTR = V.24 Circuit 108/2 and RTS = V.24 Circuit 105)

V25B.022

Level: CI-ERROR

Short Syntax: V25B.022 no bfr avl *action* nt *network ID*

Long Syntax: V25B.022 no buffer available for *action* network *network ID*

Description: A packet buffer was not available when the hardware-sppecific interface code required one to perform the specified action.

V25B.023

Level: U-INFO

Short Syntax: V25B.023 Slftst OK nt *network ID*

Long Syntax: V25B.023 Selftest completed successfully on network *network ID*

Description: Self-test of the connection between the router and the modem completed ok.

V25B.024

Level: C-INFO

Short Syntax: V25B.024 Tx CRN *destination* nt *network ID*

Long Syntax: V25B.024 Sending Dial (CRN) command for call to *destination* on network *network ID*

Description: The modem is in a now in a state where it can actually receive V.25bis commands, so we are sending it the telephone number to dial.

V25B.025

Level: C-INFO

Short Syntax: V25B.025 Clnt CR *destination* nt *network ID*

Long Syntax: V25B.025 Client connection request to *destination* on network *network ID*

Description: The client (ex: Dial Circuit or WAN

Restoral) has made a connection request to the specified address.

V25B.026

Level: C-INFO

Short Syntax: V25B.026 Clnt CR blckd *destination* nt *network ID*

Long Syntax: V25B.026 Client connection request on busy interface to *destination* on network *network ID*

Description: The client (ex: Dial Circuit or Wan Restoral) is trying to initiate a connection, but the base network is busy.

V25B.027

Level: C-INFO

Short Syntax: V25B.027 Out Call *destination* cmp T= *time* nt *network ID*

Long Syntax: V25B.027 Client connection established to *destination* in *time* seconds on network *network ID*

Description: In the specified time, the router established the connection requested (ex: Dial Circuit or Wan Restoral). The operator may care to use this value to adjust the configured connect timeout.

V25B.028

Level: ALWAYS

Short Syntax: V25B.028 Bad drct Tx prot *Protocol*, pls remap to dial circuit on nt *network ID*

Long Syntax: V25B.028 Some forwarder (*Protocol*) has attempted to transmit directly over the V.25bis network *network ID*

Description: Transmits over the V.25bis network are only supposed to be done via an associated dial circuit, which will do an appropriate encapsulation. This is caused by a mistake in the configuration of the forwarders. No forwarder should be configured to use the V.25bis network. To bound the number of these messages, they will be logged only a fraction of the actual events.

Cause: A forwarder (IP, IPX, etc) address was assigned to the V.25bis interface.

Action: Delete the address, and (probably) re-assign it to a dial circuit (which is itself mapped to the V.25bis network).

Cause: The bridge or other forwarder has been configured to use the V.25bis interface.

Action: Remove the V.25bis interface as a port used by the bridge or forwarder.

Chapter 128. V.34 Dialing (V34)

This chapter describes V.34 Dialing (V34) messages. For information on message content and how to use the message, refer to the Introduction.

V34.001

Level: CE-ERROR

Short Syntax: V34.001 I_ERR (0x *status*) len(*msglen*) on rcv nt *network ID*

Long Syntax: V34.001 Frame received with I_ERR set (status = 0x *status*) or bad length(*msglen*), on network *network ID*

Description: V.34: V34_rx() received a buffer from the driver with the error flag set or with a length less than the minimum.

Action: Report this event to customer service.

V34.002

Level: C-INFO

Short Syntax: V34.002 Rcv data/response: *type* st *state* on nt *network ID*

Long Syntax: V34.002 Received data from modem (*type*) in state *state*, on network *network ID*

Description: V34 received data from the modem while in a state other than "connected". This is normal, and ususally the modem responding to AT commands sent by the router.

V34.003

Level: U-INFO

Short Syntax: V34.003 Cll to *address* failed T = *secs*. *ms* secs on nt *network ID*

Long Syntax: V34.003 Call to *address* failed after *secs*. *ms* seconds on network *network ID*

Description: A connection attempt failed. Ref V34.016 for posssible reasons.

V34.006

Level: C-INFO

Short Syntax: V34.006 FSM st *state1* ev *event* -> st *state2* nt *network ID*

Long Syntax: V34.006 FSM transition occurred: old state *state1*, event *event*, new state *state2* on network *network ID*.

Description: The handler received an event which triggered a state change. If this occurred as a result of a modem signal change, the preceding log message (if enabled) should indicate the new signals.

V34.007

Level: C-INFO

Short Syntax: V34.007 Mdm Chg 0x *modem1* -> 0x *modem2* (DSR/CTS/CD/CI) nt *network ID*

Long Syntax: V34.007 A modem signal change was detected (0x *modem1* -> 0x *modem2* DSR/CTS/CD/CI) network *network ID*.

Description: A change in the modem signals from the DCE was detected; this may or may not precipitate an FSM transition (follows).

V34.008

Level: UE-ERROR

Short Syntax: V34.008 Dead DCE st *state* nt *network ID*

Long Syntax: V34.008 DCE not responding (current st *state*) to the handler on network *network ID*.

Description: The V.34 handler attempts to raise the modem on self-test or per normal operation. If it doesn't respond (by raising CTS), the handler assumes it is dead or non-compliant.

Cause: DCE not connected, powered-off, inoperable, or non-V.34 compliant.

Action: Attach the cable, turn it on, fix it, or get a compliant one.

V34.009

Level: P-TRACE

Short Syntax: V34.009 RxD Pkt ln *msglen* nt *network ID*

Long Syntax: V34.009 Received a frame of length (*msglen*) from network *network ID*.

Description: The V.34 handler received a data frame, which it is forwarding to its client encapsulator.

V34.010

Level: P-TRACE

Short Syntax: V34.010 TxD Pkt ln *msglen* nt *network ID*

Long Syntax: V34.010 Transmitted a frame of length (*msglen*) over network *network ID*.

Description: The V.34 handler has transmitted a data frame on behalf of its client encapsulator.

V34.012

Level: UE-ERROR

Short Syntax: V34.012 No heap on *function* nt *network ID*

Long Syntax: V34.012 Insufficient heap memory to support this function (*function*) on network *network ID*.

Description: The V.34 handler requires a certain amount of heap memory to operate, and it couldn't get it.

Cause: Either the load image, or the protocol tables are too large.

Action: Get a smaller load image, or reduce the size of the forwarder tables.

V34.013

Level: UE-ERROR

Short Syntax: V34.013 Bd cfg (*function*) nt *network ID*

Long Syntax: V34.013 Incomplete configuration (*function*) for network *network ID*.

Description: The V.34 handler requires a minimal configuration to work, and that information was not specified.

Action: Verify that the V34 configuration for this interface includes at least the Local Address.

V34.014

Level: UE-ERROR

Short Syntax: V34.014 Bd ConnID (0x *ConnID*)

Long Syntax: V34.014 V.34 function invoked with an invalid Connection Identifier (0x *ConnID*).

Description: The V.34 handler interfaces to the encapsulators via a Connection Identifier for its connection-related functions. It has been invoked with an invalid Connection Identifier.

V34.015

Level: U-TRACE

Short Syntax: V34.015 Drp RxD Pkt ln *msglen* st *state* nt *network ID*

Long Syntax: V34.015 Dropping a received Data frame of length (*msglen*) in state *state* from network *network ID*.

Description: The V.34 handler received a data frame, in a state where it doesn't expect one, so it dropped it.

V34.016

Level: U-TRACE

Short Syntax: V34.016 *indtype* Ind rsn *reason* st *state* nt *network ID*

Long Syntax: V34.016 DCE indication *indtype*, reason *reason* in state *state* on network *network ID*.

Description: The DCE has sent the specified indication. This may indicate that a connect attempt, initiated by the V.34 handler has failed (INV or CFI) for the reason specified (see the calling unit user's manual for a description of the reason code, if any accompanies this message). Alternatively, this may just be a redundant incoming call indicatation (INC), which had already been signalled by the CI Circuit 125.

Cause: Call aborted: router timed out, or modem user interface command.

Action: Extend the call establishment period or don't interrupt the call.

Cause: Local DCE Busy: the user interfered through the calling unit user interface.

Action: Do not interfere.

Cause: Engaged Tone: the remote end is busy.

Action: Try again later (the router should automatically).

Cause: No Dial tone: the telephone network isn't responding.

Action: Fix the link, contact service provider.

Cause: Number not stored.

Action: Call customer service: we don't use the corresponding command.

Cause: No Answer Tone detected: remote unit did not respond with answer tone.

Action: Check called number, verify that remote unit is on-line.

Cause: Ring Tone (but no answer).

Action: Check called number, verify that remote unit is on-line.

V34.017

Level: C-INFO

Short Syntax: V34.017 Indctn *Message* st *state* nt *network ID*

Long Syntax: V34.017 DCE sent *Message* in state *state*, on network *network ID*.

Description: The calling unit has either accepted the router's request , or is connecting the call. This is a normal event -- albeit perhaps not always reported by a given DCE.

V34.018

Level: UE-ERROR

Short Syntax: V34.018 Dlyd Cll ind *delaytime* minutes nt *network ID*

Long Syntax: V34.018 DCE indicates Call Delayed for *delaytime* minutes on network *network ID*.

Description: The calling unit (DCE) has indicated that it will not attempt additional outgoing calls for at least the indicated period. This is an optional feature of some DCEs in some administrations, which inhibits high frequencies of calls over a short period. Examine the previous log entries to determine why so many calls are being made.

Cause: Connections to a particular destination(s) are continually being cleared.

Action: Check the GateWay messages, to determine if the calls are being IDLE-d out (increase the idle period), or if the verification procedure is failing (check the calling number at both ends).

Cause: Non-responding remote DCE.

Action: Check the called number and verify that the remote DCE is on-line.

Cause: Busy remote.

Action: Increase the Call Retries timeout for that destination.

V34.019

Level: UE-ERROR

Short Syntax: V34.019 No Bf Cll nt *network ID*

Long Syntax: V34.019 Buffer unavailable for connection request on network *network ID*.

Description: The handler needs a buffer to send the "connection request" to the DCE, and couldn't obtain one. The call fails. The router should re-initiate the call at a later time.

V34.020

Level: UE-ERROR

Short Syntax: V34.020 Bd Sts CRN Tx 0x *status* nt *network ID*

Long Syntax: V34.020 Bad transmit status (0x *status*) for CRN network *network ID*.

Description: The driver reports a bad transmit status when trying to send the Call Request (CRN).

V34.021

Level: C-INFO

Short Syntax: V34.021 Set DSS *DSS* nt *network ID*

Long Syntax: V34.021 Set output signals *DSS* on network *network ID*

Description: The router is changing its output dataset signals in response to the preceding event. (DTR = V.24 Circuit 108/2 and RTS = V.24 Circuit 105)

V34.022

Level: CI-ERROR

Short Syntax: V34.022 no bfr avl *action* nt *network ID*

Long Syntax: V34.022 no buffer available for *action* network *network ID*

Description: A packet buffer was not available when the hardware-sppecific interface code required one to perform the specified action.

V34.023

Level: U-INFO

Short Syntax: V34.023 Slftst OK nt *network ID*

Long Syntax: V34.023 Selftest completed successfully on network *network ID*

Description: Self-test of the connection between the router and the modem completed ok.

V34.024

Level: C-INFO

Short Syntax: V34.024 Tx CRN *destination* nt *network ID*

Long Syntax: V34.024 Sending Dial command for call to *destination* on network *network ID*

Description: The modem is in a now in a state where it can actually receive V.34 commands, so we are sending it the telephone number to dial.

V34.025

Level: C-INFO

Short Syntax: V34.025 Clnt CR *destination* nt *network ID*

Long Syntax: V34.025 Client connection request to *destination* on network *network ID*

Description: The client (ex: Dial Circuit or WAN Restoral) has made a connection request to the specified address.

V34.026

Level: C-INFO

Short Syntax: V34.026 Clnt CR blckd *destination* nt *network ID*

Long Syntax: V34.026 Client connection request on busy interface to *destination* on network *network ID*

Description: The client (ex: Dial Circuit or Wan Restoral) is trying to initiate a connection, but the base network is busy.

V34.027

Level: C-INFO

Short Syntax: V34.027 Out Call *destination* cmp T= *time* nt *network ID*

Long Syntax: V34.027 Client connection established to *destination* in *time* seconds on network *network ID*

Description: The connection requested by a local client (ex: Dial Circuit or Wan Restoral) to the specified address has been established in the specified time. The operator may care to use this value to adjust configured connect timeout.

V34.028

Level: ALWAYS

Short Syntax: V34.028 Bad drct Tx prot *Protocol*, pls remap to dial circuit on nt *network ID*

Long Syntax: V34.028 Some forwarder (*Protocol*) has attempted to transmit directly over the V.34 network *network ID*

Description: Transmits over the V.34 network are only supposed to be done via an associated dial circuit, which will do an appropriate encapsulation. This is caused by a mistake in the configuration of the forwarders. No forwarder should be configured to use the V.34 network. To bound the number of these messages, they will be logged only a fraction of the actual events.

Cause: A forwarder (IP, IPX, etc) address was assigned to the V.34 interface.

Action: Delete the address, and (probably) re-assign it to a dial circuit (which is itself mapped to the V.34 network).

Cause: The bridge or other forwarder has been configured to use the V.34 interface.

Action: Remove the V.34 interface as a port used by the bridge or forwarder.

V34.029

Level: UE_ERROR

Short Syntax: V34.029 V34 escape and hangup command, *hangup_string*, not recognized by modem on nt *netnum*

Long Syntax: V34.029 V34 escape and hangup command, *hangup_string*, not recognized by modem on nt *netnum*

Description: The modem did not recognize the escape sequence and hangup command sent by the V34 intitialization fsm.

Cause: An incorrect hangup string has been configured.

Action: Look up the correct hangup string for the particular modem connected to the interface. The default if none is specified is ATH.

V34.030

Level: UE_ERROR

Short Syntax: V34.030 V34 reset command, *reset_string*, not recognized by modem on nt *netnum*

Long Syntax: V34.030 V34 reset command, *reset_string*, not recognized by modem on nt *netnum*

Description: The modem did not recognize the reset command sent by the V34 intitialization fsm.

Cause: An incorrect reset string has been configured.

Action: Look up the correct reset string for the particular modem connected to the interface. The default if none is specified is ATZ.

V34.031

Level: UE_ERROR

Short Syntax: V34.031 V34 factory defaults command, *factory_string*, not recognized by modem on nt *netnum*

Long Syntax: V34.031 V34 factory defaults command, *factory_string*, not recognized by modem on nt *netnum*

Description: The modem did not recognize the set factory defaults command sent by the V34 intitialization fsm.

Cause: The modem is not responding as expected to the command to set factory defaults.

Action: The error is ignored. If the modem does not operate correctly you may need to change the default configured initialization string.

V34.032

Level: UE_ERROR

Short Syntax: V34.032 V34 init command, *init_string*, not recognized by modem on nt *netnum*

Long Syntax: V34.032 V34 init command, *init_string*, not recognized by modem on nt *netnum*

Description: The modem did not recognize the init string command sent by the V34 intitialization fsm.

Cause: An incorrect init string has been configured.

Action: Look up the correct init string for the particular modem connected to the interface. The default if none is specified is at&f&s1l1&d2&c1x3.

V34.033

Level: UE_ERROR

Short Syntax: V34.033 V34 initialization failed on nt *network ID*

Long Syntax: V34.033 V34 initialization failed on net *network ID*

Description: The V34 modem initialization algortithm failed.

Cause: DCE failed to raise CTS or did not send OK to ATZ, AT&F or configurable modem init string.

Action: Look up the correct init string for the particular modem connected to the interface. The default if none is specified is at&f&s1l1&d2&c1x3.

V34.034

Level: CE-ERROR

Short Syntax: V34.034 slf tst failed, mdm sts: CTS = *cts*, DSR = *dsr*, DCD = *dcd*, nt *network ID*

Long Syntax: V34.034 Self test failed because of modem status: CTS = *cts*, DSR = *dsr*, DCD = *dcd*, network *network ID*

Description: The interface failed self test because at least one of the modem signals was off. The present state of the modem signals is shown in the ELS message. The normal state of the modem signals is CTS=ON, DSR=ON, and DCD=OFF for V34 connections.

Cause: Cable not connected to modem.

Action: Connect cable.

Cause: Modem not powered up.

Action: Power up modem.

Cause: Modem does not have good connection to other end of line (especially DCD OFF).

Action: Solve modem problem.

V34.035

Level: CE-ERROR

Short Syntax: V34.035 int dwn due to mdm sts: CTS = *cts*, DSR = *dsr*, DCD = *dcd*, nt *network ID*

Long Syntax: V34.035 Interface down because of modem status: CTS = *cts*, DSR = *dsr*, DCD = *dcd*, network *network ID*

Description: The interface was brought down because one of the modem signals was off. The normal state of the modem signals is CTS=ON, DSR=ON, and DCD=ON for V34.

V34.036

Level: C-INFO

Short Syntax: V34.036 Modem status change CTS = *cts*, DSR = *dsr*, DCD = *dcd*, nt *network ID*

Long Syntax: V34.036 Modem status change CTS = *cts*, DSR = *dsr*, DCD = *dcd*, on network *network ID*

Description: A modem status change has occurred. The present state of the modem signals is shown in the ELS message. The normal state of the modem signals is CTS=ON, DSR=ON, and DCD=ON.

V34.037

Level: C-INFO

Short Syntax: V34.037 Cll dscnnct from ISDN cll hndlr nt *network ID*

Long Syntax: V34.037 Call disconnect from ISDN call handler on network *network ID*

Description: The ISDN call handler signalled call termination for the specified network. This may be due to normal call termination, but could signal a hardware failure.

V34.038

Level: U-INFO

Short Syntax: V34.038 No nt to receive call.

Long Syntax: V34.038 No net available or able to receive incomming call.

Description: An incoming call was detected but no net available or configured to receive the call. Check your configuration to be sure there are enough nets defined to receive incoming calls.

V34.039

Level: UI-ERROR

Short Syntax: V34.039 CML call remap to non-existent PPP dial circuit on nt *network ID*

Long Syntax: V34.039 CML tried to remap a call to an unregistered PPP circuit on network *network ID*

Description: CML attempted to remap a call to a dial circuit not currently registered with the network. Be sure the dial circuit(s) on this network interface are configured correctly.

V34.040

Level: C-INFO

Short Syntax: V34.040 Configured call connect timeout of *time* seconds exceeded on nt *network ID*

Long Syntax: V34.040 The NET's configured call connect timeout of *time* seconds was exceeded on net *network ID*

Description: The call connect timeout was exceeded. The operator can increase the configured connect timeout to allow more time for call completion.

V34.041

Level: P-TRACE

Short Syntax: V34.041 Sent modem command: *type* on nt *network ID*

Long Syntax: V34.041 Sent modem command: *type* on net *network ID*

Description: The router is sending a command to the modem.

V34.042

Level: C_INFO

Short Syntax: V34.042 PPP dial circuit disconnect request on nt *network ID*

Long Syntax: V34.042 PPP dial circuit disconnect request on net *network ID*

Description: The PPP dial circuit has requested that V34 disconnect the call.

V34.043

Level: UE_ERROR

Short Syntax: V34.043 Timeout waiting for DCE to raise CTS on nt *network ID*

Long Syntax: V34.043 Timeout waiting for DCE to raise CTS on net *network ID*

Description: The router raised the DTR signal but the DCE did not raise CTS in a reasonable amount of time.

Cause: DCE not connected or powered-off, inoperable, or non-V.34 compliant.

Action: Attach the cable, check DCE configuration and power.

V34.044

Level: UE_ERROR

Short Syntax: V34.044 Timeout waiting for OK response from DCE on nt *network ID*

Long Syntax: V34.044 Timeout waiting for OK response from DCE on net *network ID*

Description: The router sent an AT command to the DCE and did not receive an OK response. Commands sent are ATZ, AT&F and the configurable modem init string.

Cause: DCE not connected, powered-off, inoperable, or non-V.34 compliant.

Action: Attach the cable, check DCE configuration and power. Power off/on modem.

Chapter 129. Web Server Cache - Core (WEBC)

This chapter describes Web Server Cache - Core (WEBC) messages. For information on message content and how to use the message, refer to the Introduction.

WEBC.001

Level: C-INFO

Short Syntax: WEBC.001 Partition *partition* initialized successfully

Long Syntax: WEBC.001 Partition *partition* initialized successfully

Description: A new cache partition was created

WEBC.002

Level: CI-ERROR

Short Syntax: WEBC.002 Partition *partition* initialization failed. *reason*

Long Syntax: WEBC.002 Partition *partition* initialization failed. *reason*

Description: The cache partition was not created due to the reason provided

WEBC.003

Level: C-INFO

Short Syntax: WEBC.003 Handle 0x *handle* obtained for partition *partition*

Long Syntax: WEBC.003 Handle 0x *handle* obtained for partition *partition*

Description: The cache handle was obtained for the given partition

WEBC.004

Level: CI-ERROR

Short Syntax: WEBC.004 Unable to obtain handle for partition *partition*. *reason*

Long Syntax: WEBC.004 Unable to obtain handle for partition *partition*. *reason*

Description: The cache handle was not granted due to the reason provided

WEBC.005

Level: C-INFO

Short Syntax: WEBC.005 Item added to partition *partition*

Long Syntax: WEBC.005 Item added to partition *partition*

Description: A cache item was added to the given partition

WEBC.006

Level: C-INFO

Short Syntax: WEBC.006 Item not added to partition *partition*

Long Syntax: WEBC.006 Item not added to partition *partition*

Description: A cache item was not added to the given partition. It did not pass the verification checks

WEBC.007

Level: CI-ERROR

Short Syntax: WEBC.007 Item not added to partition *partition*. *reason*

Long Syntax: WEBC.007 Item not added to partition *partition*. *reason*

Description: A cache item was not added to the given partition due to the reason provided

WEBC.008

Level: UI-ERROR

Short Syntax: WEBC.008 Item not added to partition *partition*. RC=0x *retcode*

Long Syntax: WEBC.008 Item not added to partition *partition*. RC=0x *retcode*

Description: A cache item was not added to the given partition

WEBC.009

Level: C-INFO

Short Syntax: WEBC.009 Item deleted from partition *partition*

Long Syntax: WEBC.009 Item deleted from partition *partition*

Description: A cache item was deleted from the given partition

WEBC.010

Level: CI-ERROR

Short Syntax: WEBC.010 Item not deleted from partition *partition*. *reason*

Long Syntax: WEBC.010 Item not deleted from partition *partition*. *reason*

Description: A cache entry was not deleted from the given partition due to the reason provided

WEBC.011

Level: UI-ERROR

Short Syntax: WEBC.011 Item not added to partition *partition*. RC=0x *retcode*

Long Syntax: WEBC.011 Item not added to partition *partition*. RC=0x *retcode*

Description: A cache entry was not deleted from the given partition

WEBC.012

Level: C-INFO

Short Syntax: WEBC.012 Partition *partition* being purged...

Long Syntax: WEBC.012 Partition *partition* being purged...

Description: The cache partition is being purged of all cache entries

WEBC.013

Level: C-INFO

Short Syntax: WEBC.013 Partition *partition* purge complete

Long Syntax: WEBC.013 Partition *partition* purge complete

Description: The cache partition has been purged of all cache entries

WEBC.014

Level: CI-ERROR

Short Syntax: WEBC.014 Partition *partition* not purged. *reason*

Long Syntax: WEBC.014 Partition *partition* not purged. *reason*

Description: The cache partition was not purged due to the reason provided

WEBC.015

Level: C-INFO

Short Syntax: WEBC.015 Item found in partition *partition*

Long Syntax: WEBC.015 Item found in partition *partition*

Description: A cache entry was found in the given partition

WEBC.016

Level: C-INFO

Short Syntax: WEBC.016 Item not found in partition *partition*

Long Syntax: WEBC.016 Item not found in partition *partition*

Description: A cache entry was not found in the given partition

WEBC.017

Level: CI-ERROR

Short Syntax: WEBC.017 Item not found in partition *partition*. *reason*

Long Syntax: WEBC.017 Item not found in partition *partition*. *reason*

Description: A cache entry was not found in the given partition due to the reason provided

WEBC.018

Level: C-INFO

Short Syntax: WEBC.018 Handle 0x *handle* retired

Long Syntax: WEBC.018 Handle 0x *handle* retired

Description: The cache handle was retired

WEBC.019

Level: C-INFO

Short Syntax: WEBC.019 Partition *partition* being terminated...

Long Syntax: WEBC.019 Partition *partition* being terminated...

Description: The cache partition is being terminated

WEBC.020

Level: C-INFO

Short Syntax: WEBC.020 Partition *partition* termination complete

Long Syntax: WEBC.020 Partition *partition* termination complete

Description: The cache partition has been terminated

WEBC.021

Level: CI-ERROR

Short Syntax: WEBC.021 Partition *partition* not terminated. *reason*

Long Syntax: WEBC.021 Partition *partition* not terminated. *reason*

Description: The cache partition was not terminated due to the reason provided

---

WEBC.022

Level: UI-ERROR

Short Syntax: WEBC.022 Garbage collection detected state *state* for partition *partition*

Long Syntax: WEBC.022 Garbage collection detected state *state* for partition *partition*

Description: The garbage collection routine detected an unknown state for the partition listed. Garbage collection stops

---

WEBC.023

Level: CI-ERROR

Short Syntax: WEBC.023 Partition *partition* init failed. RC=0x *retcode*

Long Syntax: WEBC.023 Partition *partition* initialization failed. Return code=0x *retcode*

Description: Web Server cache partition was defined, but cannot be initialized

---

WEBC.024

Level: CI-ERROR

Short Syntax: WEBC.024 Proxy init failed for partition *partition*. RC=0x *retcode*

Long Syntax: WEBC.024 Proxy initialization failed for partition *partition*. Return code=0x *retcode*

Description: Unable to define Web Server cache HTTP proxy

WEBC.025

Level: CI-ERROR

Short Syntax: WEBC.025 ECC Manager init failed. RC=0x *retcode*

Long Syntax: WEBC.025 ECC Manager initialization failed. Return Code=0x *retcode*

Description: The external cache manager initialization failed

---

WEBC.026

Level: C-INFO

Short Syntax: WEBC.026 0x *bytecnt* bytes of heap storage are available for Web Server cache

Long Syntax: WEBC.026 0x *bytecnt* bytes of heap storage are available for Web Server cache

Description: Provides total amount of heap storage that is available for the Web Server cache function to use

---

WEBC.027

Level: UE-ERROR

Short Syntax: WEBC.027 Insufficent heap storage (0x *bytecnt* bytes) to run Web Server cache

Long Syntax: WEBC.027 Insufficent heap storage (0x *bytecnt* bytes) to run Web Server cache

Description: There is not enough heap storage defined in order to run the Web Server cache function effectively. Severe storage problems may occur

---

WEBC.028

Level: C-INFO

Short Syntax: WEBC.028 Dependency (*dependency*) added to partition *partition*

Long Syntax: WEBC.028 New dependency (*dependency*) added to partition *partition*

Description: A new dependency was added to the dependency table for the partition

---

WEBC.029

Level: C-INFO

Short Syntax: WEBC.029 Dependency (*dependency*) deleted from partition *partition*

Long Syntax: WEBC.029 Dependency (*dependency*) deleted from partition *partition*

Description: A dependency was deleted from the dependency table for the partition

---

WEBC.030

Level: C-INFO

Short Syntax: WEBC.030 URL (*URL*) added to dependency (*dependency*) for partition *partition*

Long Syntax: WEBC.030 New URL (*URL*) was added to dependency (*dependency*) for partition *partition*

Description: A new URL was added to the given dependency for the partition

---

WEBC.031

Level: C-INFO

Short Syntax: WEBC.031 URL (*URL*) deleted from dependency (*dependency*) for partition *partition*

Long Syntax: WEBC.031 URL (*URL*) deleted from dependency (*dependency*) for partition *partition*

Description: A URL was deleted from the given dependency for the partition

---

WEBC.032

Level: C-INFO

Short Syntax: WEBC.032 Dependency (*dependency*) invalidated for partition *partition*

Long Syntax: WEBC.032 Dependency (*dependency*) invalidated for partition *partition*

Description: All URLs for the given dependency were removed for the partition

---

WEBC.033

Level: C-INFO

Short Syntax: WEBC.033 ECC Manager (port *port*) has started

Long Syntax: WEBC.033 External Cache Control Manager (port *port*) has started

Description: The External Cache Control Manager is listening of the given port.

WEBC.034

Level: C-INFO

Short Syntax: WEBC.034 ECC Manager (port *port*) has stopped

Long Syntax: WEBC.034 External Cache Control Manager (port *port*) has stopped

Description: The External Cache Control Manager is no longer listening on the given port.

WEBC.035

Level: C-INFO

Short Syntax: WEBC.035 ECC connection (*connection*) opened

Long Syntax: WEBC.035 External Cache Control connection (*connection*) opened

Description: A connection to the External Cache Control Manager has opened.

WEBC.036

Level: C-INFO

Short Syntax: WEBC.036 ECC connection (*connection*) closed

Long Syntax: WEBC.036 External Cache Control connection (*connection*) closed

Description: A connection to the External Cache Control Manager has closed.

Chapter 130. Web Server Cache - HTTP Proxy (WEBH)

This chapter describes Web Server Cache - HTTP Proxy (WEBH) messages. For information on message content and how to use the message, refer to the Introduction.

WEBH.001

Level: UE-ERROR

Short Syntax: WEBH.001 No Storage for HTTP Proxy(cluster *cluster* port *port*)

Long Syntax: WEBH.001 No Storage for HTTP Proxy(cluster *cluster* port *port*)

Description: The HTTP Proxy was not able to get required storage.

WEBH.002

Level: UE-ERROR

Short Syntax: WEBH.002 Parsing error for HTTP Proxy(cluster *cluster* port *port*) conn (*connection*)

Long Syntax: WEBH.002 Parsing error for HTTP Proxy(cluster *cluster* port *port*) connection (*connection*)

Description: The HTTP Proxy parser was unable to parse data. See event WEBH_8 for the TCP segment being parsed.

WEBH.003

Level: UE-ERROR

Short Syntax: WEBH.003 HTTP Proxy (cluster *cluster* port *port*) conn (*connection*) could not get additional segment descriptor from TCP

Long Syntax: WEBH.003 HTTP Proxy (cluster *cluster* port *port*) connection (*connection*) could not get additional segment descriptor from TCP

Description: The HTTP Proxy was not able to get additional segment descriptors from TCP for a block of data.

WEBH.004

Level: UE-ERROR

Short Syntax: WEBH.004 HTTP Proxy(cluster *cluster* port *port*) partition (*partition*) conn (*connection*) went to tunneling

Long Syntax: WEBH.004 HTTP Proxy(cluster *cluster* port *port*) partition (*partition*) connection (*connection*) went to tunneling

Description: The HTTP Proxy went to tunneling for the connection mention.

WEBH.005

Level: UE-ERROR

Short Syntax: WEBH.005 HTTP Proxy(cluster *cluster* port *port*) conn (*connection*) receive a non HTTP message

Long Syntax: WEBH.005 HTTP Proxy(cluster *cluster* port *port*) connection (*connection*) receive a non HTTP message

Description: The HTTP Proxy was not able to parse the message because it was not HTTP

WEBH.006

Level: UE-ERROR

Short Syntax: WEBH.006 HTTP Proxy(cluster *cluster* port *port*) conn (*connection*) received a boundary that was too big

Long Syntax: WEBH.006 HTTP Proxy(cluster *cluster* port *port*) connection (*connection*) received a boundary that was too big

Description: The HTTP Proxy was not able to parse the message because the boundary of the multipart/byteranges was bigger than this box supports.

WEBH.007

Level: UE-ERROR

Short Syntax: WEBH.007 HTTP Proxy(cluster *cluster* port *port*) conn (*connection*) received unsupported transfer-encoding

Long Syntax: WEBH.007 HTTP Proxy(cluster *cluster* port *port*) connection (*connection*) received unsupported transfer-encoding

Description: The HTTP Proxy was not able to parse the message because the transfer-encoding used was not supported.

WEBH.008

Level: P-TRACE

Short Syntax: WEBH.008 *string*

Long Syntax: WEBH.008 *string*

Description: Dump of segment. See other messages

for why it is being dumped.

WEBH.009

Level: P-TRACE

Short Syntax: WEBH.009 HTTP Proxy(cluster *cluster* port *port*) conn (*connection*) new req being parsed

Long Syntax: WEBH.009 HTTP Proxy(cluster *cluster* port *port*) connection (*connection*) new request being parser

Description: The HTTP Proxy parser is start to parse a new request

---

WEBH.010

Level: P-TRACE

Short Syntax: WEBH.010 HTTP Proxy(cluster *cluster* port *port*) conn (*connection*) new resp being parsed

Long Syntax: WEBH.010 HTTP Proxy(cluster *cluster* port *port*) connection (*connection*) new response being parsed

Description: The HTTP Proxy parser is starting to parse a new response

---

WEBH.011

Level: U-TRACE

Short Syntax: WEBH.011 HTTP Proxy(cluster *cluster* port *port*) partition (*partition*) conn (*connection*) not caching rsp because *string*

Long Syntax: WEBH.011 HTTP Proxy(cluster *cluster* port *port*) partition (*partition*) connection (*connection*) not caching rsp because *string*

Description: The HTTP Proxy parser is will not cache the response (see text for reason)

WEBH.012

Level: U-TRACE

Short Syntax: WEBH.012 HTTP Proxy(cluster *cluster* port *port*) partition (*partition*) conn (*connection*) not using cache because *string*

Long Syntax: WEBH.012 HTTP Proxy(cluster *cluster* port *port*) partition (*partition*) connection (*connection*) not using cache because *string*

Description: The HTTP Proxy parser/cache is not using cache for meeting the request (see text for reason)

WEBH.013

Level: C-INFO

Short Syntax: WEBH.013 HTTP Proxy(cluster *cluster* port *port*) partition (*partition*) started

Long Syntax: WEBH.013 HTTP Proxy(cluster *cluster* port *port*) partition (*partition*) started

Description: The HTTP Proxy was started for the cluster and port

---

WEBH.014

Level: C-INFO

Short Syntax: WEBH.014 HTTP Proxy(cluster *cluster* port *port*) partition (*partition*) ended

Long Syntax: WEBH.014 HTTP Proxy(cluster *cluster* port *port*) partition (*partition*) ended

Description: The HTTP Proxy was ended for the cluster and port

---

WEBH.015

Level: C-INFO

Short Syntax: WEBH.015 Conn (*connection*) HTTP Proxy(cluster *cluster* port *port*) partition (*partition*) opened

Long Syntax: WEBH.015 Connection (*connection*) HTTP Proxy(cluster *cluster* port *port*) partition (*partition*) opened

Description: The HTTP Proxy Connection was started for the cluster and port

---

WEBH.016

Level: C-INFO

Short Syntax: WEBH.016 Conn (*connection*) HTTP Proxy(cluster *cluster* port *port*) partition (*partition*) closed

Long Syntax: WEBH.016 Connection (*connection*) HTTP Proxy(cluster *cluster* port *port*) partition (*partition*) closed

Description: The HTTP Proxy Connection was closed for the cluster and port

---

WEBH.017

Level: C-INFO

Short Syntax: WEBH.017 Client connection *connection* accepted as Socket *socket*

Long Syntax: WEBH.017 Client connection *connection* accepted as Socket *socket*

Description: A client has started a new connection.

WEBH.018

Level: C-INFO

Short Syntax: WEBH.018 Server connection *connection* completed as Socket *socket*

Long Syntax: WEBH.018 Server connection *connection* completed as Socket *socket*

Description: A new connection has been started to the server.

WEBH.019

Level: UE-ERROR

Short Syntax: WEBH.019 CONNECTING_TO_SERVER is unexpected (socket *socket* state 0x *state*)

Long Syntax: WEBH.019 CONNECTING TO SERVER is unexpected (socket *socket* state 0x *state*)

Description: The transient CONNECTING_TO_SERVER state is unexpected here

---

WEBH.020

Level: C-INFO

Short Syntax: WEBH.020 Client connection *connection* (Socket *socket*) was handed off to *cache_member*

Long Syntax: WEBH.020 Client connection *connection* (Socket *socket*) was handed off to *cache_member*

Description: A connection has handed off to another cache in the cache array.

Chapter 131. WAN Restoral System (WRS)

This chapter describes WAN Restoral System (WRS) messages. For information on message content and how to use the message, refer to the Introduction.

WRS.001

Level: C-INFO

Short Syntax: WRS.001 Primary net *network ID* switching to secondary net *network ID*

Long Syntax: WRS.001 Primary interface number *network ID* switching to secondary interface number *network ID*

Description: The primary interface is being restored through the secondary circuit.

WRS.002

Level: C-INFO

Short Syntax: WRS.002 Primary net *network ID* restored on secondary net *network ID*

Long Syntax: WRS.002 Primary interface number *network ID* restored on secondary interface number *network ID*

Description: The primary interface has been restored through the secondary circuit.

WRS.004

Level: C-INFO

Short Syntax: WRS.004 Secondary net *network ID* switching back to primary net *network ID*

Long Syntax: WRS.004 Secondary interface number *network ID* switching back to primary interface number *network ID*

Description: The secondary interface is being restored through a secondary circuit.

WRS.006

Level: C-INFO

Short Syntax: WRS.006 Switch to sec net *network ID* averted pri net *network ID* disabled

Long Syntax: WRS.006 Switch to secondary interface number *network ID* averted primary interface number *network ID* disabled

Description: The switch to secondary interface has been averted, primary interface is disabled.

WRS.007

Level: C-INFO

Short Syntax: WRS.007 Secondary net *network ID* failed resort to primary net *network ID*

Long Syntax: WRS.007 Secondary interface number *network ID* resorting back to primary interface number *network ID*

Description: The secondary interface has gone down causing a switch back to the primary circuit.

WRS.008

Level: C-INFO

Short Syntax: WRS.008 Sec net *network ID* swt to AVL; pri net *network ID* bck ONL

Long Syntax: WRS.008 Secondary net number *network ID* switch to AVAILABLE; primary net number *network ID* back ONLINE

Description: The switch to secondary interface has been aborted, primary still active and on-line.

WRS.009

Level: C-TRACE

Short Syntax: WRS.009 Packet forwarded pri net *network ID* onto sec net *network ID*

Long Syntax: WRS.009 Packet forwarded from the primary interface number *network ID* onto the secondary interface number *network ID*

Description: A packet has been forwarded from the primary interface onto the secondary interface.

WRS.010

Level: C-TRACE

Short Syntax: WRS.010 Packet received on pri net *network ID* from sec net *network ID*

Long Syntax: WRS.010 Packet received on primary interface number *network ID* from secondary interface number *network ID*

Description: A packet has been received onto the primary interface from the secondary interface.

WRS.012

Level: C-TRACE

Short Syntax: WRS.012 Unable to forward pri net *network ID* onto sec net *network ID*

Long Syntax: WRS.012 Packet forwarded from the primary interface number *network ID* onto the secondary interface number failed *network ID*

Description: A packet cannot be forwarded from the primary interface onto the secondary interface.

WRS.013

Level: C-INFO

Short Syntax: WRS.013 Switch to sec net *network ID* aborted, sec restoral disabled

Long Syntax: WRS.013 Switch to secondary interface number *network ID* aborted secondary restoral disabled

Description: The switch to secondary interface has been aborted, secondary restoral is disabled.

WRS.022

Level: UE-ERROR

Short Syntax: WRS.022 Protocol initialization on sec ignored, prot = *type* on nt *network ID*

Long Syntax: WRS.022 Protocol initialization on secondary ignored, protocol = *type* on network *network ID*

Description: Invalid protocol configured on secondary circuit.

Cause: Software configuration out of date, contact customer service.

WRS.023

Level: UE-ERROR

Short Syntax: WRS.023 Sec int disabled, mismatch datalink nt *network ID*

Long Syntax: WRS.023 Secondary interface disabled, mismatched datalink type network *network ID*

Description: Mismatched data-link type was configured on secondary interface; data-link type must match primary interface.

WRS.024

Level: C-INFO

Short Syntax: WRS.024 Perform n_up for net *network ID*

Long Syntax: WRS.024 Perform deferred net-up for interface number *network ID*

Description: The specified primary interface has been up continuously for the configured stabilization period, so the router posts the deferred net-up notification.

WRS.025

Level: C-INFO

Short Syntax: WRS.025 Reroute pri *network ID* alt *network ID*

Long Syntax: WRS.025 Begin rerouting for primary *network ID* using alternate *network ID*

Description: The specified primary interface went down (or has not come up within the configured first-stabilization period) so the router brings up the alternate to provide rerouting service.

WRS.026

Level: C-INFO

Short Syntax: WRS.026 End reroute pri *network ID* alt *network ID*

Long Syntax: WRS.026 End rerouting for primary *network ID* using alternate *network ID*

Description: The specified primary interface no longer requires the rerouting services of the alternate. If no other primary interfaces need the alternate's services, the router restores the alternate to its state before the router brought it up for rerouting.

WRS.027

Level: C-INFO

Short Syntax: WRS.027 Queue deferred n_up for pri *network ID*

Long Syntax: WRS.027 Queue deferred net-up for interface number *network ID*

Description: The specified primary interface came up, but the router defers the net-up notification for the configured stabilization period.

WRS.028

Level: C-INFO

Short Syntax: WRS.028 Queue delayed n_up for pri *network ID*

Long Syntax: WRS.028 Queue delayed net-up for interface number *network ID*

Description: The specified primary interface came up, but the router delayed the net-up notification until the configured time-of-day revert-back start time. Stabilization period, if configured, has already passed.

WRS.029

Level: C-INFO

Short Syntax: WRS.029 Dial-ofl switch some prots pri *network ID* alt *network ID*

Long Syntax: WRS.029 Dial-on-overflow switching some protocols from primary *network ID* to alternate *network ID*

Description: The dial-on-overflow add-threshold was exceeded and the defined protocols are being switched to the alternate interface.

WRS.030

Level: C-TRACE

Short Syntax: WRS.030 Dial-ofl traffic pri *network ID* sp *speed* add *add-count* drp *drop-count* in: *in-count* out: *out-count*

Long Syntax: WRS.030 Dial-on-overflow sample traffic primary *network ID* speed *speed* add-count *add-count* drop-count *drop-count* in-count: *in-count* out-count: *out-count*

Description: The dial-on-overflow function is sampling the traffic on the primary link to determine whether the add- or drop-thresholds are exceeded. The configured link speed, the threshold counts for adding or dropping the alternate, and the actual in and out counts during the last interval are reported.

WRS.031

Level: C-INFO

Short Syntax: WRS.031 Dial-ofl not enabled pri *network ID* wrong dta lnk type

Long Syntax: WRS.031 Dial-on-overflow not enabled primary *network ID* wrong primary data link type

Description: Dial-on-overflow was configured for a primary link that has a data link type that is invalid for dial-on-overflow.

WRS.032

Level: C-INFO

Short Syntax: WRS.032 Dial-ofl pri *network ID* alt *network ID* min alt up time *sec* sec exprd

Long Syntax: WRS.032 Dial-on-overflow primary *network ID* alternate *network ID* minimum alternate up time *sec* seconds expired

Description: The minimum dial-on overflow alternate up time for the specified primary/alternate pair expired. The dial-on-overflow protocols will be switched back to the primary link.

WRS.033

Level: C-INFO

Short Syntax: WRS.033 WRR alt *network ID* down, was rerouting for pri *network ID*

Long Syntax: WRS.033 Wan reroute alternate link *network ID* has been marked down, was rerouting for primary *network ID*

Description: The wan-reroute alternate link that was actively rerouting traffic for the specified primary has been marked down.

Panic wrsimem

Short Syntax: WAN restoral initialization failed, no memory.

Description: The WAN restoral initialization failed to allocate sufficient memory to complete initialization.

Action: Contact customer service.

Chapter 132. Xerox Network Core (XN)

This chapter describes Xerox Network Core (XN) messages. For information on message content and how to use the message, refer to the Introduction.

XN.001

Level: UE-ERROR

Short Syntax: XN.001 *protocol* trunc pkt frm *source_net*/ *source_node*, xns *length* phys *length*

Long Syntax: XN.001 *protocol* truncated packet from *source_net*/ *source_node*; xns *length*, physical *length*

Description: This message is generated when a packet has an XNS packet length greater then the packets physical length.

Cause: Programming error in remote node, truncation by network.

XN.004

Level: UE-ERROR

Short Syntax: XN.004 *protocol* hop cnt ovflo frm *source_net*/ *source_node* to *destination_net*

Long Syntax: XN.004 *protocol* hop count overflow from *source_net*/ *source_node* to *destination_net*

Description: This message is generated when a packet's hop count counts up past 15 and overflows. An Error (hop count) packet will be sent.

XN.005

Level: CE-ERROR

Short Syntax: XN.005 *protocol* no gwy frm *source_net*/ *source_node* to *destination_net*

Long Syntax: XN.005 *protocol* no gateway from *source_net*/ *source_node* to *destination_net*

Description: This message is generated when a packet cannot be forwarded because there is no gateway to the destination network. An Error (unreachable) packet will be sent.

XN.007

Level: UE-ERROR

Short Syntax: XN.007 *protocol* dst hst 0 frm *source_net*/ *source_node* to *destination_net*

Long Syntax: XN.007 *protocol* destination host 0 from *source_net*/ *source_node* to *destination_net*

Description: This message is generated when a packet is addressed to node 000000000000. This is an illegal host address. An Error (checksum) packet will be sent.

XN.008

Level: P-TRACE

Short Syntax: XN.008 *protocol source_net*/ *source_node* -> *dest_net*/ *dest_node*

Long Syntax: XN.008 *protocol* Packet received from *source_net*/ *source_node* for *dest_net*/ *dest_node*

Description: This message is generated when a packet is forwarded.

XN.009

Level: UE-ERROR

Short Syntax: XN.009 *protocol* pkt too lng to frwd *pkt_size* > *max_size* nt *output network ID* frm *source_net*/ *source_node*

Long Syntax: XN.009 *protocol* packet too long to forward *pkt_size* > *max_size* net *output network ID* from *source_net*/ *source_node*

Description: This message is generated when a forwarded packet cannot be sent out the required interface because it is too long. An Error (size) packet will be sent.

XN.010

Level: UI-ERROR

Short Syntax: XN.010 *protocol type* frm *source_net*/ *source_node* for *dest_net*/ *dest_node* dsc, rsn *code*

Long Syntax: XN.010 *protocol type* from *source_net*/ *source_node* for *dest_net*/ *dest_node* discarded for reason *code*

Description: An outgoing packet was not successfully transmitted for the reason indicated by the error code.

XN.011

Level: C-INFO

Short Syntax: XN.011 *protocol* intrfc *network*/ *node* nt *network ID* up

Long Syntax: XN.011 *protocol* interface *network*/ *node* net *network ID* up

Description: The specified interface has come up, and has been enabled for the specified XNS protocol.

XN.012

Level: U-INFO

Short Syntax: XN.012 *protocol* del nt *destination_net* rt via *gateway* nt *network ID*

Long Syntax: XN.012 *protocol* deleted net *destination_net* route via *gateway* net *network ID*

Description: The specified route has been deleted because the first hop interface for that route has gone down.

Cause: Interface down.

Action: Fix network.

XN.013

Level: UI-ERROR

Short Syntax: XN.013 *protocol* tbl ovrfl, dst *destination_net*

Long Syntax: XN.013 *protocol* Table overflow, destination *destination_net*

Description: This message is generated when a new entry cannot be made to routing table because it is already full.

Cause: Routing table too small.

Action: Increase routing table size for this protocol.

XN.014

Level: C-INFO

Short Syntax: XN.014 *protocol* echo typ *operation* to skt *socket* frm *source_net*/ *source_node*

Long Syntax: XN.014 *protocol* Echo type *operation* to socket *socket* from *source_net*/ *source_node*

Description: A packet of the echo type has been received with the specified operation to the specified socket.

XN.015

Level: UE-ERROR

Short Syntax: XN.015 *protocol* bd src *source_net*/ *source_node* nt *network ID*

Long Syntax: XN.015 *protocol* bad source *source_net*/ *source_node* net *network ID*

Description: A packet was being returned to the sender, but the senders node address was a multicast address or the illegal address 000000000000. This can happen when sending an Echo reply, an Error packet, or replying to other queries. The packet will be discarded.

XN.016

Level: UE-ERROR

Short Syntax: XN.016 *protocol* bad net 0 *source_net*/ *source_node*-> *dest_net*/ *dest_node*

Long Syntax: XN.016 *protocol* bad source network 0 from *source_net*/ *source_node* for *dest_net*/ *dest_node*

Description: A packet was being returned to the sender, and the source network was zero, but the destination network was non-zero. This can happen when sending an Echo reply, an Error packet, or replying to other queries. The packet will be discarded.

XN.017

Level: UI-ERROR

Level: OOM

Short Syntax: XN.017 *protocol* no mem for err pkt

Long Syntax: XN.017 *protocol* No memory for error packet

Description: This message is generated when no memory is available to copy the offending packet into an Error packet. An Error packet will not be sent.

XN.018

Level: UE-ERROR

Short Syntax: XN.018 *protocol* short (*length*) pkt frm *source_net*/ *source_node* (?) nt *network ID*

Long Syntax: XN.018 *protocol* short (*length*) packet from *source_net*/ *source_node* (?) net *network ID*

Description: This message is generated when a packet has a physical length shorter than the minimum 30 byte XNS, IPX, or DDS header length. The source_net and source_node may or may not be valid packet data, depending on how severe the truncation is.

Cause: Programming error in remote node, truncation by network.

XN.019

Level: C-TRACE

Short Syntax: XN.019 *protocol* chg src net to *new_source_net*, pkt *source_net*/ *source_node* -> *dest_net*/ *dest_node*

Long Syntax: XN.019 *protocol* changing source network to *new_source_net* on packet received from *source_net*/ *source_node* for *dest_net*/ *dest_node*

Description: This message is generated when an IPX packet is received with a source network number of 0. The router corrects this to be the network number of the interface the packet was received on.

XN.020

Level: UE-ERROR

Short Syntax: XN.020 *protocol* bad src net 0, hop count *hop_count*, *source_net*/ *source_node* -> *dest_net*/ *dest_node*, nt *network ID*

Long Syntax: XN.020 *protocol* bas source network 0 with hop count *hop_count* on packet received from *source_net*/ *source_node* for *dest_net*/ *dest_node* via network *network ID*

Description: This message is generated when an IPX packet is received with a source network number of 0, and the hop count (transport control) is non-zero. The source network number will not be corrected, since it is probably not the network it was received on. The dest_node will be unable to reply.

Cause: This would indicate that the packet has already been forwarded by another router that does not correct the source network number when forwarding, or that the originating node sent the packet with a non-zero hop count.

Action: Correct programming error at remote node or router.

Chapter 133. X.25 Network Interface (X25)

This chapter describes X.25 Network Interface (X25) messages. For information on message content and how to use the message, refer to the Introduction.

X25.001

Level: CI-ERROR

Short Syntax: X25.001 no nde addr nt *network ID*

Long Syntax: X25.001 node address not assigned network index *network ID*

Description: The X.25 network handler cannot continue initialization due to lack of X.25 node address assignment.

X25.002

Level: CI-ERROR

Short Syntax: X25.002 prtcl not cnfg nt *network ID*

Long Syntax: X25.002 protocol forwarder not configured network index *network ID*

Description: The X.25 network handler received a protocol pre-initialization which resulted in using default configuration. The protocol has not been configured.

X25.003

Level: UI-ERROR

Short Syntax: X25.003 req unkn nt *network ID*

Long Syntax: X25.003 request unknown network index *network ID*

Description: The X.25 network handler received an unknown request either via the console interface or due to a forwarder problem. The request is simply ignored.

X25.004

Level: UI-ERROR

Short Syntax: X25.004 dev int dwn drng init nt *network ID*

Long Syntax: X25.004 device driver constantly down during initialization network index *network ID*

Description: The X.25 network handler is waiting on the device driver to complete the CPU to COM-4 initialization sequence.

Action: If the situation persists, reset the COM-4 interface. Contact customer service.

X25.005

Level: CI-ERROR

Short Syntax: X25.005 PVC *LCN* rnge nt *network ID*

Long Syntax: X25.005 PVC *LCN* lies outside configured PVC range: network *network ID*

Description: The X.25 network handler cannot continue initialization due to a configuration conflict: the identified PVC lies outside the configured PVC range.

X25.006

Level: CI-ERROR

Short Syntax: X25.006 LCN overlap nt *network ID*

Long Syntax: X25.006 One or more logical channel ranges overlap : network *network ID*

Description: The X.25 network handler cannot continue initialization due to a configuration conflict: the configured logical channel ranges overlap. For non-zero ranges, the following inequalities must hold: LOW-PVC <= HI-PVC < LOW-INBOUND <= HIGH-INBOUND < LOW-TWO-WAY <= HIGH-TWO-WAY < LOW-OUTBOUND <= HIGH-OUTBOUND.

X25.007

Level: CI-ERROR

Short Syntax: X25.007 pkt dflt > max nt *network ID*

Long Syntax: X25.007 Packet default size greater than maximum size: network *network ID*

Description: The X.25 network handler cannot continue initialization due to a configuration conflict: configured default packet size exceeds configured maximum packet size.

X25.008

Level: UI-ERROR

Short Syntax: X25.008 call req prot not supprtd nt *network ID*, *x25_source*-> *x25_destination* cud= *call_user_data*

Long Syntax: X25.008 call request protocol not supported network index *network ID*, calling dte

x25_source -> called dte *x25_destination* with call user data *call_user_data*

Description: The X.25 network handler received a call request indicating an unsupported protocol.

X25.009

Level: UI-ERROR

Short Syntax: X25.009 No heap for nt *network ID*

Long Syntax: X25.009 Insufficient heap to complete initialization of network *network ID*

Description: The X.25 network requires a sizeable amount of heap storage to initialize, based on the number of PVCs, the size of the PVC range, and to a lesser extent, the number of addresses defined, protocols enabled,and the size of the SVC ranges. If this memory isn't available, X.25 cannot run. The interface will disable itself, and stay disabled.

Action: Consider reducing the size of the X.25 tables, or the size of other configurable tables (routing tables) in the router.

X25.010

Level: CI-ERROR

Short Syntax: X25.010 net int dwn nt *network ID*

Long Syntax: X25.010 network interface went down network index *network ID*

Description: The X.25 network handler detected the network interface moving to a down state. The handler will monitor for a brief period prior to notifying protocol forwarders of the situation.

Panic x25intm

Short Syntax: X25: net intf mismatch

Description: The X.25 data structure ″net″ is not X.25 related.

Action: Contact customer service.

Panic x25iprt

Short Syntax: X25: unsuppt prt drng init

Description: The X.25 network handler detected an unsupported protocol during initialization.

Action: Contact customer service.

Panic x25imem

Short Syntax: X25: mem alloc fld

Description: The X.25 network handler failed to allocate sufficient memory during the initialization phase.

Action: Contact customer service.

Chapter 134. X.25 Network Interface Frame Layer (X252)

This chapter describes X.25 Network Interface Frame Layer (X252) messages. For information on message content and how to use the message, refer to the Introduction.

X252.003

Level: C-INFO

Short Syntax: X252.003 frm lyr up nt *network ID*

Long Syntax: X252.003 Frame layer up network *network ID*

Description: The frame layer is up.

X252.004

Level: C-INFO

Short Syntax: X252.004 frm lyr dn reason *reason* nt *network ID*

Long Syntax: X252.004 Frame layer down reason *reason* network *network ID*

Description: The frame layer is down.

X252.005

Level: P-TRACE

Short Syntax: X252.005 I-frm rxd nt *network ID*

Long Syntax: X252.005 I-frame received from network *network ID*

Description: A good I-frame was received from the network.

X252.006

Level: P-TRACE

Short Syntax: X252.006 I-frm txd nt *network ID*

Long Syntax: X252.006 I-frame transmitted to network *network ID*

Description: A good I-frame was transmitted to the network.

X252.007

Level: P-TRACE

Short Syntax: X252.007 rr rxd nt *network ID*

Long Syntax: X252.007 rr received from network *network ID*

Description: A frame layer RR was received from the network.

X252.008

Level: P-TRACE

Short Syntax: X252.008 rr txd nt *network ID*

Long Syntax: X252.008 rr transmitted to network *network ID*

Description: A frame layer RR was transmitted to the network.

X252.009

Level: P-TRACE

Short Syntax: X252.009 rnr rxd nt *network ID*

Long Syntax: X252.009 rnr received from network *network ID*

Description: A frame layer RNR was received from the network.

X252.010

Level: P-TRACE

Short Syntax: X252.010 rnr txd nt *network ID*

Long Syntax: X252.010 rnr transmitted to network *network ID*

Description: A frame layer RNR was transmitted to the network.

X252.011

Level: P-TRACE

Short Syntax: X252.011 rej rxd nt *network ID*

Long Syntax: X252.011 rej received from network *network ID*

Description: A frame layer Reject was received from the network.

X252.012

Level: P-TRACE

Short Syntax: X252.012 rej txd nt *network ID*

Long Syntax: X252.012 rej transmitted to network *network ID*

Description: A frame layer Reject was transmitted to the network.

X252.013

Level: P-TRACE

Short Syntax: X252.013 sabme rxd nt *network ID*

Long Syntax: X252.013 sabme received from network *network ID*

Description: A SABME frame was received from the network.

X252.014

Level: P-TRACE

Short Syntax: X252.014 sabme txd nt *network ID*

Long Syntax: X252.014 sabme transmitted to network *network ID*

Description: A SABME frame was transmitted to the network.

X252.015

Level: P-TRACE

Short Syntax: X252.015 sabm rxd nt *network ID*

Long Syntax: X252.015 sabm received from network *network ID*

Description: A SABM frame was received from the network.

X252.016

Level: P-TRACE

Short Syntax: X252.016 sabm txd nt *network ID*

Long Syntax: X252.016 sabm transmitted to network *network ID*

Description: A SABM frame was transmitted to the network.

X252.017

Level: P-TRACE

Short Syntax: X252.017 disc rxd nt *network ID*

Long Syntax: X252.017 disc received from network *network ID*

Description: A DISC frame was received from the network.

X252.018

Level: P-TRACE

Short Syntax: X252.018 disc txd nt *network ID*

Long Syntax: X252.018 disc transmitted to network *network ID*

Description: A DISC frame was transmitted to the network.

X252.019

Level: P-TRACE

Short Syntax: X252.019 dm rxd nt *network ID*

Long Syntax: X252.019 dm received from network *network ID*

Description: A DM frame was received from the network.

X252.020

Level: P-TRACE

Short Syntax: X252.020 dm txd nt *network ID*

Long Syntax: X252.020 dm transmitted to network *network ID*

Description: A DM frame was transmitted to the network.

X252.021

Level: P-TRACE

Short Syntax: X252.021 ua rxd nt *network ID*

Long Syntax: X252.021 ua received from network *network ID*

Description: A UA frame was received from the network.

X252.022

Level: P-TRACE

Short Syntax: X252.022 ua txd nt *network ID*

Long Syntax: X252.022 ua transmitted to network *network ID*

Description: A UA frame was transmitted to the network.

X252.026

Level: UE-ERROR

Short Syntax: X252.026 frmr I-frm too lng txd nt *network ID*

Long Syntax: X252.026 frame reject for I-frame too

long transmitted to network *network ID*

Description: A frame reject indicating that an I-frame was too long was sent to the network.

X252.027

Level: UE-ERROR

Short Syntax: X252.027 frmr N(R) invld rxd nt *network ID*

Long Syntax: X252.027 frame reject for N(R) invalid received from network *network ID*

Description: A frame reject indicating that an invalid N(R) was received from the network.

X252.028

Level: UE-ERROR

Short Syntax: X252.028 frmr N(R) invld txd nt *network ID*

Long Syntax: X252.028 frame reject for N(R) invalid transmitted to network *network ID*

Description: A frame reject indicating that an invalid N(R) was received was sent to the network.

X252.033

Level: C-INFO

Short Syntax: X252.033 t2 tmr exp nt *network ID*

Long Syntax: X252.033 T2 timer expired network *network ID*

Description: The T2 timer has expired for the indicated network.

Chapter 135. X.25 Network Interface Packet Layer (X253)

This chapter describes X.25 Network Interface Packet Layer (X253) messages. For information on message content and how to use the message, refer to the Introduction.

X253.003

Level: C-INFO

Short Syntax: X253.003 pkt lyr up nt *network ID*

Long Syntax: X253.003 Packet layer up network *network ID*

Description: The packet layer is up.

X253.004

Level: C-INFO

Short Syntax: X253.004 pkt lyr dn reason *reason* nt *network ID*

Long Syntax: X253.004 Packet layer down reason *reason* network *network ID*

Description: The packet layer is down.

X253.005

Level: P-TRACE

Short Syntax: X253.005 data pkt rxd lcn *lcn* nt *network ID*

Long Syntax: X253.005 Data Packet received on lcn *lcn* from network *network ID*

Description: A good Data Packet was received from the network.

X253.006

Level: P-TRACE

Short Syntax: X253.006 data pkt txd lcn *lcn* nt *network ID*

Long Syntax: X253.006 Data Packet transmitted on lcn *lcn* to network *network ID*

Description: A good Data Packet was transmitted to the network.

X253.007

Level: P-TRACE

Short Syntax: X253.007 call ind rxd lcn *lcn* nt *network ID*

Long Syntax: X253.007 Call indication received for lcn *lcn* from network *network ID*

Description: A Call Indication was received for the indicated lcn from the network.

X253.008

Level: P-TRACE

Short Syntax: X253.008 cll rq txd lcn *lcn* nt *network ID* with calling dte *calling_dte_addr* and called dte *called_dte_addr* facilities *facilities call_user_data* cud

Long Syntax: X253.008 Call request packet transmitted for lcn *lcn* network *network ID* from calling dte *calling_dte_addr* to called dte *called_dte_addr* facilities *facilities call_user_data* cud

Description: A Call Request was transmitted for the indicated lcn to the network.

X253.009

Level: P-TRACE

Short Syntax: X253.009 cll cnf rxd lcn *lcn* nt *network ID*

Long Syntax: X253.009 Call Confirmation Packet received for lcn *lcn* network *network ID*

Description: A call conformation for the indicated lcn was received from the network.

X253.010

Level: P-TRACE

Short Syntax: X253.010 cll acpt txd lcn *lcn* nt *network ID*

Long Syntax: X253.010 Call Accepted Packet transmitted for lcn *lcn* network *network ID*

Description: A Call Accepted for the indicated lcn was transmitted to the network.

X253.011

Level: P-TRACE

Short Syntax: X253.011 rr rxd lcn *lcn* nt *network ID*

Long Syntax: X253.011 RR Packet received for lcn *lcn* network *network ID*

Description: An RR for the indicated lcn was received from the network.

X253.012

Level: P-TRACE

Short Syntax: X253.012 rr txd lcn *lcn* nt *network ID*

Long Syntax: X253.012 RR Packet transmitted for lcn *lcn* network *network ID*

Description: An RR for the indicated lcn was transmitted to the network.

X253.013

Level: P-TRACE

Short Syntax: X253.013 rnr rxd lcn *lcn* nt *network ID*

Long Syntax: X253.013 RNR Packet received for lcn *lcn* network *network ID*

Description: An RNR for the indicated lcn was received from the network.

X253.014

Level: P-TRACE

Short Syntax: X253.014 rnr txd lcn *lcn* nt *network ID*

Long Syntax: X253.014 RNR Packet transmitted for lcn *lcn* network *network ID*

Description: An RNR for the indicated lcn was transmitted to the network.

X253.017

Level: P-TRACE

Short Syntax: X253.017 clr rq rxd lcn *lcn* cse *clearing_cause* diag *clearing_diagnostic* nt *network ID*

Long Syntax: X253.017 Clear request received for lcn *lcn* cause *clearing_cause* diagnostic *clearing_diagnostic* network *network ID*

Description: A clear request for the indicated lcn was received from the network.

X253.018

Level: P-TRACE

Short Syntax: X253.018 clr rq txd lcn *lcn* cse *clearing_cause* diag *clearing_diagnostic* nt *network ID*

Long Syntax: X253.018 Clear request transmitted for lcn *lcn* cause *clearing_cause* diagnostic *clearing_diagnostic* network *network ID*

Description: A clear request for the indicated lcn was transmitted to the network.

X253.019

Level: P-TRACE

Short Syntax: X253.019 clr cnf rxd lcn *lcn* nt *network ID*

Long Syntax: X253.019 Clear confirm received for lcn *lcn* network *network ID*

Description: A clear confirm for the indicated lcn was received from the network.

X253.020

Level: P-TRACE

Short Syntax: X253.020 clr cnf txd lcn *lcn* nt *network ID*

Long Syntax: X253.020 Clear confirm transmitted to lcn *lcn* network *network ID*

Description: A clear confirm for the indicated lcn was transmitted to the network.

X253.023

Level: P-TRACE

Short Syntax: X253.023 rset rxd lcn *lcn* cse *reset_cause* diag *reset_diagnostic* nt *network ID*

Long Syntax: X253.023 Reset received for lcn *lcn* cause *reset_cause* diagnostic *reset_diagnostic* network *network ID*

Description: A reset for the indicated lcn was received from the network.

X253.024

Level: P-TRACE

Short Syntax: X253.024 rset txd lcn *lcn* cse *reset_cause* diag *reset_diagnostic* nt *network ID*

Long Syntax: X253.024 Reset transmitted for lcn *lcn* cause *reset_cause* diagnostic *reset_diagnostic* network *network ID*

Description: A reset for the indicated lcn was transmitted to the network.

X253.025

Level: P-TRACE

Short Syntax: X253.025 rset cnf rxd lcn *lcn* nt *network ID*

Long Syntax: X253.025 Reset confirm received for lcn *lcn* network *network ID*

Description: A reset confirm for the indicated lcn was received from the network.

X253.026

Level: P-TRACE

Short Syntax: X253.026 rset cnf txd lcn *lcn* nt *network ID*

Long Syntax: X253.026 Reset confirm transmitted for lcn *lcn* network *network ID*

Description: A reset confirm for the indicated lcn was transmitted to the network.

X253.027

Level: P-TRACE

Short Syntax: X253.027 rstrt rxd lcn *lcn* cse *restart_cause* diag *restart_diagnostic* nt *network ID*

Long Syntax: X253.027 Restart received lcn *lcn* cause *restart_cause* diagnostic *restart_diagnostic* network *network ID*

Description: A restart was received from the network.

X253.028

Level: P-TRACE

Short Syntax: X253.028 rstrt txd lcn *lcn* cse *restart_cause* diag *restart_diagnostic* nt *network ID*

Long Syntax: X253.028 Restart transmitted lcn *lcn* cause *restart_cause* diagnostic *restart_diagnostic* network *network ID*

Description: A restart was transmitted to the network.

X253.029

Level: P-TRACE

Short Syntax: X253.029 rstrt cnf rxd lcn *lcn* nt *network ID*

Long Syntax: X253.029 Restart confirm received lcn *lcn* network *network ID*

Description: A restart confirm was received from the network.

X253.030

Level: P-TRACE

Short Syntax: X253.030 rstrt cnf txd lcn *lcn* nt *network ID*

Long Syntax: X253.030 Restart confirm transmitted lcn *lcn* network *network ID*

Description: A restart confirm was transmitted to the network.

X253.031

Level: P-TRACE

Short Syntax: X253.031 diag txd diag cde *diagnostic_code* nt *network ID*

Long Syntax: X253.031 Diagnostic transmitted diagnostic code *diagnostic_code* network *network ID*

Description: A diagnostic packet was transmitted to the network.

X253.032

Level: P-TRACE

Short Syntax: X253.032 diag rxd diag cde *diagnostic_code* nt *network ID*

Long Syntax: X253.032 Diagnostic received diagnostic code *diagnostic_code* network *network ID*

Description: A diagnostic packet was received from the network.

X253.033

Level: C-INFO

Short Syntax: X253.033 rstrt tmr exp lcn *lcn* nt *network ID*

Long Syntax: X253.033 Restart timer expired lcn *lcn* network *network ID*

Description: The restart timer has expired for the indicated network.

X253.040

Level: U-INFO

Short Syntax: X253.040 lost data - excessive X.25 mbit processing lcn *network ID*/

Long Syntax: X253.040 lost data - excessive X.25 mbit processing lcn *network ID*/

Description: X.25 mbit processing is demanding too many buffers. Fine tune packet size and mtu size configuration parameters for either larger packets or a smaller message size.

X253.041

Level: U-INFO

Short Syntax: X253.041 *lcn* expired on lcn *network ID*/

Long Syntax: X253.041 *lcn* expired on lcn *network ID*/

Description: A packet layer timer expired...the indicated circuit may have been reset or cleared.

X253.042

Level: P-TRACE

Short Syntax: X253.042 call ind rxd lcn *lcn* nt *network ID* with calling dte *calling_dte_addr* and called dte *called_dte_addr* facilities *facilities call_user_data* cud

Long Syntax: X253.042 Call indication received for lcn *lcn* from network *network ID* from calling dte *calling_dte_addr* to called dte *called_dte_addr* facilities *facilities call_user_data* cud

Description: A Call Indication was received for the indicated lcn from the network.

X253.043

Level: U-INFO

Short Syntax: X253.043 *peer_DTE_addr* for DTE addr *network ID/ buffers_returned* buffers returned

Long Syntax: X253.043 *peer_DTE_addr* for DTE addr *network ID/ buffers_returned* buffers returned

Description: A packet layer timer expired... returned buffers because no data being processed

Chapter 136. X.25 Network Interface Physical Layer (X251)

This chapter describes X.25 Network Interface Physical Layer (X251) messages. For information on message content and how to use the message, refer to the Introduction.

X251.001

Level: C-INFO

Short Syntax: X251.001 Mdm sts chg: DSR/DCD/CTS *DSR/ DCD/ CTS* nt *network ID*

Long Syntax: X251.001 Modem status changed DSR = *DSR* DCD = *DCD* CTS = *CTS* on network *network ID*

Description: The (input) modem control signals have changed, the present state of the input signals is as specified.

X251.003

Level: C-INFO

Short Syntax: X251.003 Srl prt up, nt *network ID*

Long Syntax: X251.003 Serial port came up sucessfully, on network *network ID*

Description: x25_s2 routine liked the results of the load and init.

X251.004

Level: UI-ERROR

Short Syntax: X251.004 TxCmp Rsys Schd fll nt *network ID*

Long Syntax: X251.004 Rsys ring full on Transmit complete: network *network ID*

Description: An attempt to enqueue a transmit complete notification to the frame layer of X.25 failed, due to a full internal scheduler ring. This will result in the loss of buffers.

X251.006

Level: CE_ERROR

Short Syntax: X251.006 RxOvr nt *network ID*

Long Syntax: X251.006 Receiver overrun: frame too long network *network ID*

Description: A frame was received with a correct CRC, but which exceeded the (configured) maximum length.

X251.007

Level: CE_ERROR

Short Syntax: X251.007 RxErr st *status* nt *network ID*

Long Syntax: X251.007 Receiver error: Erroneous frame (driver status *status*) received on network *network ID*

Description: A frame was received in error (bad CRC, modem signals down, etc).

X251.008

Level: C-INFO

Short Syntax: X251.008 Frm Rxd nt *network ID*

Long Syntax: X251.008 Frame received from network *network ID*

Description: A good frame was received from the network.

X251.009

Level: CE-ERROR

Short Syntax: X251.009 Frm Tx Flsh nt *network ID*

Long Syntax: X251.009 Outbound frame flushed on network *network ID*

Description: A frame transmit was aborted due to protocol state or event.

X251.010

Level: CE-ERROR

Short Syntax: X251.010 Frm Txd Fail st *status* nt *network ID*

Long Syntax: X251.010 Frame transmission failed, status *status*, on network *network ID*

Description: A frame transmission to the network failed; the driver returned the specified status.

X251.011

Level: C-INFO

Short Syntax: X251.011 Frm Txd nt *network ID*

Long Syntax: X251.011 Frame sucessfully transmitted to network *network ID*

Description: A frame was successfully transmitted to the network.

X251.012

Level: CI-ERROR

Short Syntax: X251.012 Cfg err nt *network ID*

Long Syntax: X251.012 Configuration error on network index *network ID*

Description: The X.25 network handler cannot continue initialization due to a missing datum or conflict in the network configuration. Check the node address, Virtual Circuit ranges and PVC assignments (if any).

X251.013

Level: CE-ERROR

Short Syntax: X251.013 Tx flsh cmp *network ID*

Long Syntax: X251.013 Outbound buffer flush completed by driver on network *network ID*

Description: A protocol event has required that the frame layer flush all buffers queued to the driver. It does so by issuing a flush command. The driver marks the last such buffer, which yields this message.

Chapter 137. X.25 Transport over TCP/IP (XTP)

This chapter describes X.25 Transport over TCP/IP (XTP) messages. For information on message content and how to use the message, refer to the Introduction.

XTP.002

Level: U-INFO

Short Syntax: XTP.002 q overflow, nt *network ID*

Long Syntax: XTP.002 Queue overflow on packet from net *network ID*

Description: This message is generated when there is a input queue overflow causing the forwarder to discard the packet.

Cause: Input queue overflows happen when a packet is received from an interface that is short on buffers and the length of the XTP queue is greater than the fair share. This may be caused by either a burst or steady state of traffic arriving faster than the XTP forwarder can forward it.

Action: Reduce traffic bursts. Upgrade to a faster router.

XTP.005

Level: UI-ERROR

Short Syntax: XTP.005 invalid tcp mesg type recvd

Long Syntax: XTP.005 Undefined message received on TCP

Description: The TCP message received is not among the ones defined.

XTP.006

Level: U-INFO

Short Syntax: XTP.006 called dte *dteaddr,* in call req from peer *ipaddr* not configured

Long Syntax: XTP.006 Called DTE *dteaddr,* in CALL REQUEST from peer *ipaddr* is not configured on any interface.

Description: The DTE to which a call needs to be made is not configured on any interface.

XTP.007

Level: UI-ERROR

Short Syntax: XTP.007 call req to dte *dteaddr,* from peer *ipaddr,* failed with diag *clearing_diagnostic*

Long Syntax: XTP.007 Call request to DTE *dteaddr,* from peer *ipaddr,* failed with diagnostic *clearing_diagnostic*

Description: The call request made to the DTE failed.

XTP.008

Level: U-INFO

Short Syntax: XTP.008 tcp pkt or x25 pkt has invalid cktid

Long Syntax: XTP.008 We received a tcp packet or an x.25 pkt with invalid circuit id in it.

Description: It could be just a cleaning up phase and probably would not effect anything.

XTP.009

Level: UI-ERROR

Short Syntax: XTP.009 tcp pkt alloc failed

Long Syntax: XTP.009 Memory allocation for XTP tcp packet failed

Description: Allocation of a XTP tcp packet failed.

XTP.010

Level: UI-ERROR

Short Syntax: XTP.010 unkn x25 data type recvd in tcp pkt

Long Syntax: XTP.010 X.25 data encapsulated in TCP packet is neither Q nor U data.

Description: X.25 data contained in the tcp message is invalid.

XTP.011

Level: UI-ERROR

Short Syntax: XTP.011 data xfer req from peer *ipaddr* on an inactive x25 ckt

Long Syntax: XTP.011 TCP Request from peer *ipaddr* to send data on an inactive X.25 circuit is received.

Description: Data cannot be sent over the X.25 circuit as it is not in the active state.

XTP.012

Level: UI-ERROR

Short Syntax: XTP.012 data xfer req from peer *ipaddr,* on nt *network ID* failed

Long Syntax: XTP.012 Data from peer *ipaddr,* could not be sent on net *network ID*

Description: Data could not be sent over the X.25 circuit

XTP.013

Level: UI-ERROR

Short Syntax: XTP.013 mem alloc failed in *function_name*

Long Syntax: XTP.013 Memory allocation failed in function *function_name*

Description: Memory could not be allocated.

XTP.014

Level: UI-ERROR

Short Syntax: XTP.014 *network ID*

Long Syntax: XTP.014 *network ID*

Description: XTP generic debug message.

XTP.015

Level: UI-ERROR

Short Syntax: XTP.015 call req from rmt dte *dteaddr,* on peer *ipaddr,* to local dte *dteaddr* failed

Long Syntax: XTP.015 Call request from DTE *dteaddr,* configured on peer *ipaddr,* to local DTE *dteaddr* failed

Description: X.25 Call request failed.

XTP.017

Level: U-INFO

Short Syntax: XTP.017 xtp init successful

Long Syntax: XTP.017 XTP forwarder initialization on the router is successful

Description: The XTP forwarder initialized succesfully.

XTP.019

Level: UI-ERROR

Short Syntax: XTP.019 multiple dtes configured on dce interface *interface*

Long Syntax: XTP.019 More than one DTE has been configured on DCE interface *interface*

Description: Multiple DTEs are configured on a interface of equipment type DCE.

XTP.020

Level: UI-ERROR

Short Syntax: XTP.020 internal ip addr not set

Long Syntax: XTP.020 The internal IP address has not been set in IP config

Description: The internal IP address of the router is not set.

Action: Set the internal IP address in IP config.

XTP.021

Level: U-INFO

Short Syntax: XTP.021 net dwn, nt *network ID*

Long Syntax: XTP.021 net down for net *network ID*

Description: This message is generated when the net goes down

XTP.022

Level: UI-ERROR

Short Syntax: XTP.022 pvc req for unconfigured local dte or lcn *dteaddr*

Long Syntax: XTP.022 A PVC is configured for a unconfigured local DTE or lcn *dteaddr*

Description: Configure the local DTE or lcn.

XTP.023

Level: UI-ERROR

Short Syntax: XTP.023 excess pvcs configured

Long Syntax: XTP.023 The number of PVCs configured exceeds the limit

Description: More than the defined limit of PVCs are configured.

XTP.024

Level: UI-ERROR

Short Syntax: XTP.024 recv on null tcb

Long Syntax: XTP.024 Receive posted on null tcb

Description: Receive posted on null tcb.

XTP.025

Level: UI-ERROR

Short Syntax: XTP.025 null iob recvd on tcp

Long Syntax: XTP.025 The iob received on TCP is null

Description: A null iob was received over TCP.

XTP.026

Level: UI-ERROR

Short Syntax: XTP.026 tcp post rcv failed

Long Syntax: XTP.026 The receive posted by TCP failed

Description: The receive posted by TCP to get data failed.

XTP.027

Level: UI-ERROR

Short Syntax: XTP.027 xtp_tcp hdr rcvd from peer *ipaddr,* has invalid msg size *msg_size*

Long Syntax: XTP.027 xtp_tcp header received from peer *ipaddr,* has an invalid message size *msg_size*.

Description: The message size in xtp_tcp header is invalid.

XTP.028

Level: UI-ERROR

Short Syntax: XTP.028 null apphnd in tcbp, TCP cnn to peer *ipaddr* is down

Long Syntax: XTP.028 null apphnd in tcbp, TCP cnn to peer *ipaddr* is down.

Description: The circuit ID in a TCP control block is null even though we have data for a TCP circuit. The application handle is NULL in the tcbp, probably due to the fact that the TCP connection to the associated peer router came down right after data from it arrived. This is a normal occurence if a XTP.56 message was logged just before this message.

XTP.029

Level: UI-ERROR

Short Syntax: XTP.029 tcp sess to peer *ipaddr* is reset

Long Syntax: XTP.029 The TCP session to peer *ipaddr* is being reset

Description: The TCP session is reset.

XTP.030

Level: UI-ERROR

Short Syntax: XTP.030 tcp send to peer *ipaddr* failed

Long Syntax: XTP.030 TCP send to peer *ipaddr* failed

Description: The TCP call to send data failed.

XTP.031

Level: UI-ERROR

Short Syntax: XTP.031 x25 api reg failed on int *interface*

Long Syntax: XTP.031 Registration with the X.25 service failed on interface *interface*

Description: X.25 API call to register with X.25 service failed

---

XTP.032

Level: U-INFO

Short Syntax: XTP.032 xtp listening on tcp port *port*

Long Syntax: XTP.032 TCP component of XTP did a passive open on tcp port *port*

Description: XTP TCP module successfully did a passive open.

XTP.033

Level: UI-ERROR

Short Syntax: XTP.033 xtp passive open failed on tcp port *port*

Long Syntax: XTP.033 TCP component of XTP did a passive open on port *port* which failed

Description: XTP TCP module failed doing a passive open.

XTP.034

Level: U-INFO

Short Syntax: XTP.034 xtp active open from *ipaddr*

Long Syntax: XTP.034 TCP component of XTP did a active open from *ipaddr*

Description: XTP TCP module successfully did a active open.

XTP.035

Level: UI-ERROR

Short Syntax: XTP.035 xtp tcp active open from *ipaddr* failed

Long Syntax: XTP.035 TCP component of XTP did a failed active open from *ipaddr*

Description: XTP TCP module failed doing a active open.

XTP.036

Level: U-INFO

Short Syntax: XTP.036 tcp cnn from unconfigured peer *ipaddr* not accepted

Long Syntax: XTP.036 TCP connection from unconfigured peer *ipaddr* not accepted

Description: Router from which a connection request is received is not configured as a peer router.

XTP.037

Level: UI-ERROR

Short Syntax: XTP.037 rtr with greater IP addr *ipaddr* - an error

Long Syntax: XTP.037 Some how an router with greater IP address does an active open, against the design. src *ipaddr*

Description: An active open from a greater ipaddr was received at a lower ipaddr which is an error.

XTP.038

Level: U-INFO

Short Syntax: XTP.038 closing prev tcp cnn to peer *ipaddr*

Long Syntax: XTP.038 Closing previously opened TCP connection to *ipaddr*

Description: A previously opened TCP connection is closed.

XTP.039

Level: U-INFO

Short Syntax: XTP.039 tcp cnn estab from *ipaddr*

Long Syntax: XTP.039 TCP connection established between *ipaddr*

Description: TCP connection is established between peer routers.

XTP.041

Level: U-INFO

Short Syntax: XTP.041 tcp msg *msg_type,* sent to router *ipaddr,* which has *unss* messages queued

Long Syntax: XTP.041 TCP packet containing message *msg_type,* is sent to router *ipaddr,* which has *unss* messages queued.

Description: TCP message is being sent, just a LOG.

XTP.042

Level: U-INFO

Short Syntax: XTP.042 tcp msg *Network ID*

Long Syntax: XTP.042 TCP packet containing message *Network ID*

Description: TCP message has been received, just a LOG.

XTP.043

Level: UI-ERROR

Short Syntax: XTP.043 tcp cnn to *Network ID*

Long Syntax: XTP.043 TCP connection to *Network ID* has reached limit.

Description: As one TCP connection carries multiple SVCs and if data is comes too fast on SVCs TCP cnn is unable to handle so much traffic.

XTP.044

Level: U-INFO

Short Syntax: XTP.044 peer *ipaddr* closed tcp cnn

Long Syntax: XTP.044 Remote host *ipaddr* has closed the TCP connection.

Description: Connection has been closed, do the clean up.

XTP.045

Level: U-INFO

Short Syntax: XTP.045 appln handle in tcbp retn from tcp NULL

Long Syntax: XTP.045 In a TCP upcall we found that application handle for a TCP connection is NULL

Description: This may not be a terrible error

XTP.046

Level: UI-ERROR

Short Syntax: XTP.046 invalid message size *msg_size* requested for transport on TCP

Long Syntax: XTP.046 The message size *msg_size* requested by X.25 for transport on TCP is invalid.

Description: Check the message sizes in X.25.

XTP.047

Level: UI-ERROR

Short Syntax: XTP.047 request to send on a non-established TCP connection

Long Syntax: XTP.047 You got a request to forward either x25 call req, x25 data on a non-established TCP connection.

Description: Possible misconfiguration in XTP or X.25

XTP.048

Level: UI-ERROR

Short Syntax: XTP.048 invalid tcp cnn, src and dst same - *ipaddr*

Long Syntax: XTP.048 Peer IP address same as the local IP address - *ipaddr*.

Description: Check the configuration of IP address under XTP.

XTP.050

Level: UI-ERROR

Short Syntax: XTP.050 pvc with lcn *lcn* not configured in x25

Long Syntax: XTP.050 PVC with LCN *lcn* is configured in XTP but not in X.25

Description: Configure the PVC in X.25 also.

XTP.051

Level: UI-ERROR

Short Syntax: XTP.051 xtp init not successful on nt *network ID*

Long Syntax: XTP.051 XTP is not successfully initialized on net *network ID*

Description: Interface in question can be down or not of X25 type or

---

XTP.052

Level: U-INFO

Short Syntax: XTP.052 call req for call *dteaddr,--> dteaddr,* cannot be fwd to peer *ipaddr*, no TCP cnn

Long Syntax: XTP.052 An X.25 Call Request from calling DTE *dteaddr,* to called DTE *dteaddr,* cannot be forwarded to peer *ipaddr* since the TCP connection to the peer is not active.

Description: The connection to the called DTE cannot be established throuth the peer router at this time since the TCP connection to the peer is currently inactive.

XTP.053

Level: U-INFO

Short Syntax: XTP.053 call req for call *dteaddr,--> dteaddr,* redrive to peer *ipaddr*

Long Syntax: XTP.053 An X.25 Call Request from calling DTE *dteaddr,* to called DTE *dteaddr,* is being re-driven to peer router *ipaddr*.

Description: The connection to the called DTE is being attempted through the peer router since a previous attempt through a different peer failed.

XTP.054

Level: U-INFO

Short Syntax: XTP.054 no other peer for call req redrive for call *dteaddr,--> dteaddr*

Long Syntax: XTP.054 There are no other available peer routers to attempt a redrive of the X.25 Call Request from calling DTE *dteaddr,* to called DTE *dteaddr*.

Description: The connection to the called DTE cannot be attempted through another peer router.

XTP.055

Level: U-INFO

Short Syntax: XTP.055 call req timeout for call *dteaddr,--> dteaddr,* via peer *ipaddr*

Long Syntax: XTP.055 An X.25 Call Request from calling DTE *dteaddr,* to called DTE *dteaddr,* through peer router *ipaddr* timed out.

Description: The connection to the called DTE that was being attempted through the peer router was never responded to within the connection time out value.

XTP.056

Level: U-INFO

Short Syntax: XTP.056 tcp cnn to peer *ipaddr* has closed

Long Syntax: XTP.056 The TCP connection to remote host *ipaddr* has closed.

Description: The TCP connection to remote host has been closed.

XTP.057

Level: U-INFO

Short Syntax: XTP.057 tcp cnn to peer *ipaddr* closed - no keepalives

Long Syntax: XTP.057 The TCP connection to remote host *ipaddr* has been closed due to a lack of received Keepalive messages.

Description: The TCP connection to the remote host has been closed because this router is no longer receiving X.25 Transport Keepalive messages from it.

XTP.058

Level: U-INFO

Short Syntax: XTP.058 unsupported tcp msg *Network ID*, message dropped

Long Syntax: XTP.058 An unsupported TCP packet containing message *Network ID*, clear sent

Description: An unsupported TCP message has been received. There is an incompatibility between this router and the router that sent the TCP message.

XTP.059

Level: UI-ERROR

Short Syntax: XTP.059 xtp forcing tcp cnn to *Network ID*

Long Syntax: XTP.059 XTP connection to *Network ID*, has reached limit.

Description: As one XTP connection carries multiple SVCs and if data comes in faster than X.25 can transmit, XTP must flow control the TCP/IP connection.

XTP.060

Level: UI-ERROR

Short Syntax: XTP.060 XTP exiting flow control on tcp cnn to *Network ID*

Long Syntax: XTP.060 XTP connection to *Network ID*.

Description: As one XTP connection carries multiple SVCs and if data comes in faster than X.25 can transmit, XTP must flow control the TCP/IP connection and XTP can now accept data.

Readers' Comments — We'd Like to Hear from You

Nways
Event Logging System Messages Guide

Publication No. SC30-3682-13

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name

Address

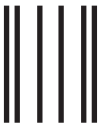
Company or Organization

Phone No.

Readers' Comments — We'd Like to Hear from You
SC30-3682-13

Cut or Fold
Along Line

Fold and Tape | **Please do not staple** | Fold and Tape



NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Corporation
Design & Information Development
Department CGF/Bldg. 656
PO Box 12195
Research Triangle Park, NC 27709-9990

Fold and Tape | **Please do not staple** | Fold and Tape

Cut or Fold
Along Line

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SC30-3682-13