Network Utility

Installation, Getting Started, and User's Guide

GA27-4167-02

Network Utility

Installation, Getting Started, and User’s Guide

GA27-4167-02

Note

Before using this information and the product it supports, be sure to read the general information under "Appendix A. Notices" on page 357 and safety information in "Appendix B. Safety Information" on page 361.

Third Edition (June 1999)

This edition applies to the Network Utility Models TN1 and TX1, and to Multiprotocol Access Services (MAS) V3.3.

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

A form for readers' comments appears at the back of this publication. If the form has been removed, address your comments to:

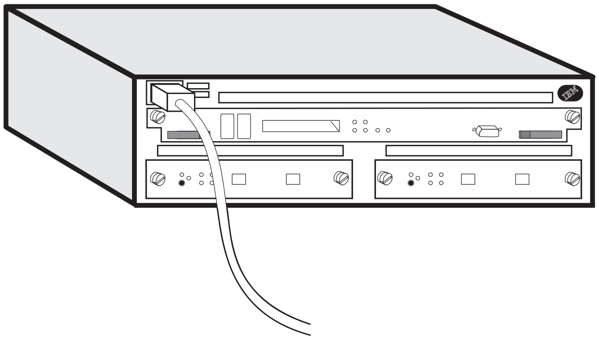
Department CGF
Design & Information Development
IBM Corporation
PO Box 12195
RESEARCH TRIANGLE PARK NC 27709-9990
USA

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

About This Guide

This guide explains how to set up the IBM Network Utility, perform initial configuration, correct problems that might occur during installation, and use the Network Utility. It also contains detailed configuration examples for some common Network Utility network configurations.



There are two models of the IBM Network Utility: the Network Utility TN3270E Server (Model TN1) and the Network Utility Transport (Model TX1). Unless explicitly stated, the term *Network Utility* applies to both the Model TN1 and the Model TX1.

This guide is a part of the documentation for Network Utility that is described in “Library Overview” on page xii. This guide helps you get started with the more detailed reference information that is documented in the other books.

Who Should Use This Guide

This guide is intended to be used by the person responsible for installing, configuring, and managing the Network Utility.

How to Proceed

Installation and Initial Configuration

1. Install the chassis and the cables (see Chapter 1) using the *Installation and Initial Configuration Guide* provided with the product. (Alternatively, installation by IBM service personnel is available. Contact your IBM representative for additional information.)

   Note: Installation of the cables for the Parallel Channel Adapter (FC 2299) requires IBM service or channel-trained personnel.
2. Connect a terminal or workstation to be able to configure and operate the product (see Chapter 2) using either the serial port on the system card for a local connection, or connect a phone line to the PCMCIA Modem that is plugged into the system card for remote connection.
3. Decide which configuration method that you want to use and perform an initial configuration of the 2216 Model 400 or Network Utility Network Utility (see Chapter 3).

Learning

- If you already have some experience with the command-line interface of IBM routing products or if you prefer to try tasks without following a tutorial, use Chapter 4 to review some of the basics of navigating the command-line interface. Scan the other chapters in Part 2. Learning About Network Utility, so that you know where to find additional information that you may need.

  If the command-line interface of IBM routing products is new to you, use Chapter 5 as a tutorial to learn about basic concepts and navigation.
- If you are familiar with basic configuration and operation functions, select from the configuration scenarios that we provide in Part 3. Configuration and Management Specifics. Select a configuration that resembles your network characteristics:
  - Model TN1 users—see “Chapter 12. TN3270E Server” on page 127.
  - Model TX1 users—see “Chapter 14. Channel Gateway” on page 203, “Chapter 16. Data Link Switching” on page 239 or “Chapter 19. Virtual Private Networks” on page 275.
  - All users—see “Chapter 18. Sample Host Definitions” on page 259 if your configuration involves IBM host networking products.

Final Configuration and Operation

1. Use the operations and management tasks that are introduced in Part 2. Learning About Network Utility and the scenarios that are documented in Part 3. Configuration and Management Specifics to debug and complete your initial configuration.
2. Perform final configuration. Refer to the *Configuration Program User’s Guide* and *Software User’s Guide*.

Library Overview

The Network Utility and the IBM 2216 Model 400 share many of the same publications. The following figure shows the publications in the library, arranged according to tasks.

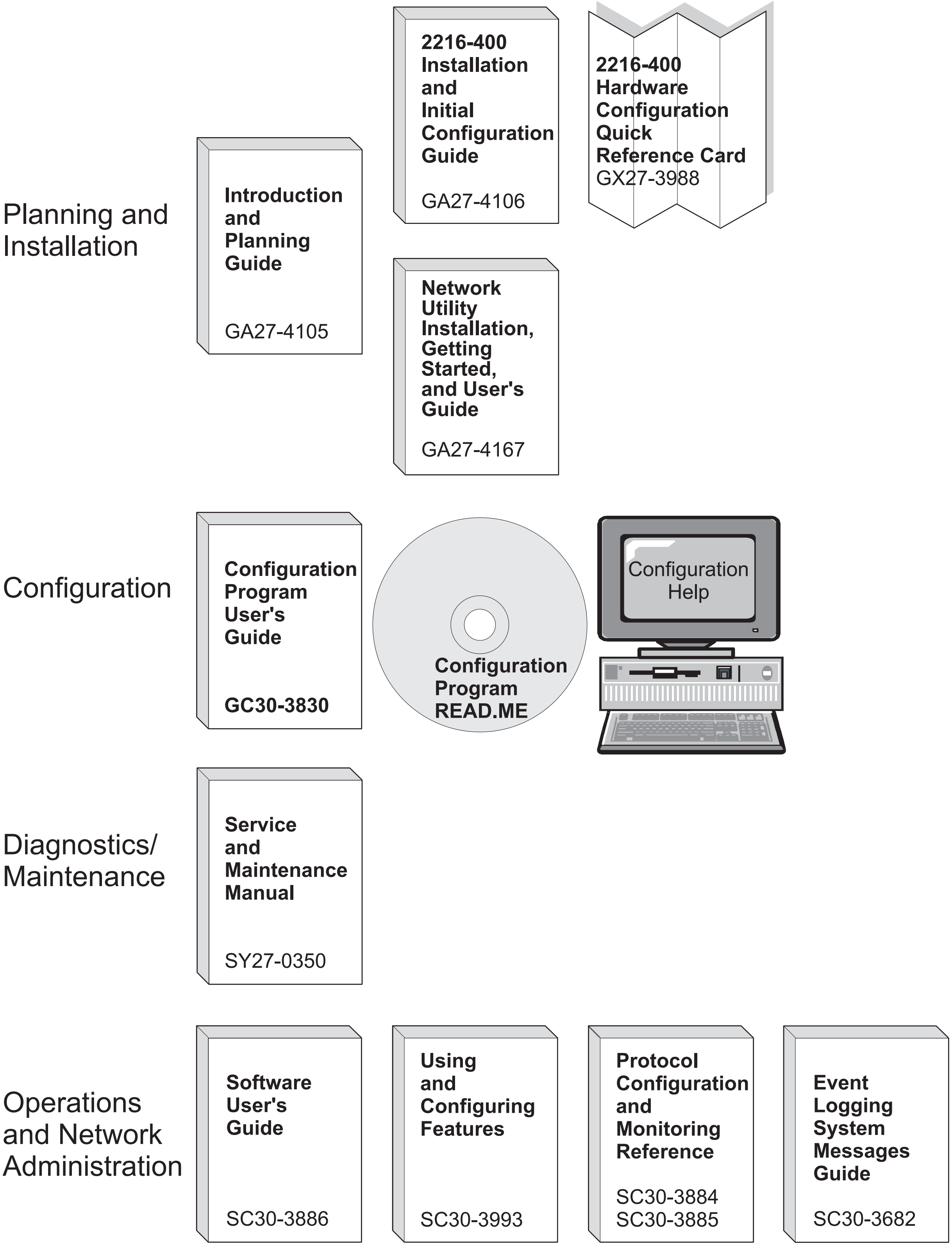

Planning and Installation
Introduction and Planning Guide
GA27-4105
2216-400 Installation and Initial Configuration Guide
GA27-4106
2216-400 Hardware Configuration Quick Reference Card
GX27-3988
Network Utility Installation, Getting Started, and User's Guide
GA27-4167
Configuration
Configuration Program User's Guide
GC30-3830
Configuration Program READ.ME
Configuration Help
Diagnostics/ Maintenance
Service and Maintenance Manual
SY27-0350
Operations and Network Administration
Software User's Guide
SC30-3886
Using and Configuring Features
SC30-3993
Protocol Configuration and Monitoring Reference
SC30-3884
SC30-3885
Event Logging System Messages Guide
SC30-3682

Figure 1. Common Tasks and the Library for the IBM 2216 Model 400 and Network Utility

Hardcopy Publications That are Shipped with the Product

These documents are shipped in hardcopy and are also contained on this product's Documentation CD-ROM, SK2T-0405.

Planning

GA27-4105

2216 Nways Multiaccess Connector and Network Utility Introduction and Planning Guide

This book explains how to prepare for installation and select the hardware that you want to purchase. It includes specifications for the hardware and software for your network. It also provides information on the management of routing networks.

Installation and Learning

GA27-4167

Network Utility only:

Network Utility Installation, Getting Started, and User's Guide

This book explains how to install a Network Utility and verify its installation. In addition it explains how to use the product and has sample configurations for the product.

GA27-4106

2216 Model 400 only:

2216 Nways Multiaccess Connector Model 400 Installation and Initial Configuration Guide

This booklet explains how to install the 2216 Model 400 and verify its installation.

GX27-3988

2216 Model 400 only:

2216 Nways Multiaccess Connector Hardware Configuration Quick Reference

This reference card is used for entering and saving hardware configuration information used to determine the correct state of an IBM 2216 Model 400.

Diagnostics and Maintenance

SY27-0350

2216 Nways Multiaccess Connector and Network Utility Service and Maintenance Manual

This book provides instructions for diagnosing problems with and repairing the Model 400 and the Network Utility.

Safety

SD21-0030

Caution: Safety Information—Read This First

This book provides translations of caution and danger notices applicable to the installation and maintenance of a device.

Configuration

GC30-3830

Configuration Program User’s Guide

This book discusses how to use the Nways Multiprotocol Access Services Configuration Program.

Publications Shipped as Softcopy on the CD-ROM

These publications are also separately orderable as hardcopy.

Operations and Network Management

SC30-3886

Nways Multiprotocol Access Services Software User’s Guide

This book explains how to:

- Configure, monitor, and use the Nways Multiprotocol Access Services software and microcode.
- Use the Nways Multiprotocol Access Services command-line router user interface to configure and monitor the network interfaces and link-layer protocols shipped with the 2216 base.

SC30-3993

Nways Multiprotocol Access Services Using and Configuring Features

This book describes Multiprotocol Access Services (MAS)features and explains the commands to use them. Features are any functions that enhance protocols or stand alone. Some examples are: MAC Filtering, which filters frames based on their MAC addresses, Bandwidth Reservation System, which enables you to reserve bandwidth for chosen types of traffic over a PPP or Frame Relay serial interface, or Network Address Translation, which enables you to represent one IP address with another when you are running IP.

SC30-3884

Nways Multiprotocol Access Services Protocol Configuration and Monitoring Reference, Volume 1

SC30-3885

Nways Multiprotocol Access Services Protocol Configuration and Monitoring Reference, Volume 2

These books describe how to access and use the Nways Multiprotocol Access Services command-line user interface to configure and monitor the routing protocol software shipped with the product.

They include information about each of the protocols that the device supports.

SC30-3682

Nways Event Logging System Messages Guide

This book contains a listing of the error codes that can occur, along with descriptions and recommended actions to correct the errors.

Ordering IBM Publications

In the U.S.A., you can order IBM publications by calling 1-800-879-2755. Within or outside the U.S.A., you can order IBM publications through the IBM Publications Direct Catalog on the World Wide Web at:

`http://www.elink.ibmlink.ibm.com/pbl/pbl`

IBM translates many publications into a variety of languages. The publications you need may be available in your language.

Visit our Web Sites

These IBM web pages provide product information:

For the Network Utility: `http://www.networking.ibm.com/networkutility`

For the Model 400: `http://www.networking.ibm.com/216/216prod.html`

This IBM web page provides 2216 base and Network Utility books online:

`http://www.networking.ibm.com/did/2216bks.html`

Information, Updates, and Corrections

This page provides information on engineering changes, clarifications, and fixes that were implemented after the books were printed:

`http://www.networking.ibm.com/216/216changes.html`

Product Support

These pages provide downloads and additional support information:

For the Network Utility:
`http://www.networking.ibm.com/support/networkutility`

For the Model 400: `http://www.networking.ibm.com/support/2216`

Part 1. Getting Started

Chapter 1. Setting Up the Hardware

This chapter covers the following topics:

- Defining what you need to install and configure the Network Utility
- Rack-mounting or surface-mounting the Network Utility chassis
- Inserting PCMCIA cards
- Powering on the Network Utility for the first time
- Verifying that the LEDs show a healthy system

Installing the Network Utility

Before You Begin: The illustrations assume that all of the adapter slots are filled. A fully populated Network Utility weighs about 15 kg (33 lb).

Pre-installation Requirements—You need to provide:

- An ASCII terminal or a workstation (PC)
- For the workstation, either Telnet client or ASCII terminal emulation software (for example, ProComm)
- If you are dialing into the Network Utility PCMCIA modem, a modem for your remote workstation
- If you will transfer configuration files or code into Network Utility (other than via Xmodem), a LAN adapter for your workstation
- If you will use the Network Utility PCMCIA EtherJet card, a small Ethernet hub or a cross-over cable to directly attach an Ethernet-capable workstation

Rack-Mounting Requirements—You can use any EIA standard 19-inch rack. The rack can be open or closed. However, if you choose a closed rack, you must make sure that enough air flows through the Network Utility. Covers on the front of the rack that would not let air reach the Network Utility must be removed or modified to let air pass. Similarly, unvented rear rack covers that would not let air exit the Network Utility or would cause back pressure to build up from several machines must not be used.

1. Verify contents

Verify that the following items were included with your Network Utility.

Documentation

In addition to this document, the package should include the following publications:

- *Caution: Safety Information–Read This First*, SD21-0030
- *2216 Nways Multiaccess Connector and Network Utility Introduction and Planning Guide*, GA27-4105
- *2216 Nways Multiaccess Connector and Network Utility Service and Maintenance Manual*, SY27-0350
- *Configuration Program User’s Guide*, GC30-3830
- *2216 Documentation CD-ROM*, SK2T-0405

Hardware

- Network Utility with the adapters already installed
- Any cables that were ordered
- Rack-mount installation aid
- Power cord
- PCMCIA modem (except in countries where the PCMCIA modem is not available)
- IBM EtherJet PC card
- Rack-mounting cable bracket if the Network Utility contains FC 2299 (Parallel Channel Adapter)
- Null modem and two 9-to-25-pin serial communication cables

Software

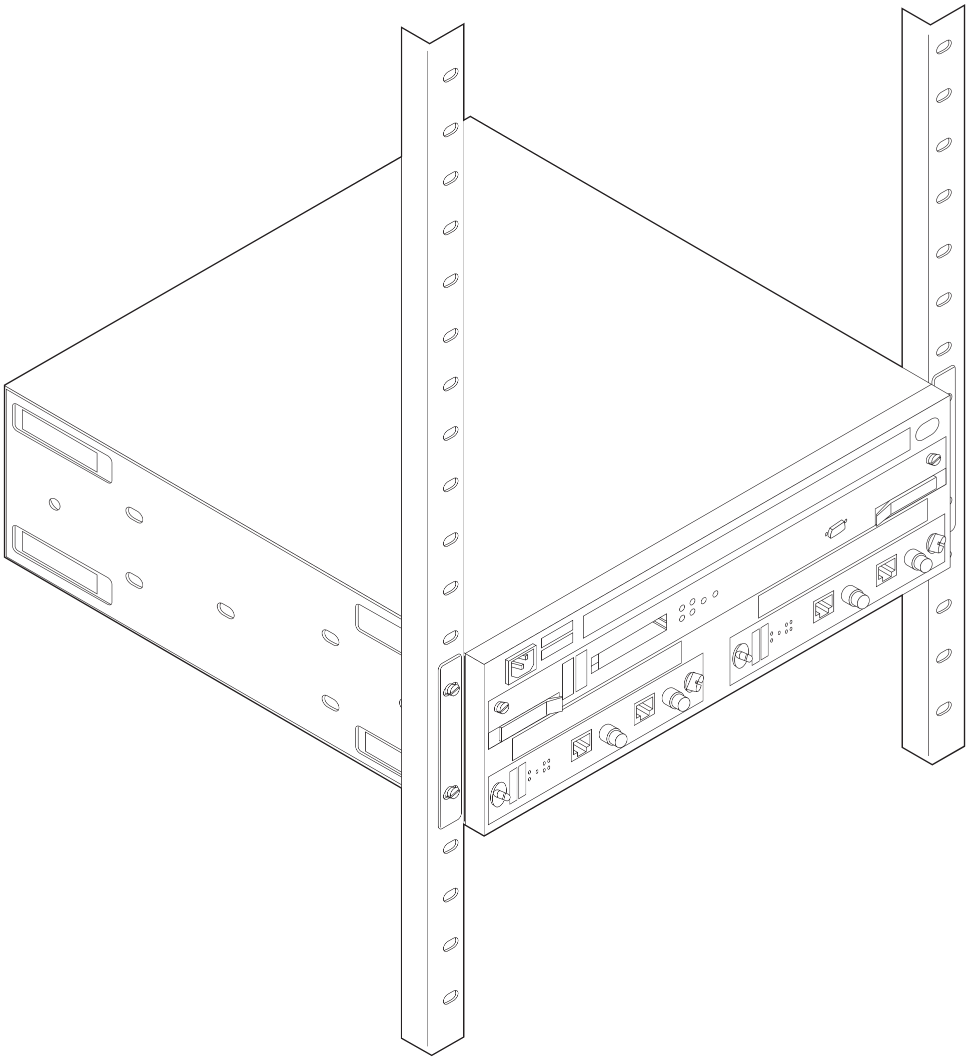
- IBM 2216 Model 400 and Network Utility Configuration Program CD-ROM
- The operating code is preloaded onto the Network Utility

Proceed with

Surface-mounting - go to step 7 on page 9.

Rack-mounting - go to step 2 on page 5.

2. Rack-Mounting the Network Utility



You need the following items:

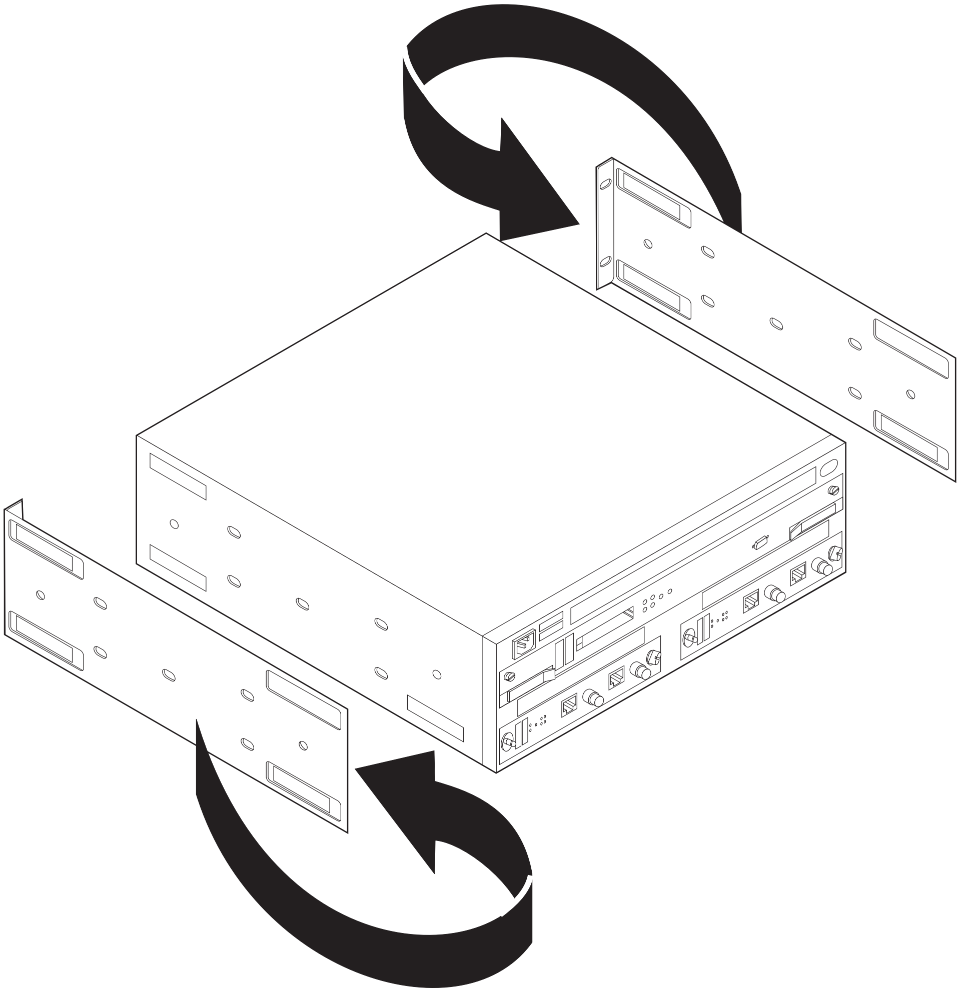
- Cables, as required
- Four rack-mounting screws
- Screwdriver

Notes:

1. If you have a shelf for the rack, install it before continuing.
2. Do not use the installation aid if you have a shelf installed.

Continue with step 3 on page 6.

3. Rack-Mounting (Optional for Surface-Mounting)

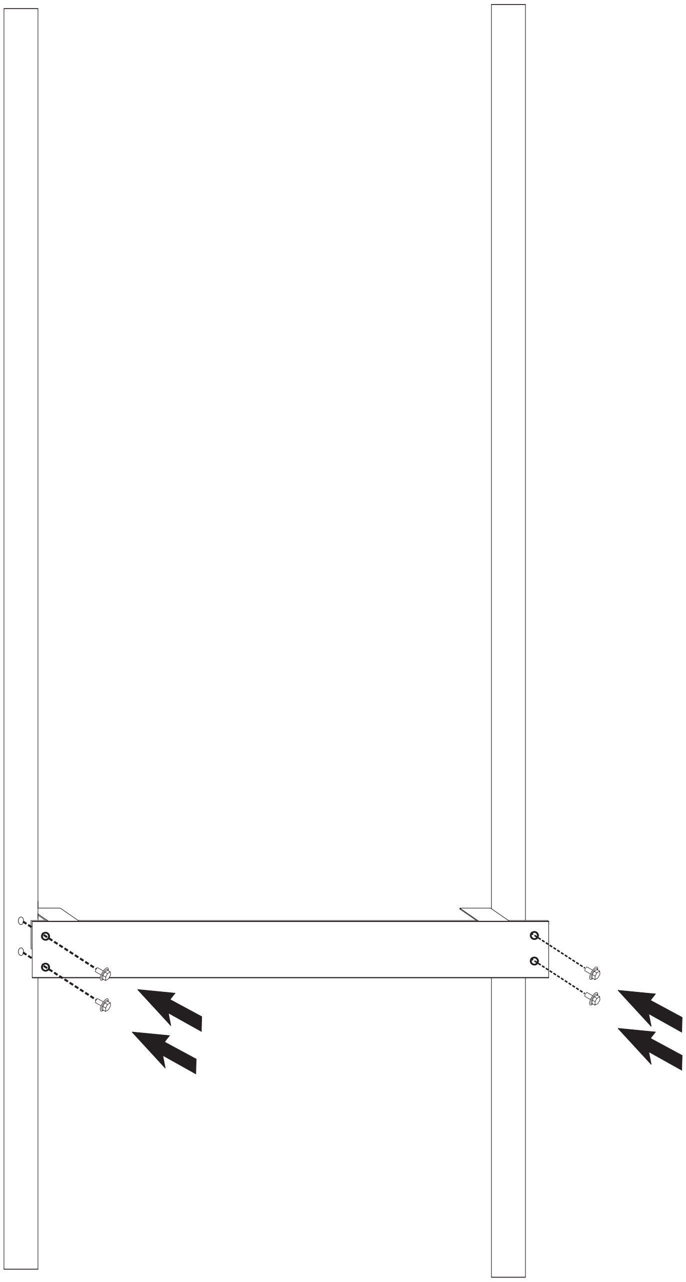


The Network Utility mounting brackets are shipped with the flanges facing the rear:

1. Remove the two screws from each bracket (one at the front and one at the rear).
2. Reverse each bracket so that the Network Utility can be rack-mounted.
3. Reinstall the four screws.

When the brackets are fitted correctly, the letter embossed on each bracket is on the rear edge; an A on the right side and a B on the left side.

4. Rack-Mounting



The installation aid is a metal bar that supports the Network Utility as you install it in the rack. The installation aid ensures that the Network Utility and rack are lined up correctly.

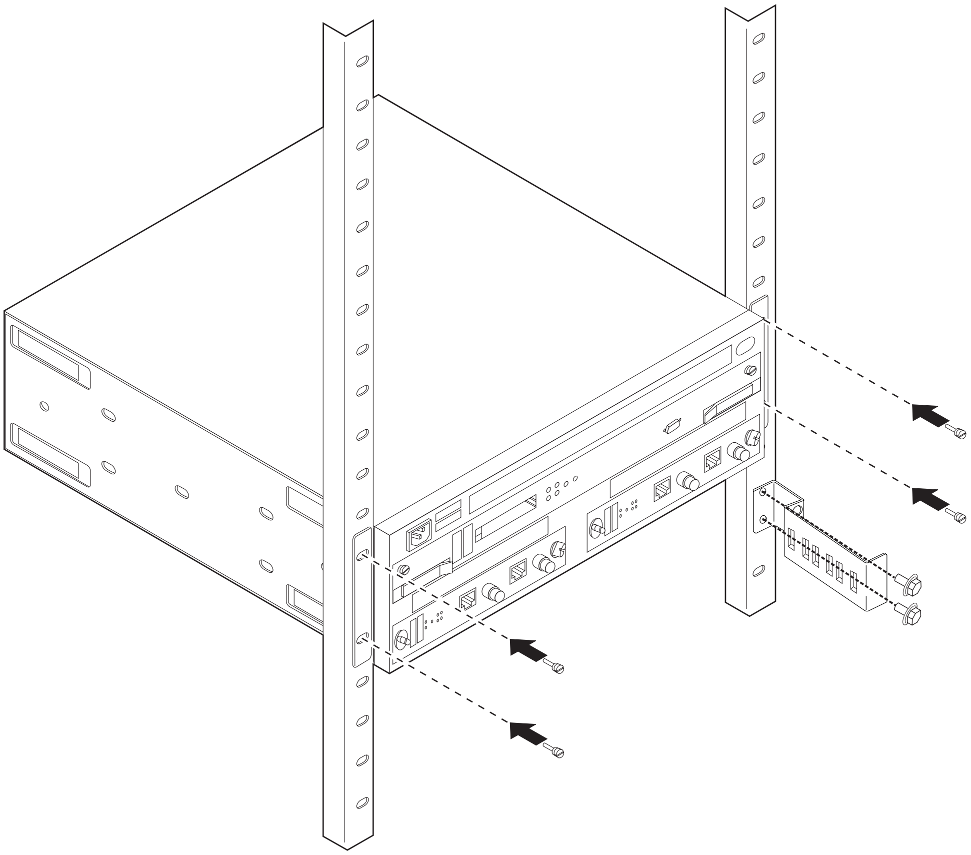
Line up the holes in the installation aid with the rack and install all screws.

5. Rack-Mounting

Set the Network Utility on the IBM 2216 installation aid or on the shelf. The mounting brackets keep the Network Utility from falling into the rack during installation.

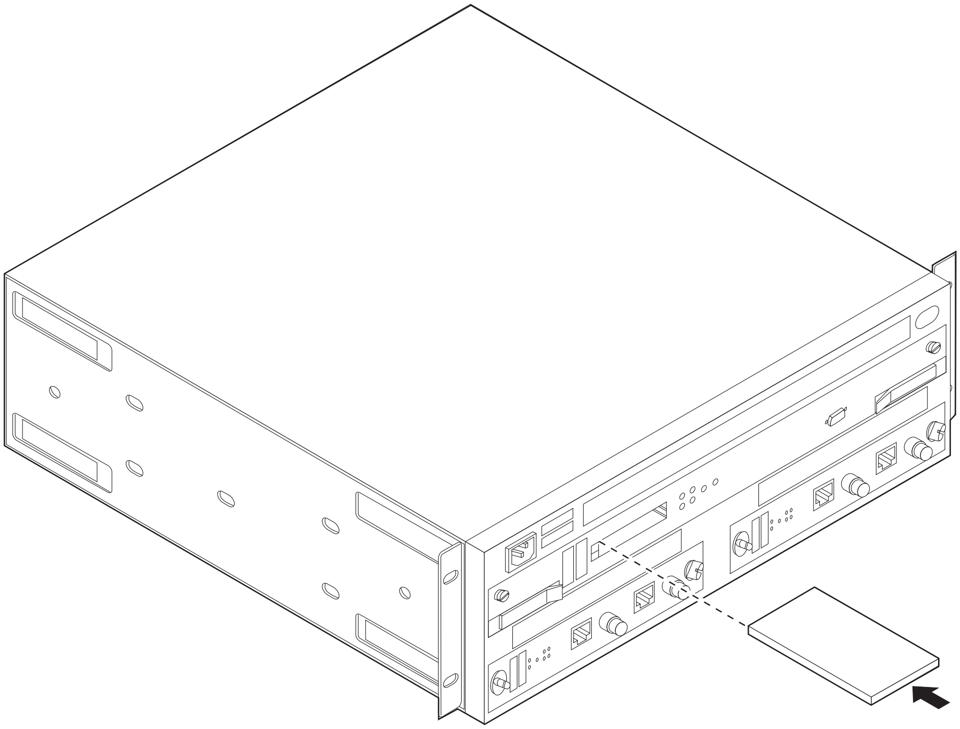
With the installation aid installed, steady the Network Utility while you complete the next step.

6. Rack-Mounting



1. Install the screws beginning with the lower screws.
2. For FC 2299: Using 2 screws, install the rack-mounting cable bracket onto the front of the rack below the Network Utility.

7. Rack- or Surface-Mounting

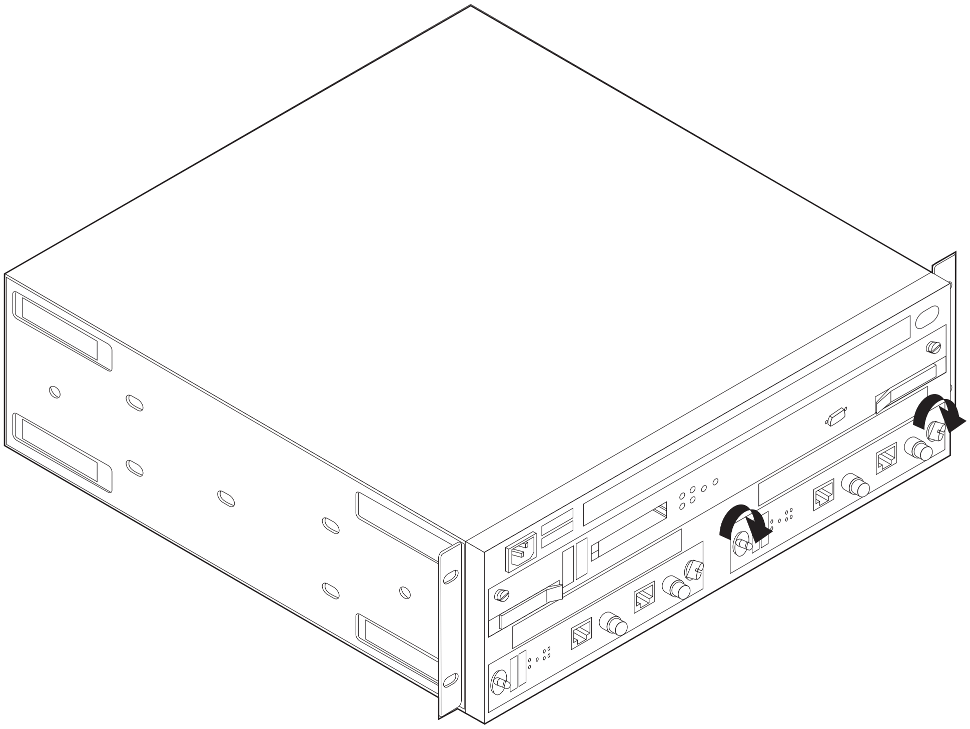


If you are installing a PCMCIA modem or PCMCIA EtherJet LAN adapter, slide it into either of the PCMCIA slots on the system card. Attach the telephone cable to the modem (a triangle identifies the left side of the cable).

Notes:

1. You cannot substitute a different Ethernet PCMCIA card for the EtherJet LAN adapter that is shipped with the Network Utility.
2. The system will not boot up if you install two PCMCIA modems or two PCMCIA Ethernet adapters into a Network Utility.

8. Rack- or Surface-Mounting



1. Verify that ***all*** thumbscrews are tight (even if you did not loosen them during installation).
2. Connect the power cord to the Network Utility and the power outlet (to power on the unit). After about 4 to 5 minutes, verify that the correct LEDs are on (see Table 3 on page 11). Monitor the LED states as shown in Figure 2 on page 12.

   While the unit is booting and the adapters are being tested, it is normal for:

   - Both the green and yellow System Card LEDs to be on for a short period of time.
   - Both the green and yellow Adapter Card LEDs to be on for a short period of time.
   - The Hard Drive and the adapter Wrong Slot yellow LEDs to be on for a short period of time.
3. If you see a problem, use the tables and procedures in “Problem Solving” on page 13 to resolve or report the problem.

9. Complete the Setup (Rack- or Surface-Mounting)

1. Connect the cables (except for the Parallel Channel Adapter, FC 2299).
   Note: If you have FC 2299, the installation of the cables requires a channel-trained IBM service representative or a customer’s channel-trained person.

   Call the IBM service representative to install the cables for FC 2299. The Parallel Channel and its attached devices will be disrupted if the cables are not installed correctly.
2. Proceed with “Chapter 2. Bringing Up a User Console” on page 15 to set up a user terminal console.

10. IBM Service Representative Tasks for FC 2299

1. Connect the adapter cables to FC 2299 (using the procedures in the *Service and Maintenance Manual* under “Installing Channel Adapters”). Do not connect to the host channel cables yet.
2. Run wrap tests to verify that all adapter cables are OK.
3. Connect the host channel cables to the adapter cables.

Verifying the Hardware Setup

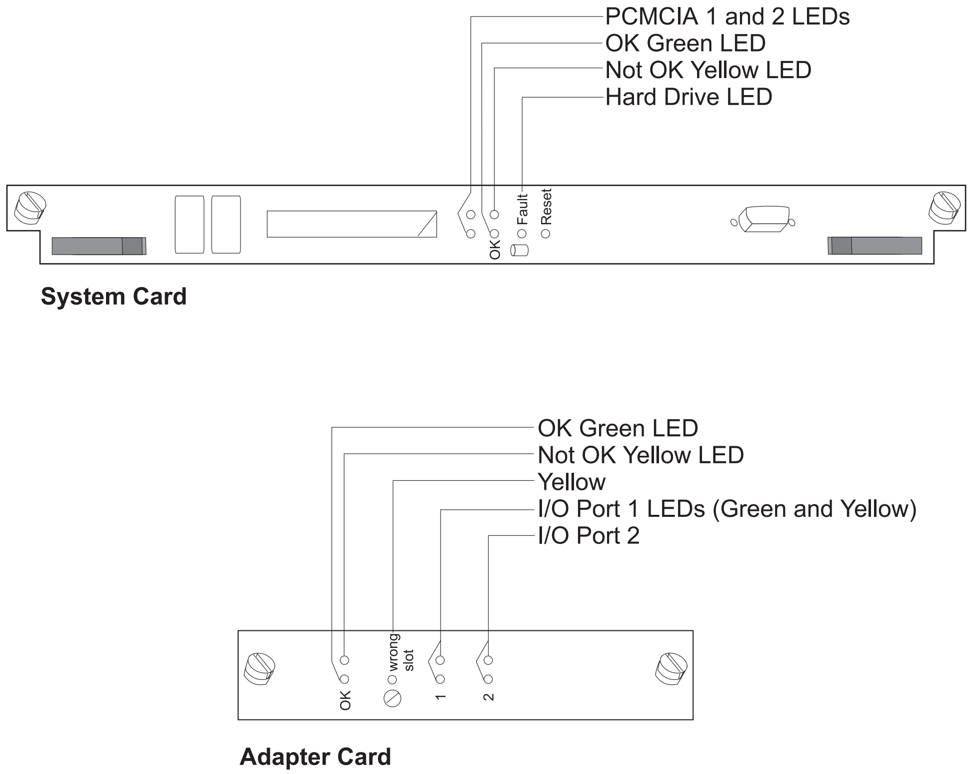
Table 3 shows the correct state of each LED on the front of the unit after it has completed booting (**about 4-5 minutes after a power-on)**. If all LEDs are in the correct state, you can begin to configure the unit. See Figure 2 on page 12 for the locations of the LEDs on the Network Utility.

Table 3. Machine LED States When Operational

CARD	LED Name	Color	State
System card	PCMCIA 1 (with device installed)	Yellow	OFF
	PCMCIA 2 (with device installed)	Yellow	OFF
	OK	Green	ON
	not OK	Yellow	OFF
For all adapter cards	OK	Green	ON
	not OK	Yellow	OFF
	Wrong slot	Yellow	OFF
	I/O port (before the configuration is loaded on the unit)	Green	OFF
	I/O port	Yellow	OFF

LED Indicators

The Network Utility has a number of light-emitting diodes (LEDs) that indicate how the unit is functioning.



System Card

Adapter Card

Figure 2. System Card and Adapter Card LEDs

System Card Status

LEDs	Meaning
PCMCIA 1 or PCMCIA 2 (Yellow)	On - PCMCIA device has a fault, is not installed, or is not seated correctly. Off - Device passed self-tests
OK (Green)	On - Card hardware is operating normally. Blinking - Loading from hard file
(Yellow)	On - Card hardware has a fault.
Fault Hard Drive (Yellow)	On - Hard drive has failed.

Adapter Card Status

LEDs	Meaning
OK (Green)	On - Adapter is operational.
(Yellow)	On - Adapter has a fault.
Wrong Slot (Yellow)	On - Contact your service representative.

LEDs	Meaning
Green port[1]	On - Port is operating normally (enabled and configured). Off - Port is not configured or is disabled. Blinking (for ESCON adapter only) - The optical power measurement test is running.
Yellow port[1]	On - One or more ports has a hardware fault. Blinking - One or more ports has a port I/O or network failure. Use the Maintenance Analysis Procedures (MAPs) in the *2216 Nways Multiaccess Connector and Network Utility Service and Maintenance Manual* to isolate. Off - No problem detected.

Important Phone Numbers

Contact Name	Telephone Number
System Administrator:	
Service Representative:	

Problem Solving

To identify and correct any problems that occur during setup, answer the questions and take the appropriate actions, as indicated:

On the system card, is the Not OK yellow LED on?

Yes: There is a fault in the card.

1. Disconnect the system from its power source.
2. Reseat the card.
3. Reconnect the system to its power source.
4. Wait 4 to 5 minutes, and verify the state of the LEDs.

If the problem is not corrected, contact your service representative.

No: Go to the next question.

On the system card, is the OK green LED off?

Yes: The green LED is switched on by the operational code.

If the green LED fails to come on, contact your service representative.

No: Go to the next question.

On the system card, is the PCMCIA port LED on?

Yes: Either the PCMCIA card slot is empty or the card failed the power-on self-test. Reseat the card.

If the problem is not corrected, contact your service representative.

No: Go to the next question.

[1] The port LEDs of the multiport WAN adapters (FC 2282, FC 2290, and FC 2291) reflect the status of one or more of the ports.

On I/O cards in slots 1 and 2, are the Not OK yellow LEDs On?

Yes: There is a fault in the card. Reseat the adapter.

If the problem is not corrected, contact your service representative.

No: Go to next question.

On I/O cards in slots 1 and 2, are the OK green LEDs On?

Yes: The Network Utility appears to be OK.

No: Reseat the card. If the OK green LED still fails to come on, the card is bad. Contact your service representative.

Chapter 2. Bringing Up a User Console

You must set up a terminal to access the Network Utility for configuration and operation. The information in this chapter helps you:

- Learn about the ways you can set up a terminal
- Choose the best method for your environment
- Attach and activate the terminal using default settings

When you are finished with this chapter, you should have an active terminal and it should be at the initial command prompt ready for configuration.

Access Methods

You can access and connect to the Network Utility in several ways that are summarized in Table 4.

Table 4. User Console Attachment Options

Physical Attachment	Line Protocol	Access Protocol	Default IP Addresses
Service port + null modem Service port + external modem PCMCIA modem	Asynchronous characters	ASCII Terminal emulation	Not Applicable
	SLIP	Telnet	Network Utility = 10.1.1.2 Workstation = 10.1.1.3
PCMCIA EtherJet	IP	Telnet	Network Utility = 10.1.0.2 Workstation = 10.1.0.3
Any IP network interface	IP	Telnet	No defaults

Make the physical connections in one of the following ways when you want to use:

1. An **ASCII terminal** or a **workstation** that is running terminal emulation software:
   - Local connection through a null-modem cable attached to the EIA 232 service port (see Figure 3 on page 16). This type of connection uses the null-modem adapter and the two 9-to-25 pin serial cables that are supplied with this product.
   - Remote dial-in (using telephone lines) through the PCMCIA modem (see Figure 4 on page 16).
   - Remote dial-in (using telephone lines) with an external modem (not pictured) attached to the EIA 232 service port. This configuration would be used in countries where there is no approved PCMCIA modem. Use an asynchronous modem that supports the Hayes AT command set. To determine which modems are supported, refer to the product literature sales pages at: `http://www.networking.ibm.com/networkutility`.
2. The **Telnet protocol** on a workstation that is running TCP/IP software:
   - Any of the physical connections that are described in the methods in alternative 1.

     For these physical connections, the Telnet workstation is running TCP/IP software that supports the Serial Line Internet Protocol (SLIP). SLIP is a method for sending IP packets across asynchronous lines.

Telnet over SLIP provides access only to the operational code command-line interface, and not to the firmware menu interface.

- Local cable from a Network Utility PCMCIA LAN adapter (an IBM EtherJet PC card) to a workstation using a local Ethernet hub. Figure 5 on page 17 shows a version of this configuration.

  The workstation Ethernet adapter could also be directly attached to the EtherJet card via a crossover cable, or there could be a wide area network between the Ethernet LAN and the Telnet workstation.

  The Network Utility IBM EtherJet PC card is for service and operations purposes, such as providing a user console and transferring files. It cannot be used as a normal network routing interface.

- A network-connected workstation that is attached to any IP-capable network interfaces of the adapters that are in the adapter slots.

  This configuration is not pictured. The network interface could be on a LAN adapter such as Token-Ring, 10/100-Mbps Ethernet, or FDDI. It could also be on any other adapter, because all of them support IP routing. The Telnet workstation could be locally or remotely connected.

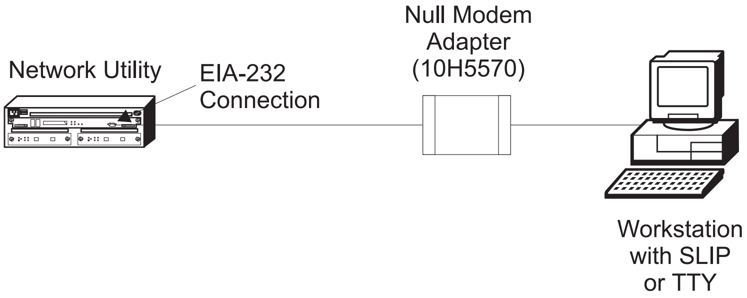


Figure 3. Local Workstation Serial Connection to the EIA 232 Port

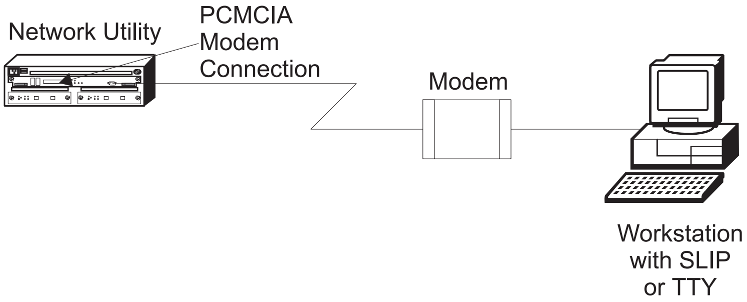


Figure 4. Remote Serial Connection to the PCMCIA Modem

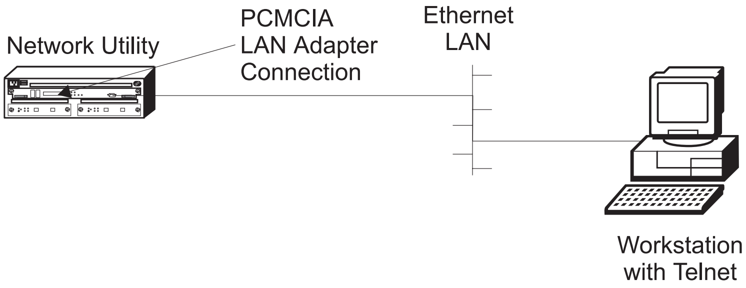


Figure 5. LAN Connection through the PCMCIA LAN Adapter

Which Access Method Should I Use?

- **If you are a new user and are physically adjacent to the Network Utility**, attach a workstation directly to the unit (see Figure 3 on page 16) using ASCII terminal emulation for your terminal console (see “ASCII Terminal Setup and Usage”). The key advantages of this method are:
  - It provides easy setup.
  - It works well with basic terminal emulation software.
  - It does not require the unit to be configured.
  - It provides a steady connection if you repeatedly configure and reboot the unit while you learn how to use the product.
  - It provides access to the firmware user interface, which you may want to learn about or use.
- **If you are a new user and are remote from the Network Utility**, dial-up terminal emulation is preferable to Telnet for some of the same reasons as for a new user who is physically adjacent to the unit.
- **If you are placing a configured Network Utility into a production network**, choose the terminal console access method that best fits your network configuration and also your service and operations strategy. You can use Telnet as the ″everyday″ terminal console access method, and dial-up terminal emulation as the backup service method when either the network is not available or firmware access is required. IBM service personnel will use either method when they debug configuration and network problems.

ASCII Terminal Setup and Usage

Use this section if you are setting up an ASCII terminal or a workstation with terminal emulation. You can use ASCII terminal emulation to access Network Utility whether or not it has ever been configured.

An ASCII terminal console provides access both to the main operational code (the command-line interface), and to the firmware user interface (see “Firmware” on page 68). If you are remotely dialed-in to the PCMCIA or an external modem and you reboot the unit, you will lose your console connection and need to re-dial[2]. If you are locally connected, your console connection is maintained during a reboot.

[2] If you are using an external modem and it can be set to ignore a drop in DTR from the Network Utility, then you will not lose the console connection when you reboot the Network Utility. Refer to the user documentation for the modem.

Attaching an ASCII Terminal

Attach an ASCII terminal or emulator (with the appropriate emulation software) to provide local or remote access as shown in Figure 4 on page 16 and Figure 3 on page 16. DEC VT100 and DEC VT220 ASCII terminals are supported, as well as devices such as personal computer systems that are configured to emulate them.

Serial Port and PCMCIA Modem Default Settings

These are the default settings for the serial port:

Speed	19.2 Kbps
Parity	None
Data Bits	8
Stop Bits	1
Terminal type	VT220, Monochrome

To change the settings for the serial port, follow these steps:

1. Reboot the Network Utility to the firmware main menu, using one of the tasks in “Boot Options: Fast Boot and Reaching Firmware” on page 45.
2. Select Option 1, **Manage Configuration**
3. Move the cursor to the row for the COM1 serial port and press **Enter**
4. Move the cursor to the characteristic you want to change (for example, baud rate), and press **Enter**
5. Select the new value and press **Enter**
6. Press **Esc** to return to the firmware main menu
7. If you want to continue the current boot sequence and have the operational code start using the new settings, press **F9** (Start OS).

   If you want to reboot into the firmware and have the firmware start using the new settings, press **F3** (Reboot).
8. Change the settings of your terminal or terminal emulation software to match the new settings of the Network Utility serial port.

The PCMCIA modem is a standard item that is shipped with the Network Utility in most countries. It is a 33.6 Kbps V.34 data modem, and it negotiates the data rate to be used between it and the partner modem on the other side of the telephone network. Using data compression, this modem is capable of data throughput greater than 33.6 Kbps.

The data rate between the Network Utility system and its PCMCIA modem defaults to 19.2 Kbps, but you can raise it to accommodate the higher throughput that the two modems may be able to achieve between themselves. For example, you may want to set this rate to 57.6 Kbps so that it is higher than the effective data rate of two 33.6 Kbps modems running data compression. If your modems are both faster than 19.2 Kbps, raising this rate will lower Xmodem file transfer time.

To change the data rate and any of the other settings for the PCMCIA modem, follow the same procedure given above for serial port settings, but select COM2, the PCMCIA modem, instead of the serial port.

ASCII Terminal Setup Attributes

This is a list of all the options required to set up a terminal or terminal emulator that is connected to the Network Utility service port. Not every terminal (particularly 3151 and 3161) will have all these options. You should use the information to set the options that you can set on your terminal.

Terminal Settings and Function Keys

Baud Rate:	19200 bits per second **Note:** Baud rate must match the speed of the Network Utility serial service port.
Parity:	None
Data bits:	8
Stop bits:	1
Duplex:	Full Duplex
Flow Control:	XON/XOFF and RTS/CTS (see Note 1)
Screen Control:	ANSI Full screen
Screen Width:	80 Characters
Screen Height:	24 Lines
Line Wrap:	ON
Screen Scroll:	ON
Carriage Return Translation:	CR (0Dx)
Backspace Translation:	Destructive

Notes:

1. Set terminals and terminal emulator programs that do not have flow control options to “Permanent Request to Send”.
2. Set terminal emulators that require a terminal type selection to VT-220.

Function Keys

When accessing the firmware, you will need to use function keys F1, F2, F3, F4, F6, and F9. Not all terminals or terminal emulators provide standard support for these function keys (for example, the VT100 types).

The simplest way to simulate these functions keys is to type the following sequence, with no more than two seconds between each step:

1. **Ctrl-a**
2. The number (not the function key itself) of the function key you want
3. **Enter**

Alternatively, you can set up your terminal emulator to generate the following escape key sequences when you press a function key:

```
Function 1 (F1):  <Esc> O P        Hex: 1B 4F 50
Function 2 (F2):  <Esc> O Q        Hex: 1B 4F 51
Function 3 (F3):  <Esc> O R        Hex: 1B 4F 52
Function 4 (F4):  <Esc> O S        Hex: 1B 4F 53
Function 6 (F6):  <Esc> [ 0 0 6 q   Hex: 1B 5B 30 30 36 71
Function 9 (F9):  <Esc> [ 0 0 9 q   Hex: 1B 5B 30 30 39 71
```

Note: In the definitions of function keys:

O = uppercase O

0 = the number zero

All characters are case-sensitive

Multiple Terminal Users

One user at a time can have an active terminal console through the system card serial port or the PCMCIA modem interface. If a workstation is connected locally to the serial port and a call comes in over the PCMCIA modem, priority is given to the call. After the call, the user at the local workstation will have to reconnect.

Telnet Setup and Usage

Use this section if you are setting up Telnet terminal console access.

Telnet provides access only to the main operational code (the command-line interface), and not to the firmware user interface. If you reboot the unit from the command-line interface, you lose your Telnet connection and you need to re-establish it after the unit has rebooted.

If your unit has never been configured, the only way you can Telnet to it is by using the default SLIP or PCMCIA EtherJet IP addresses.

SLIP Addresses

The default SLIP IP addresses for use with the PCMCIA or external modems are:

For the workstation:
10.1.1.3

For the Network Utility:
10.1.1.2

For instructions about installing SLIP, refer to the documentation for your version of TCP/IP PC software.

PCMCIA LAN IP Addresses

The default IP addresses for use with the PCMCIA EtherJet PC card are as follows:

For the workstation:
10.1.0.3

For the Network Utility:
10.1.0.2

You can change these addresses either from the operational code command-line interface or from the firmware. (Use the procedures that are documented in “Basic IP Configuration and Operation” on page 42.) You must first bring up your initial user console using ASCII terminal emulation or by telnetting to the default IP addresses.

Network Interface IP Addresses

There are no default IP addresses for network interfaces (those on the adapters in the adapter slots). Use either the command-line interface or the Configuration Program to set up IP addresses for network interfaces. All the example configuration tables in “Part 3. Configuration and Management Specifics” on page 117 show how to set up IP addresses on interfaces.You cannot Telnet in through a network interface until you activate the IP address configuration change.

In addition to assigning IP addresses to an interface, you can assign one to the entire unit. This IP address is known as the *internal* IP address, and it remains active independent of the state of individual network interfaces.

If you have a Model TN1 and are using the TN3270 server function, you must configure the IP address and TCP port number to be used by TN3270. If you accept the default Telnet port number 23 for TN3270, you must attach your console Telnet sessions to a different IP address than the one you have configured for the TN3270 server. This allows the unit to distinguish console Telnet sessions from TN3270 client sessions.

Multiple Telnet Users

Two users at a time can bring up Telnet consoles through network interfaces. A third user’s Telnet attempt will be rejected until one of the first two users has disconnected. One user at a time can have an active console through the system card service port or PCMCIA interfaces, including Telnet through SLIP or the PCMCIA LAN card.

Getting to the Command Prompt

After you have set up your user console, look for the messages and go to one of the command prompts described here.

What You Should See

If you have an active user console from the time you power on a Network Utility until it presents the first command prompt, you see a sequence of informational status messages about:

- Escaping to change the terminal type
- Memory initialization
- System board diagnostics
- Other diagnostics
- Boot progress (including how to interrupt the boot in order to reach the firmware menus)
- Loading the operational code from disk, ending with the following messages:

```
Please press the space bar to obtain the console.

Loading /hd0/sys0/LMX.ld from disk ...
Loading /hd0/sys0/LML.ld from disk ...
Loading /hd0/sys0/sysext.ld from disk ...
Loading /hd0/sys0/diags.ld from disk ...
Loading /hd0/sys0/snmp.ld from disk ...
Loading /hd0/sys0/router.ld from disk ...
Loading /hd0/sys0/appn.ld from disk ...
Loading /hd0/sys0/tn3270e.ld from disk ...

<you press the space bar>
Console granted to this interface
Config (only)>
```

At any time after you see the prompt `Please press the space bar to obtain the console`, press the space bar to attach the Network Utility console process to your session. The system acknowledges this action with the message `Console granted to this interface`, and by displaying a command prompt after the code loading is complete.

If you are at a Network Utility that has never been configured, the system presents the command prompt `Config (only)>`. You can then proceed as described in Chapter 3. Performing the Initial Configuration, to configure the Network Utility. If the Network Utility has been configured sufficiently to become fully operational, the system presents the asterisk (*) command prompt.

Only a directly attached ASCII device can show you all the messages from the entire boot sequence. If you are dialing in through the PCMCIA modem or telnetting in to bring up your user console, the Network Utility needs to be at least partially booted before it can respond to your connection attempt. When you do connect, the boot process may be in one of its later phases or may complete. The system grants you the console immediately and then gives you a command prompt after the boot process completes.

Solving ASCII Terminal Problems

Garbage, random characters, or the inability to connect your terminal to the Network Utility service port can have a number of causes. The most common cause of garbage or random characters is that the terminal baud rate is not synchronized with the Network Utility.

The Network Utility is always set to a specific baud rate, which by default is 19.2 Kbps. The only way to change this rate is through the firmware, so you must have a working console to change it. If your console in unreadable, try different baud rate values on the terminal side until you find the one that gives you readable status messages or command prompts.

Other causes of connection problems include:

- No null modem on the serial cable
- Defective terminal or Network Utility ac grounds
- Defective, incorrectly shielded, or incorrectly grounded cable between the terminal and the Network Utility.
- Defective terminal or terminal emulator
- Defective Network Utility system board

Refer to ″Service Terminal Display Unreadable″ in the *2216 Nways Multiaccess Connector and Network Utility Service and Maintenance Manual* for more information on handling these problems.

Solving Telnet Problems

The most common Telnet problem is the inability to reach the Network Utility through your IP network. You can use the standard debug tools (ping and traceroute) to determine what is happening. If you are trying to ping to the Network Utility internal IP address, you need to set up a host route in your workstation to that address, with the next hop being the interface IP address through which you will be entering the Network Utility.

You can also try to ping from the Network Utility back to your workstation. The firmware provides a way to do this from an EtherJet or SLIP port, and the operational code Console process provides a way to do this from network interfaces. See “Basic IP Configuration and Operation” on page 42 for a summary of these procedures.

Chapter 3. Performing the Initial Configuration

This chapter introduces the basics of configuring Network Utility, and gives specific procedures for configuring a new Network Utility. These procedures move the Network Utility from a passive state where it is waiting to be configured, to a state where it has active network interfaces and protocols.

Before using these procedures, you must connect a user console as described in “Chapter 2. Bringing Up a User Console” on page 15.

Configuration Basics

A Network Utility configuration is a collection of data items that control how the software operates, including such elements as:

- What interfaces to activate
- What links to bring up
- What protocols and features to make active
- What functions in a given protocol or feature to make active
- What network addresses and names to use

When you boot up a Network Utility, the system reads its configuration information from a file on the hard disk, and activates interfaces and protocols according to the information in that file. You create the file in one of two ways:

- Using the command-line interface from a user terminal console

  You type commands to create configuration data items in memory and then write the configuration to the Network Utility hard disk.
- Using a graphical configuration program that runs on a stand-alone workstation

  You create the configuration on the workstation and then transfer it to the Network Utility hard disk.

  The Network Utility Configuration Program is shipped on a CD-ROM in the carton with every new Network Utility, and is also downloadable from the Web. Versions are available for Windows 95 and Windows NT, AIX, and OS/2. Workstation requirements are documented in the *Configuration Program User’s Guide*, which is also shipped hardcopy in the carton with Network Utility.

Choosing Your Configuration Method

Some IBM routing product users prefer the Configuration Program, others prefer the command-line interface, and still others use a combination of the two. The approach you take is up to you.

Here are some of the factors users cite in favor of the Configuration Program:

- It enables centralized maintenance of configuration files for multiple Network Utilities and 2216s.
- It provides table-oriented, intuitive organization of data items.
- It performs more input validation and cross-checking of parameters than the command-line method.
- It includes online helps for individual data items.

Here are some of the factors users cite in favor of the command-line interface:

- It provides a single integrated method for configuration, dynamic reconfiguration, and monitoring.
- It is well documented in product publications and IBM ″redbooks.″
- It is simple to make and try quick configuration changes.
- Setting up a user console does not require as many workstation resources or as much time as installing the Configuration Program.

Getting Started from Config-only Mode

If you boot a Network Utility and see the `Config (only)>` prompt from the user console, you are in config-only mode. A Network Utility boots up into config-only mode when the current configuration file on the hard disk has no data items that would allow it to do any useful functions like forward data packets[3]. You need to configure at least one adapter port and one protocol (for example, IP, DLSw, or APPN) and reboot in order for the Network Utility to start up in normal working mode.

If you have a Network Utility at the `Config (only)>` prompt, perform these actions:

1. Choose whether you want to use the command line or the Configuration Program for your initial configuration. It is easy to switch methods later if you want to try both.
2. Based on your choice, follow one of these procedures:
   - “Procedure A: Command-Line Procedure for Initial Configuration”
   - “Procedure B: Configuration Program Initial Configuration” on page 29

Procedure A: Command-Line Procedure for Initial Configuration

Use this procedure to configure a Network Utility for the first time starting from the `Config (only)>` command-line prompt:

Part 1: Create a Minimal, Basic Configuration

1. Use the **add device** command to configure at least one network interface as follows:
   a. Type **add dev ?** to see a list of supported adapter types.
   b. Type **add dev** *type*, where *type* consists of the first few letters from a row of the adapter list. For example, **add dev tok** selects the Token-Ring adapter. Type enough letters to uniquely identify the adapter you want.
   c. When prompted for slot number, enter **1** for the left-hand adapter slot of the Network Utility, or **2** for the right-hand slot.
   d. If you are adding a multiport adapter, the system prompts you for the port number of the interface you want to configure. Port numbers on adapters are fixed as follows:
      - Ports on multi-port LAN adapters are numbered 1 and 2 and are labelled on the adapter face.
      - Ports on multi-port WAN adapters are numbered starting with 0 and are labelled on the connectors at the end of the adapter cable.

[3] This also happens if your configuration becomes corrupted.

e. The system then assigns a logical *interface number*, also known as a *net number*. This is the key number by which you refer to this interface on every other command in the system. For example, if you want to delete the configuration for this interface, type **delete interface** and then give the logical interface number.

f. If necessary, make the following adjustments to the default device configuration:

   If you added a Token-Ring port and you want it to run at 16 Mbps instead of the default 4 Mbps, type these commands:

   net *interface number*

   speed 16

   exit

   If you added a 10 Mbps (not 10/100) Ethernet port and you want to use the BNC (10BASE2) connector instead of the default RJ45 (10BASET) connector, type these commands:

   net *interface number*

   conn bnc

   exit

Repeat step 1 for each interface you want to configure.

2. If you want to be able to dynamically add interfaces in the future without needing to reboot Network Utility, type **set spare** *number* from the `Config (only)>` prompt, where *number* is the maximum number of interfaces you need to add without rebooting.
3. Use the **qconfig** command to start the ″Quick Config″ program. Use this program to configure IP and SNMP access to Network Utility as shown below.

   Quick Config is a feature of the command-line configuration process. Instead of waiting for you to type commands, it asks you questions and creates configuration data based on your replies. An example of a Quick Config question is:

   ```
   Configure Bridging? (Yes, No, Quit): [Yes]
   ```

   The values in parentheses are the possible responses. The value in square brackets is the default response. To accept the default, press **Enter**.

   Respond as follows to the Quick Config questions (some of these are default responses):

   a. Configure Bridging by responding **no** to `Configure Bridging?`

   b. Configure Protocols by responding **yes** to `Configure Protocols?`

   c. Configure IP as follows:

      1) Enter **yes** to `Configure IP?`

      2) For any interfaces to which you want to assign an IP address, respond **yes** to `Configure IP on this interface?` If you intend to use the PCMCIA EtherJet card as your only IP interface, respond **no** for every configured network interface.

      3) Enter the IP address at the `IP Address` prompt.

      4) Enter the IP mask at the `Address Mask` prompt.

      5) If you want to enable RIP or OSPF, respond **yes** to `Enable Dynamic Routing?` and respond to subsequent related questions.

6) If at some point you may want to send a configuration directly from the Configuration Program to this Network Utility, respond **yes** to `Define Community with Read_Write_Trap Access?` and enter any single-word name you want as the community name.

   If you never expect to use the Configuration Program, respond **no**.

7) Respond **yes** to `Save this configuration?` This saves the IP part of the configuration in memory.

d. Save the Configuration file by responding **yes** to `Do you want to write this configuration?`

Part 2: Activate the New Configuration

You have now configured at least one interface and one protocol (IP, with SNMP). This small configuration is sufficient to leave config-only mode.

1. From the `Config (only)>` prompt, type **reload** and respond **yes** to the confirmation prompt. The Network Utility reboots and activates your new configuration.

   If you see a prompt about saving configuration changes, that means you have made some configuration changes after saving the configuration file when you completed Part 1 of this procedure. Type **yes** to save these changes as part of your new configuration before the reboot proceeds.
2. Verify the Network Utility reboot

   If your user console is using a dial or Telnet connection, reboot causes you to lose your connection. Reconnect after a few minutes. Otherwise just watch the boot messages from your console.

   When the reboot completes, your console should display the * command prompt, indicating that you are in normal operating mode and no longer in config-only mode. The configuration you created in Part 1 of this procedure is now active.

Part 3 - Add Additional Protocol Information

You are now in normal operating mode with the interfaces you configured, running only IP.

If you are a new user and want to become familiar with the product before configuring the rest of your functions (such as TN3270 or DLSw), skip the rest of this procedure and see the guidelines in “What to Do Next” on page 33.

If you want to configure all your functions right now, continue here.

1. Select the configuration scenario from “Part 3. Configuration and Management Specifics” on page 117 that most nearly resembles the use to which you are placing this Network Utility.

   - Model TN1 Users - See “Chapter 12. TN3270E Server” on page 127.
   - Model TX1 Users - See “Chapter 14. Channel Gateway” on page 203, “Chapter 16. Data Link Switching” on page 239 or “Chapter 19. Virtual Private Networks” on page 275.

   If none of these scenarios is suitable, use the *MAS Protocol Configuration and Monitoring Reference*, *MAS Using and Configuring Features*, and *MAS Software User’s Guide* to determine what you need to configure.

2. In the ″Example Configuration Details″ chapter that follows your selected scenario, find the configuration parameter table that corresponds to that scenario[4]. Use the ″Command-Line Commands″ column to guide you in configuring that scenario, changing the values for your particular adapters and network.

   If you find that you are having trouble navigating the command line and entering commands, you may want to get more familiar with general command-line configuration before proceeding. See “What to Do Next” on page 33 for suggestions on how to proceed.

3. When you have finished entering configuration commands, repeat the steps in “Part 2: Activate the New Configuration” on page 28, but issue the **reload** command from the * prompt instead of the `Config (only)>` prompt.

Procedure B: Configuration Program Initial Configuration

Use this procedure to configure a Network Utility for the first time using the Network Utility Configuration Program.

Part 1: Create the Configuration at the Configuration Program

1. From the Configuration Program CD-ROM, install the appropriate version of the Configuration Program onto your workstation.

   For installation instructions, see:

   - The Network Utility README file on the CD-ROM.
   - The *Configuration Program User’s Guide*, which is shipped along with the CD-ROM.

   Start the Configuration Program. If you want to try the program by doing a new configuration from scratch, select **New configuration** and **Network Utility** from the **Configure** option on the menu bar in the Navigation Window.

2. Select the configuration scenario from “Part 3. Configuration and Management Specifics” on page 117 that most nearly resembles the use to which you are placing this Network Utility.

   - Model TN1 Users - See “Chapter 12. TN3270E Server” on page 127.
   - Model TX1 Users - See “Chapter 14. Channel Gateway” on page 203, “Chapter 16. Data Link Switching” on page 239 or “Chapter 19. Virtual Private Networks” on page 275.

   If none of these scenarios is suitable, use the *MAS Protocol Configuration and Monitoring Reference*, *MAS Using and Configuring Features*, and *MAS Software User’s Guide* manuals to determine what you need to configure. Use any of configuration parameter tables in “Part 3. Configuration and Management Specifics” on page 117 as an example of mapping command-line commands to Configuration Program panels. When you have completed your configuration, skip to step 7 on page 30.

3. In the ″Example Configuration Details″ chapter that follows your selected scenario, find the configuration parameter table that corresponds to that scenario.[4]

4. From your Web browser, follow the `Support and Downloads` links from the main Network Utility web page

[4] If no corresponding table exists, use the ″Keys to Configuration″ section for that scenario, to get started.

`http://www.networking.ibm.com/networkutility`,

and find the example configuration file that matches your selected scenario. Download this file in binary and transfer it to the workstation running the Configuration Program.

5. Select **Open Configuration ...** from the Navigation Window and select the path and file name of the example configuration file you downloaded.
6. Use the ″Configuration Program Navigation″ and ″Configuration Program Values″ columns in the table from step 3 on page 29 to guide you in moving through the configuration and changing the values for your particular adapters and network.
7. When you have a configuration ready to send to your Network Utility, select **Save configuration as ...** to save the configuration on your workstation. You may want to choose a new name so you can leave the original example configuration file unchanged.

Part 2: Transfer the Configuration to the Network Utility and Activate It

You have now created the initial configuration. All that remains is to transfer the configuration to the Network Utility hard disk and reboot the Network Utility to activate it. How you should do this transfer depends on your connection setup, as follows:

- If your Configuration Program workstation supports TCP/IP and has physical connectivity to either the Network Utility PCMCIA EtherJet card or a network adapter in slot 1 or 2, use Procedure A.
- If your user console is via ASCII terminal emulation and you prefer using Xmodem to setting up the above IP connectivity, use Procedure B.

You can also refer to “Loading New Configuration Files” on page 82 for a complete list of the ways to transfer a configuration to Network Utility. You will need TFTP server software on a TCP/IP workstation if you choose not to follow either Procedure A or B. **Procedure A: Direct transfer through Network Utility PCMCIA EtherJet or a network adapter**

Use this procedure if your Configuration Program workstation supports TCP/IP and has physical connectivity to the Network Utility PCMCIA EtherJet card or a network adapter in slot 1 or 2.

1. Configure Network Utility quickly from the command line, so that it has an IP address on at least one interface, and IP and SNMP enabled.
   a. From your user console, perform the steps in “Part 1: Create a Minimal, Basic Configuration” on page 26. Be sure to:
      1) Use **add device** to define at least one interface in slot 1 or 2
      2) In Quick Config, respond **yes** to `Define Community with Read_Write_Trap Access?`
   b. In the Configuration Program, verify that the configuration you are about to send has SNMP enabled and the same community name defined with ″read-write trap″ access. This is required so that after you activate this configuration, you will be able to repeat step 3 on page 31 of this procedure to send another configuration.
   c. Perform the steps in “Part 2: Activate the New Configuration” on page 28 to reboot Network Utility and activate this temporary command-line configuration.

2. If you plan to use the PCMCIA EtherJet card, set up its IP addresses as follows after the Network Utility reboot is complete:

   From the * prompt, type **talk 6**. From the `Config>` prompt, type **system set ip** and enter the following values as prompted:

   - IP address: the IP address you want to use for the EtherJet card
   - Netmask: the mask for the subnet attached to the EtherJet card
   - Gateway address: the IP address for the Configuration Program workstation, or the IP address of a router through which the Network Utility can reach it

   Next to each prompt, the system shows the current value as the default. To accept the default, press **Enter**. After you enter all the values, any address change you specified takes effect immediately. The values are stored in Network Utility NVRAM and not as part of any configuration file.

3. Send the configuration from the Configuration Program (using SNMP):

   a. From the **Configure** drop-down menu, select **Communications** and **Single router**.

   b. On the Communicate panel, enter:

      - IP address or name: The IP address of the Network Utility interface you want to send the configuration through. This is either the PCMCIA EtherJet IP address, or the network interface IP address you assigned in Quick Config.
      - Community: The community name you assigned in Quick Config.

   c. Select **Send configuration** and **Restart router**. Accept or enter the current date and time, so that Network Utility will reboot with the new configuration immediately after receiving it.

   d. Click on **OK**. The Configuration Program immediately starts sending configuration data items to the specified routers using SNMP.

      The Configuration Program provides status and result messages about the transfer. If the send operation fails, the Configuration Program lists possible reasons which you should then verify and correct.

   After the Configuration Program completes its configuration transfer, the Network Utility stores the configuration on disk and reboots itself as you directed.

4. Verify the Network Utility reboot

   If your console is through a dial or Telnet connection, reboot causes you to lose your connection. Reconnect after a few minutes. Otherwise just watch the boot messages from your user console.

   When the reboot completes, your console should display the * command prompt, indicating that you are in normal operating mode and no longer in config-only mode. The configuration you created in Part 1 of this procedure is now active.

Procedure B: Indirect Xmodem transfer through user console session

Use this procedure if your console is via ASCII terminal emulation and you prefer using Xmodem to setting up IP connectivity from the Configuration Program workstation.

1. From the Configuration Program, export your configuration into the file format understood by Network Utility

   From the **Configure** drop-down menu, select **Create router configuration** and specify the path and file name for a .CFG file. Click on **OK** to write the file.

2. If necessary, transfer the .CFG file from the Configuration Program workstation to your terminal emulation workstation.
3. From your console at the `Config (only)>` prompt, follow this sequence:

   ```
   Config (only)>boot
   Boot configuration
   Boot config>dis auto
   Select the duration to disable autoboot: (once, always): [always] once
   AutoBoot mode is now disabled once.

   Operation completed successfully.
   Boot config>exit
   Config (only)>rel y
   ```

   If you are prompted about saving configuration changes, respond **no**. Network Utility reboots and stops at the firmware menu.

   If your console is through a dial connection, reboot causes you to lose your connection. Reconnect after a few minutes and you see the firmware menu.
4. Make the following sequence of firmware menu selections:
   a. System Management Services (main menu): Option 4, **Utilities**
   b. System Management Utilities: Option 12, **Change Management**
   c. Change Management Software Control: Option 12, **Xmodem software**
   d. Select Type: **Config**
   e. Select Bank: choose **Bank A** (active bank)
   f. Select Config: choose position **1**[5]

   The firmware tells you when to start the file transfer.
5. Go to your terminal emulation package and start the transfer of the file from your workstation server, using whatever name you like. When the Network Utility has received the configuration file, the status of the file position will change from CORRUPT to AVAIL. You can verify that this has happened using option 7, **List Software**, from the firmware Change Management menu.
6. Boot the Network Utility using the configuration you just loaded.
   a. Use Option 9 **Set Boot Information** to select the current op-code bank and the new configuration.
   b. Press **Esc** to reach the main menu and then **F9** (Start OS) to boot the Network Utility with the new configuration.
7. Verify the Network Utility boot

   If your console is through a dial connection, you do not lose the connection when you use the Start OS option. Watch the boot messages from your console.

   When the boot completes, your console should display the * command prompt, indicating that you are in normal operating mode and no longer in config-only mode. The configuration you created in Part 1 of this procedure is now active.

[5] This selection of bank and configuration file position assumes that this is the first time you have booted this Network Utility. For more background on this topic, see “Configuration Files on Disk” on page 72.

What to Do Next

If you have followed the procedures in this chapter, your Network Utility is now in full operational mode with a configuration you created. With your user console at the * prompt, you are now in a position to use the command-line interface to:

- Query the status of interfaces and protocols
- Activate events and monitor the event log
- Issue operator commands to effect status changes
- Make dynamic configuration changes without rebooting

These are the basic tools to see whether your new configuration is working properly, and to make small adjustments to that configuration.

If the command-line interface is new to you, you can use “Chapter 5. A Guided Tour through the Command-Line Interface” on page 51 to familiarize yourself with its concepts and how to use it.

If you have some previous experience with IBM routing products or prefer to try tasks without following a tutorial, you can use “Chapter 4. Quick Reference to the User Interface” on page 35 as summary information about command-line navigation and some common tasks.

You can use Chapters 6 through 10 to get more background on:

- Managing configuration files
- Dynamic reconfiguration
- Managing what Network Utility is doing, both locally and using remote network management products
- Updating software and firmware
- Requesting service and support

You may have already used the example configuration information in “Part 3. Configuration and Management Specifics” on page 117. The chapters there also contain introductory information about configuring and monitoring the functions:

- TN3270E server
- Channel gateway
- Data Link Switching
- Virtual Private Networking

If you have already configured one of these functions in your initial configuration, use the ″Managing″ section from the corresponding chapter to begin monitoring and debugging that configuration.

Chapter 4. Quick Reference to the User Interface

This chapter contains summary information about navigating the command-line interface, entering commands, and performing common tasks. For a complete explanation of this material with examples, see "Chapter 5. A Guided Tour through the Command-Line Interface" on page 51.

Navigating

The command-line interface consists of a tree of menus whose root is the asterisk (*) prompt. You type commands and use control keys to move to various places in the tree and then you type commands to actually perform tasks.

Processes and Prompts

From the * prompt, use the **talk** command (abbreviated **t**) to attach to one of several processes, or ways of viewing the system. Each process from which you enter commands is identified by a different command prompt.

Table 5. Key Processes

Name	Command to Access	Purpose	Top-level Prompt
Config	**t 6** or **config**	View and modify the configuration	`Config>`
Console	**t 5** or **console**	View and control running status, make dynamic configuration changes	+ (plus sign)
Monitor	**t 2** or **event**	View real-time event message log	(none)

Type **t *n*** and then press **Enter** twice to obtain the command prompt. Type **Ctrl-p** to return to the * prompt from inside any process.

The monitor process has no command prompt because instead of issuing commands in that process, you watch a running log of event messages. You can type **Ctrl-s** to pause scrolling, and **Ctrl-q** to resume it.

Subprocesses

When you are working inside the talk 6 or talk 5 processes, some commands change the input prompt and provide you with a new command menu that is specific to a functional area. For example,

- Typing **protocol dlsw** under talk 6 moves you to the Config subprocess for configuring Data Link Switching. The command prompt becomes `DLSw config>`.
- Typing **perf** under talk 5 moves you to the Console subprocess for viewing CPU utilization statistics The command prompt becomes `PERF Console>`.

You can also move from one subprocess into another subprocess. For example, typing **ban** from the DLSw Config subprocess moves you to the Boundary Access Node Config subprocess. You have gone one nesting level deeper in the menu system; you must return through the DLSw subprocess.

The following navigation rules apply:

- To enter a subprocess, type the specific command that takes you there. Type **?** at any menu to see the available commands. You know you have entered a subprocess when the command prompt changes.
- To leave any subprocess and return to the next higher level menu, type **exit**.
- To leave any subprocess and move immediately to the * prompt, type **Ctrl-p**. This also takes you out of the current process.
- To resume a subprocess after having typed **Ctrl-p**, type **t** *n* (where *n* is the process number you left), then **Enter** twice. You resume that process in the subprocess where you typed **Ctrl-p**.

Entering Commands

You type commands to enter processes, enter and leave subprocesses, and perform tasks. Some task commands prompt you for parameter values, while others do not require any input other than the command name.

Forming Commands

A command is a sequence of one or more key words, optionally followed by parameter values that were typed on the original command line. The following guidelines apply to forming a command:

- You must type a complete command before the system takes the action or prompts you for input parameters. If you type only part of a valid command (not enough key words), the system responds with `Command not fully specified`.
- You can type **?** at any process or subprocess prompt, or after any incomplete command, to see a menu of command keywords available from that point. You can use this to find or complete a command, as shown in this abbreviated example:

  ```
  Config>?
  ADD (device, user)
  BOOT and load file functions
  CHANGE (device, password, user)
    ...             < other commands not shown>
  Config>add
  Command not fully specified
  Config>add ?
  DEVICE
  NAMED-PROFILE
  PPP-USER
  TUNNEL-PROFILE
  USER
  Config>add user
  Enter user name:  []?  <enter>
  No user was added
  Config>
  ```

  In the example, **add** was not a complete command, but **add user** was. After the user typed the complete command, the system prompted for an input parameter value.
- You can abbreviate most command keywords to the minimum number of characters that uniquely select them from the menu on which they appear. For example, you can type **t 6** instead of **talk 6**, and **p appn** instead of **protocol appn**. In the example above, the user could have typed **a u** instead of **add user**.
- You can work with previously entered commands in both talk 6 and talk 5 using the following keys:

Ctrl-B to scroll backward through previously entered commands

Ctrl-F to scroll forward through the list of previously entered commands

Ctrl-U to clear a retrieved command off the command line

Backspace
to edit a retrieved command starting from the end

The command history buffer is shared by talk 6 and talk 5.

Automatic Command Completion

Beginning with MAS V3.3, Network Utility can assist you in forming commands by automatically completing keywords you type and by showing you available menu options. You configure this command completion function to be disabled or enabled, either at the command line or from the Configuration Program. Command Completion is enabled by default when you start a new MAS V3.3 configuration, but if you upgrade an existing configuration, this function is disabled by default. New users are recommended to run with command completion enabled (type **enable command** from either the **Config (only)>** or **Config>** prompts).

To illustrate the behavior of Command Completion, assume that the following commands are allowed in a given menu context. (This is an example menu only.)

enable	auto-refresh
	caching
set	cache-size
	cache-timeout
	priority

- If you type **ena** and press the Space Bar, the full command is shown as **ENABLE**. If you now type **?**, a list of possible items to enable are shown (**auto-refresh** and **caching**) and the command **ENABLE** remains on the command line.
- If you type **ena** and press **Enter**, a message is printed that the command is not fully specified, and a list of possible items to enable are shown (**auto-refresh** and **caching**) and the command **ENABLE** remains on the command line.
- Because the **ENABLE** command requires an item to enable, it appears in a list of possible command completions with "..." in the left margin to indicate that more input is required for the command.
- If your input matches multiple commands, a list of possible completions is displayed. Your input on the new command line is expanded to the longest common prefix. For example, if you enter **set ca**, and then press the space bar, **CACHE-SIZE** and **CACHE-TIMEOUT** will be listed, and the new command line will be expanded to **SET cache-**, since "cache-" is common to both possible completions. Now you must type the letter "s" or the letter "t" to distinguish between the possible completions ″size″ or ″timeout″.
- Common commands sometimes appear in an alternate form (**SHOW**, **DISPLAY**, **LIST**). If the Command Completion does not yield a match on a common command, such as **SHOW**, the alternatives **DISPLAY** or **LIST** will be displayed, if found.
- If the search for a command (and alternatives) does not yield an exact match, you are presented with a list of possible completions, using some portion of your

input. For example, **enable** followed by the Space Bar would be replaced with **ena** and **ENABLE** would be listed as the possible completion.

- When a list of possible commands is shown, you can use the Tab key to cycle through one command at a time on the current command line. You can use the Space Bar or Enter key to select the command shown.

To get comprehensive online help for the Command Completion function, type **<esc> ?** from any command prompt.

Entering Command Parameter Values

Some of the commands that perform a task require you to supply values for input parameters. You can either let the system prompt you for these input values, or (in most cases) type them ahead on the command line following the command name.

If you do not type parameter values ahead:

- Type only the command name and press **Enter**.
- The system prompts you for each parameter in turn, supplying the default value for that parameter inside square brackets. Some defaults are fixed, but most are the last value you assigned to that particular parameter.
  - To accept the default value, press **Enter**
  - To supply a new value, type the value and press **Enter**
  - If the brackets are adjacent, as in [], there is no default and you need to supply a value

  The system performs a validity check on your response before prompting you with the next value.
- When you have responded to the final parameter prompt, the system takes the action specified by the command.

If you want to type parameter values ahead:

- Type the command name following by one or more parameter values separated by blanks, and then press **Enter**.
- The system parses the command line and supplies the first value to the first parameter, the second value to the second parameter, and so on. You must supply the values in the order expected.

  The system performs a validity check as it assigns each value to the corresponding parameter.
- If the command requires more parameters than you have supplied values for, the system prompts you for the additional values as outlined above.
- When the system has supplied a valid value to each parameter, it takes the action specified by the command.

Typing values ahead can be a convenient shortcut for experienced users. You need to be careful that you provide valid parameters in the right order.

You should be alert for cases where you type **?** following a full command, and the command treats the ″?″ as a typed-ahead value for its first input parameter. If this happens, abort or undo the command and try again.

Common Error Messages

Table 6 explains several standard error messages from the command-line interface:

Table 6. Error Messages and Corrective Actions

Error Message	Explanation and Corrective Action
`Command error`	The command you typed does not exist on the current menu. You may have a typo, or be in the wrong place to issue this command, or not have typed enough characters to identify the command from the menu. Look at your prompt to verify where you are, and type **?** to see the available commands. Correct the command or move to the right place.
`Command not fully specified`	The command keywords you typed do not form a complete command. Type **Ctrl-b** to retrieve the command, then add ″ **?**″ to the end of it to see your choices for the next keyword. Pick the next keyword to add and re-issue the command replacing **?** with that keyword. You may also want to consult the appropriate MAS command-line reference manual for the command you are trying to enter.
`Command syntax error`	You typed an incorrect form of a valid command. You may have supplied an invalid or unexpected parameter. Try the command again with no parameter values, or consult the appropriate MAS command-line reference manual entry.
`Feature <name> available but not enabled`	Under talk 5, you tried to enter the Console subprocess for a feature that is supported in your software load but is not actively running. Your current configuration either did not enable the feature, or is missing key values required in order to activate the feature. If you are using the Configuration Program, look on the Navigation Panel for **?**s, indicating required parameters not set. Follow the **?** trail to the panel or panels with field names in red that are not set. If you are doing configuration from the command line, consult the example configurations in this book and in the MAS reference manual chapter for this feature. Look for the key parameters that are shown as base parameters for enabling the function.
`Protocol <name> available but not configured`	The same as described above for `Feature available but not enabled`, but applied to a protocol.

Key User Tasks

This section organizes common user tasks into groups and provides tables with a quick reference to the commands to perform each task.

Configuring Physical Adapters and Interfaces

Table 7 on page 40 describes how to perform tasks relating to configuring physical adapters and interfaces.

Table 7. How to Configure Physical Adapters and Interfaces

Task	How to do it
Add an interface at initial configuration	1. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 2. Type **add dev ?** to see a list of supported adapter types. 3. Type **add dev** *type*, where *type* is the keyword from the list for the adapter type you want. 4. Enter the physical slot and port number (if asked) of the interface you are configuring. Slots are 1 and 2 from left to right. LAN ports are numbered on the adapter face, and WAN ports are numbered on the cable connectors. 5. Note the new logical interface (net) number the Network Utility assigns to this interface. 6. Type **net** *logical interface number* to enter the Config subprocess for the particular interface type. Use the commands in that subprocess to verify or change from the default settings for the interface. 7. Type **exit** to return to the `Config>` prompt. 8. Type **write** to save this configuration, then **reload** followed by **yes** to reboot with it.
Enable the dynamic addition of interfaces after initial configuration	Before you can add an interface dynamically, the active Network Utility configuration must have ″spare interfaces″ defined. 1. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 2. Type **set spare** and enter the number of spare interfaces you want. 3. Type **write** to save this configuration, then **reload** followed by **yes** to reboot with it.
Add an interface dynamically after initial configuration	1. Verify that you have active spare interfaces: a. From the * prompt, type **talk 5** and press **Enter** twice to reach the + prompt. b. Type **int** and verify that you have NULL interfaces. c. Type **Ctrl-p** to return to the * prompt. If you have no spare interfaces, you must follow the procedure above to add some to your configuration and reboot. 2. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 3. Use **add dev** and **net** commands to configure a new interface, as described in the initial configuration procedures. Note the new logical interface number assigned by the **add dev** command. 4. Use the **protocol** and **feature** commands to move to Config subprocesses and configure protocol information relating to the new interface. 5. Type **Ctrl-p**, **talk 5**, and press **Enter** twice to reach the + prompt. 6. Type **activate int** and give the new logical interface number. The system activates the new interface dynamically. 7. If you want to save the new interface configuration so that it will survive a reboot, go back to talk 6 and type **write** to write the modified configuration to disk. Or, make the corresponding changes at the Configuration Program and download the revised configuration to the Network Utility.

Table 7. How to Configure Physical Adapters and Interfaces (continued)

Task	How to do it
Dynamically change interface configuration	1. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 2. Type **list dev** to see the logical interface number for the interface you want to change. 3. Type **net** *logical interface number* to enter the Config subprocess for the specific interface type. Enter commands to change the configuration of the interface. Type **exit** to return to the Config> prompt. 4. Use the **protocol** and **feature** commands to reach protocol and feature Config subprocesses. Enter commands to change parameters related to the interface. 5. Type **Ctrl-p**, then **talk 5** and press **Enter** twice to reach the + prompt. 6. Type **reset** and enter the logical number of the interface you just reconfigured. Network Utility takes the interface down and brings it back up using the modified configuration. 7. If you want to save these configuration changes so that they will survive a reboot, go back to talk 6 and type **write** to write the modified configuration to disk. Or, make the corresponding changes at the Configuration Program and download the revised configuration to the Network Utility.

Managing Physical Adapters and Interfaces

Table 8 describes how to perform tasks relating to managing physical adapters and interfaces.

Table 8. How to Manage Physical Adapters and Interfaces

Task	How to do it
Look at the status of an interface	1. From the * prompt, type **talk 5** and press **Enter** twice to reach the + prompt. 2. Type **config** to see information about the software and, at the end, the current state of all interfaces. If the display output pauses with `--More--` displayed, press the space bar to see the next screen of output. 3. Type **int** to see slot and port numbers and activation counts for interfaces. 4. Type **stat** to see packet and byte statistics for interfaces. 5. Type **err** to see error counts for interfaces. 6. Type **queue** and **buff** to see buffer counts for interfaces. 7. Type **net** *logical interface number* to enter the Console subprocess for the specific interface type. Use the commands in that subprocess to display type-specific interface status information.
Recycle (disable/enable) an interface	1. From the * prompt, type **talk 5** and press **Enter** twice to reach the + prompt. 2. Type **int** to see the logical ″net″ number for the interface you want to recycle. 3. Type **disable int** *logical interface number* to take the interface offline dynamically. 4. Type **test** *logical interface number* to bring the interface back up.

Table 8. How to Manage Physical Adapters and Interfaces (continued)

Task	How to do it
Recycle (disable/enable) an adapter	**Note**: If you intend to remove the adapter while it is disabled (the standard ″hot plug″ procedure), you should also refer to the ″Removal and Replacement Procedures″ chapter in the *2216 Nways Multiaccess Connector and Network Utility Service and Maintenance Manual*. 1. From the * prompt, type **talk 5** and press **Enter** twice to reach the + prompt. 2. Type **disable slot** *slot number*, where 1 is the left-hand slot and 2 is the right-hand slot. This disables all interfaces on the adapter in that slot. 3. Type **enable slot** *slot number* to activate all interfaces on the adapter in that slot.

Basic IP Configuration and Operation

Table 9 describes basic configuration and operation tasks for IP adapters and interfaces.

Table 9. Basic IP Configuration and Operation

Task	How to do it
Add an IP address to a network adapter	1. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 2. Type **prot ip** to reach the IP Config subprocess. 3. Type **li addr** to see currently configured IP addresses. 4. Type **add addr** to add an IP address. Supply the logical interface (net) number of the interface, the IP address, and the address mask. 5. If you want to activate this and other IP configuration changes in the running Network Utility: a. Type **Ctrl-p**, then **talk 5** and press **Enter** twice to reach the + prompt. b. Type **prot ip** to reach the IP Console subprocess. c. Type **int** to see currently active interface IP addresses. d. Type **reset ip** to activate the new address. e. Type **int** to verify the new address.
Set the IP address of the PCMCIA EtherJet adapter	1. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 2. Type **system set ip** and supply the following information (defaults are the current values of these parameters): • IP address - the address to be used for the EtherJet adapter • IP netmask - the network mask for that address • IP gateway address - the address of the IP workstation you are likely to communicate with, or the router you use to reach that workstation. Any changes you make take effect immediately and are stored in Network Utility nonvolatile memory. These addresses are not part of the Network Utility configuration. You can also set the EtherJet IP address from the firmware. Follow the procedure below for EtherJet Ping, but select option 1 **IP Parameters**, instead of option 3 **Ping**.

Table 9. Basic IP Configuration and Operation (continued)

Task	How to do it
Add a static route	1. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 2. Type **prot ip** to reach the IP Config subprocess. 3. Type **li route** to see currently configured routes. 4. Type **add route** to add a static route. Supply the information requested.
Ping and traceroute from a network adapter	1. From the * prompt, type **talk 5** and press **Enter** twice to reach the + prompt. 2. Type **prot ip** to reach the IP Console subprocess. 3. To ping an address with default parameters, type **ping** *ip address*. To modify parameters, type only **ping** and respond to the prompts. Type **Ctrl-c** to end the ping. 4. To trace the route to an address with default parameters, type **trace** *ip address*. To modify parameters, type only **trace** and respond to the prompts. Type **Ctrl-c** to end the traceroute.
Ping from the PCMCIA EtherJet adapter	1. Use one of the procedures in “Boot Options: Fast Boot and Reaching Firmware” on page 45 to reach the firmware main menu. 2. Bring up the panel from which you do a Ping a. Select option 4, **Utilities**. b. Select option 11, **Remote Initial Program Load Setup**. c. Select option 3, **Ping**. d. Select the PCMCIA Ethernet interface. 3. Enter the IP addresses you want to use for the ping (these temporarily override the configured addresses) and press **Enter**.

Managing the Command-Line Configuration

Table 10 describes how to manage the command-line configuration.

Table 10. How to Manage the Command-Line Configuration

Task	How to do it
Erase the configuration for a protocol, or for all protocols	1. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 2. Type **clear ?** to see a list of sets of configuration information you can clear with a single command. 3. Type **clear** *protocol name* to clear information for a particular protocol, or **clear all** to clear information for all protocols (but not device information). These commands change the current configuration in memory but do not affect the operational state of the Network Utility.

Table 10. How to Manage the Command-Line Configuration (continued)

Task	How to do it
Erase the configuration for an interface, or for all interfaces	1. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 2. Type **del int** if you want to delete the configuration for a particular interface, including all protocol configuration related to that interface. 3. Type **clear dev** if you want to delete the configuration for all interfaces. This command does not clear associated protocol information, so you would normally use it with **clear all** to completely erase a configuration. These commands change the current configuration in memory but do not affect the operational state of the Network Utility.
Activate the entire current talk 6 configuration	1. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 2. Type **write** to write the current configuration in memory to disk in the next available configuration file position of the active bank. 3. Type **reload** then **yes** to reboot Network Utility and activate that configuration. If you activate a configuration with no protocol or no device information, the Network Utility will enter config-only mode. You will have to define one protocol and one interface and reboot before the Network Utility can be fully operational.

General Status Monitoring

Table 11 describes how to perform general status monitoring tasks.

Table 11. How to do General Status Monitoring

Task	How to do it
Look at CPU utilization	1. From the * prompt, type **talk 5** and press **Enter** twice to reach the + prompt. 2. Type **perf** to reach the performance monitoring Console subprocess. 3. Type **list** and verify that the CPU Monitor State is ENABLED. This is the default setting for Network Utility. If the state is not ENABLED, type **enable cpu**. 4. Type **report** to see recent CPU utilization statistics. The most current snapshot is the value ″Most recent short window.″ 5. If you want CPU utilization to be reported every so often as an event message you can monitor with talk 2, type **enable t2**. Type **Ctrl-p** and **talk 2** to watch CPU utilization messages being generated. Type **Ctrl-p** to exit talk 2. 6. If you want the talk 2 CPU reporting to be continued after your next reboot, move to talk 6 and repeat the above commands. Or, configure the same settings on the CPU Utilization panel from the Configuration Program, and transfer the updated configuration to the Network Utility.

Table 11. How to do General Status Monitoring (continued)

Task	How to do it
Look at memory utilization	1. From the * prompt, type **talk 5** and press **Enter** twice to reach the + prompt. 2. Type **mem** to see current global memory statistics. This command reports the total physical installed memory and details about the part of memory used by the routing function. The routing function includes all network protocols and features except APPN and TN3270 server. 3. If you are running APPN or TN3270 server, type **p appn** to reach the APPN Console subprocess. Type **mem** to see current APPN memory statistics and threshold states. TN3270 server usage is included in these statistics, even if you are running only subarea TN3270 host attachment.
Turn on default ELS messages	1. From the * prompt, type **talk 5** and press **Enter** twice to reach the + prompt. 2. Type **event** to reach the event logging Console subprocess. 3. Type **disp sub all** to activate the STANDARD level of logging for all defined subsystems. This includes error messages and uncommon informational messages. 4. Type **Ctrl-p** then **talk 2** to watch any messages being generated, and **Ctrl-p** to exit talk 2. 5. If you want these settings to be maintained after your next reboot, move to talk 6 and repeat the above commands. This will make the settings part of your configuration.

Boot Options: Fast Boot and Reaching Firmware

Table 12 describes how to perform the boot option tasks for fast boot and reaching firmware.

Table 12. Boot Options: Fast Boot and Reaching Firmware

Task	How to do it
Minimize boot time in a test environment	1. Type **talk 6** and then **boot** to reach the boot Config subprocess. 2. Type **en fast** to enable the fast boot option. The next time you reboot the Network Utility, it will boot more quickly by skipping some of the power-on diagnostics. This option is not recommended for production environments. You can use **dis fast** to go back to the normal full diagnostic mode.

Table 12. Boot Options: Fast Boot and Reaching Firmware (continued)

Task	How to do it
Reach the firmware if you have a directly connected terminal console	1. Make sure your terminal emulation screen size is set to 24 rows by 80 columns, or set auto-wrap off in your terminal emulator. 2. From the * prompt, type **reload**, then **yes** to the confirmation message. Start watching the boot status messages closely. 3. When you see the message `Starting Boot Sequence` followed by `Strike F1 key now to prematurely terminate Boot`, type **Ctrl-c** or **F1** immediately. To make sure you do not miss this message, you can start holding down **Ctrl-c** at any time after the start of system board diagnostics. Continue to hold **Ctrl-c** until you see the firmware main menu or the prompt for a supervisory password. 4. Within a few seconds of the `Strike F1 key now to prematurely terminate Boot` message, you should be either at the firmware main menu or at a prompt for a supervisory password. If neither of these appear and you see disk load messages, you waited too long and missed the time window for typing **Ctrl-c** or **F1**. Wait for the boot sequence to complete, then repeat steps 2 and 3 of this procedure. Or, use the dial-in procedure to ensure you will stop in the firmware without having to press a key at the right time. 5. If the system prompts you for a supervisory password, enter the current password, originally set to ″2216″ at the factory. The system then presents the firmware main menu.
Reach the firmware if you have a dialed-up terminal console	1. Make sure your terminal emulation screen size is set to 24 rows by 80 columns, or set auto-wrap off in your terminal emulator. 2. From the * prompt, type **talk 6** and press **Enter** twice to reach the `Config>` prompt. 3. Type **boot** to reach the boot Config subprocess. 4. Type **disable auto-boot** to select the mode where a boot sequence will always stop at the firmware. If you are prompted with the duration (once/always) prompt, select whether you want to stop in the firmware with only the next reboot or with every reboot hereafter. 5. Type **Ctrl-p** to reach the * prompt, then **reload yes** to reboot Network Utility. The reboot causes you to lose your dial connection. 6. After a few minutes, dial back in and you should be either at the firmware main menu or at a prompt for a supervisory password. 7. If the system prompts you for a supervisory password, enter the current password, originally set to ″2216″ at the factory. The system then presents the firmware main menu. If you were given the duration (once/always) prompt and you selected always or if you were not given that prompt, do an **enable auto-boot** the next time you reach the operational code.
Boot from the firmware into the operational code	1. From within the firmware menu structure, press **Esc** as required to reach the firmware main menu. 2. If you want to continue the current boot sequence up into the operational code, press **F9** (Start OS). If you want to completely reboot starting from power-on diagnostics press **F3** (Reboot). This will cause you to lose your connection if you are dialed into the Network Utility PCMCIA modem or system card service port. 3. Dial back in if necessary, or just monitor the disk load messages. Press the space bar to obtain the command prompt if the system asks you to do so.

Part 2. Learning About Network Utility

Chapter 5. A Guided Tour through the Command-Line Interface

This chapter is a tutorial to walk users who are new to IBM routing products through the concepts and basic navigation of the Network Utility command-line interface. It covers:

- Basic concepts of adapter and port numbering
- How to move to different parts of the system and what each is for
- Example tasks and commands from different processes
- How to navigate menus and issue commands
- How to configure, query status, and watch the system log
- Basic concepts of saving and activating configuration changes
- What firmware is, how to get to it, and a few things you can do with it
- Ways the automatic command completion function assists you with the syntax for commands entered at the command line

The tutorial text makes the most sense if you follow it from beginning to end with the same Network Utility.

If you already have experience with the IBM 2216, you will find the Network Utility interface to be nearly identical. This is also true for IBM 2212 users, except for the firmware interface. IBM 2210 users will find familiar prompts and menu navigation, but differences in areas including configuring adapters, saving configurations, and rebooting the product.

Prompts and Processes

If you followed one of the initial configuration procedures in “Chapter 3. Performing the Initial Configuration” on page 25, you have configured your Network Utility and booted it into normal operating mode. Your user console should show the asterisk (*) command prompt.

In normal operating mode, the routing function in Network Utility is running. You as the operator can use the command-line interface to look at and modify the configuration, view the active system status, look at the message log, and so on. You navigate to different parts of the command-line interface to perform these different tasks, and the * prompt is the root of the navigation tree.

Type **?** from the * prompt to see the commands available from this point:

```
  *?
CONFIGURATION       (Talk 6)
CONSOLE             (Talk 5)
EVENT Logging System (Talk 2)
ELS Console         (Talk 7)
LOGOUT
PING   <IP Address>
RELOAD
TELNET to IP-Address <this terminal type>
---------------------------------------------------
DIAGS hardware diagnostics
DIVERT output from process
FLUSH output from process
HALT output from process
INTERCEPT character is
```

```
MEMORY statistics
STATUS of process(es)
SUSPEND command completion
TALK to process
*
```

Although each of these commands has its purpose, you will use two of them far more than any of the others:

talk Attaches your console to one of various processes, or ways of viewing the system.

reload Reboots the Network Utility.

To use the **talk** command, type **t** *n*, where *n* (a *process id*) usually takes one of the following values:

6 To look at and modify the configuration (the *Config* process)

5 To look at current system status, actively control the state of the running system, and activate dynamic configuration changes (the *Console* process)

2 To look at a rolling log of informational and status messages (the *Monitor* process)

To undo the **talk** command and move from inside any process directly back to the * prompt, type **Ctrl-p**.

MAS V3.3 introduced more natural commands that perform the same function of the **talk** command. In place of **talk 6**, you can simply type **config**. Likewise, **console** can substitute for **talk 5** and **event** for **talk 2**.

The following three sections describe each of the major processes and explain some of the tasks you can perform inside each process. Along the way, you will become familiar with moving around between processes and menus, and entering commands.

Configuring (using talk 6, the Config process)

From the * prompt, type **t 6** or **config** to enter the command-line process for configuring the Network Utility:

```
*         <Enter>
*t 6
Gateway user configuration
Config>   <Enter>
Config>
```

Now that you are inside the Config process, the command prompt has changed from * to `Config>`. Both the Config and Console processes have unique prompts so you can tell at a glance which process you are in. The status message `Gateway user configuration` shows up only the first time you enter the Config process following a reboot (″gateway″ is used as a synonym for ″router″ in various places in the system).

When you have been in a process before and reentry it using the **talk** command, the system gives you a blank line instead of an immediate command prompt. Press **Enter** and you are returned to where you were the last time you were inside that process:

```
Config>   <Ctrl-p>        <---- leave Config and go back to *
*         <Enter>
*t 6                      <---- go back into Config
          <Enter>
Config>                   <---- we're back at the main Config prompt
```

When you are working inside the Config process, you are changing how the Network Utility is configured to operate. With a few exceptions, these changes have no effect on the running state of the router. To activate talk 6 changes you must either:

- Issue one of several commands to activate a set of changes, or
- Save the changes to the hard disk and reboot the system

As you follow this tutorial you will see examples of both methods.

Command Overview

At the main `Config>` prompt, type **?** to see an alphabetical list of the commands available to you:

```
Config>?
ADD (device, user)
BOOT and load file functions
CHANGE (device, password, user)
CLEAR configuration information
DELETE (interface, user)
DISABLE (interface, console-login, etc)
ENABLE (interface, console-login, etc)
EVENT logging system and messages
FEATURE (non-protocol and network features)
LIST (devices, configuration, patches, users)
LOAD (add, delete, list)
NETWORK interface configuration
PATCH global configuration parameters
PERFORMANCE monitor
PROTOCOL configuration
QCONFIG (quick configuration)
SET system-wide parameters
SYSTEM
TIME of day parameters
UNPATCH global configuration parameters
UPDATE
WRITE
Config>
```

Some of these commands are for actually configuring the functions of the box, and others are for configuration management and system administration. To give you a feel for the types of things you do under talk 6, the following list groups key commands by user task:

- Configuring adapters and ports

  add device
  Configures a single adapter slot and port

  change device
  Moves or copies a slot configuration to another slot

  delete interface
  Deletes a single interface (adapter port) and associated protocol information

  disable/enable interface
  Controls whether a specific interface will be activated

list device
Shows all configured interfaces

net *interface number*
Goes to the subprocess to configure the specified interface, below the protocol level

set data-link
Changes a newly-added WAN adapter port from the default of PPP to Frame Relay, SDLC, SDLC Relay, or X.25

system set/display ip
Sets/shows the IP parameters for the PCMCIA LAN adapter

- Configuring protocols and features

protocol *name*
Goes to the subprocess to configure the specified protocol

feature *name*
Goes to the subprocess to configure the specified feature

- Managing configurations and software loads

boot Goes to the subprocess to manage transfer and usage of configuration files and software loads on disk

clear Can clean out all device, all protocol, or specific pieces of the current configuration in RAM

write Saves the current configuration in RAM to the hard disk

- Configuring to monitor the box

event Goes to the subprocess to configure which event logging system (ELS) messages are active

performance
Goes to the subprocess to configure CPU utilization monitoring

- Administering the system

add/change/delete/list user, change password
Administers user IDs for controlled console access

disable/enable console-login
Controls remote access to console

set host/prompt/contact/location
Sets up a host name, prompt prefix, contact person, or location

time Sets the time and the time format, or whether to get the time from a remote host

- Servicing the software

disable/enable/set dump, reboot
Controls dumping and rebooting if the Network Utility box abnormally terminates operation

patch, unpatch
Controls specialized software functions to get around problems in specific user environments

system retrieve
Sends a compressed system dump of the router to a server

system view
Shows information about current dump files

The following examples show how to use some of these talk 6 commands to perform basic configuration tasks. As you work through the examples, you will gain experience not only with the tasks shown, but also generally with moving around through menus and issuing commands. The examples begin with a task that may already be familiar if you used the command line procedure for initial configuration.

Example: Configuring a Port on an Adapter

In the running example used throughout this tutorial, the user first booted a Network Utility with the following configuration:

- ESCON adapter in slot 1, IP not configured
- Token-Ring adapter in slot 2, port 2 configured with IP address 192.1.1.8

If you want to follow this example, use **clear dev** to erase your own device configuration[6] and then use **add dev** and **del int** to enter the ESCON/TR device configuration, as shown below.

From the `Config>` prompt, type **list device** (or **li dev**, abbreviated) to see the adapters and ports that are defined in the current configuration. If you have no configuration, or no adapter ports defined, **li dev** gives no output but simply reissues the user prompt. Because you have cleared all devices, you can add one. Type **add dev ?** to see a list of all the adapter types you can add:

```
Config>clear dev
You are about to clear all Device configuration information.
Are you sure you want to do this? ? [No]: yes
Device configuration cleared
Config>li dev
Config>add dev ?
ATM            1-port 155 Mbps ATM adapter
EIA-232E       8-port EIA-232E/V.24 adapter
ESCON Channel  1-port ESCON Channel adapter
ETHERNET       2-port Ethernet adapter
ETH100         1-port 10/100 Mb Ethernet adapter
FDDI           1-port FDDI adapter
HSSI           1-port HSSI adapter
PCA            1-port Parallel Channel adapter
TOKEN-RING     2-port Token-Ring adapter
V35/V36        6-port V.35/V.36 adapter
X21            8-port X.21 adapter
Config>
```

Use the **add dev** command to configure a single port on a single adapter. For a multi-port adapter, you must specify which port you are adding to the configuration, and reissue the command for each port you want to have active. Here we add a single-port ESCON adapter, and both ports of a 2-port Token-Ring adapter:

```
Config>add dev esc
Device Slot #(1-2) [1]? 1
Adding ESCON Channel device in slot 1  port 1 as interface #0
Use "net 0" to configure ESCON Channel parameters
Config>add dev tok
Device Slot #(1-2) [1]? 2
Device Port #(1-2) [1]? 1
Adding Token-Ring device in slot 2  port 1 as interface #1
Use "net 1" to configure Token-Ring parameters
Config>add dev tok
Device Slot #(1-2) [1]? 2
Device Port #(1-2) [2]? 2
```

[6] Normally, you use **clear dev** only in conjunction with **clear all**, which clears out protocol information.

```
Adding Token-Ring device in slot 2  port 2 as interface #2
Use "net 2" to configure Token-Ring parameters
Config>li dev
Ifc 0     ESCON Channel                     Slot: 1   Port: 1
Ifc 1     Token-Ring                        Slot: 2   Port: 1
Ifc 2     Token-Ring                        Slot: 2   Port: 2
Config>
```

To specify the adapter type, you type on the same line as **add dev** the first few characters of the words in the left column of the **add dev ?** output list (enough characters to distinguish the adapter type you want). When prompted, you must supply the slot and (for multi-port adapters only) the port number. Slot and port numbering is fixed as follows:

- The two adapter slots on a Network Utility are numbered 1 and 2, from left to right as you look at the front of the box.
- Ports on multi-port LAN adapters are numbered 1 and 2 and are labelled on the adapter face.
- Ports on multi-port WAN adapters are numbered starting with 0 and are labelled on the connectors at the end of the adapter cable.

The **add dev** command makes sure that you do not try to add two different adapters in the same slot, add an adapter to a slot that does not exist, or specify a port number that does not exist on a given adapter. It does ***not*** validate the device type you select against the adapters that are physically installed in your Network Utility. This allows you to configure adapters you have not yet installed, or produce a configuration for a different Network Utility. The system validates device configuration only when you boot up with a particular configuration or try to activate an interface dynamically. The system reports mismatches through the LEDs on the front of your adapter, as well as from a locally viewable event log. You can also type commands to see adapter status, as you will see later in this tutorial.

Logical Interface Numbers

In response to your **add dev** command, the Network Utility assigns a logical *interface number* or *net number* to the port you have just added. This is the key number by which you refer to this interface on every other command in the system. Only the **add dev** command uses physical slot and port numbers; all other commands use the logical interface number. When you subdivide a physical (″base″) port such as ESCON into multiple virtual interfaces, each virtual interface also has an interface number. As shown above, you can use the **li dev** command to see the interface number for every physical and virtual interface.

Example: Deleting an Interface

If you make a mistake and want to undo the **add dev** command, or want to delete adapter/port configuration for any reason, use the **delete interface** command. (It is not named ″delete device″, because it deals with logical interface numbers and not adapter slot/port numbers.) To continue the example, assume that you want to use only port 2 of the Token-Ring adapter. Delete port 1 (which happens to be interface 1) as follows:

```
Config>li dev
Ifc 0     ESCON Channel                     Slot: 1   Port: 1
Ifc 1     Token-Ring                        Slot: 2   Port: 1
Ifc 2     Token-Ring                        Slot: 2   Port: 2
Config>del int
Interface number? 1
Interface being deleted... please be patient.
```

```
The router must be restarted
Interface 1 deleted successfully
Config>li dev
Ifc 0     ESCON Channel                     Slot: 1   Port: 1
Ifc 1     Token-Ring                        Slot: 2   Port: 2
Config>
```

Note that Token-Ring port 2 has now become logical interface 1. If there had been other interfaces with numbers higher than 1, these numbers also would have been decremented by 1. If you want to delete every interface in a configuration, just delete interface 0 repeatedly until there are no more interfaces.

In addition to device configuration itself, it is normal to have protocol configuration that is associated with a particular interface. When you delete an interface using the **del int** command, the system also deletes all protocol configuration associated with that interface, and renumbers all protocol configuration associated with renumbered interfaces[7]. You need to reboot Network Utility for a **del int** operation to take effect in the running system.

Example: Setting the Host Name using Menus

To look more closely at how to issue commands in general, try something simple, like using the **set** command to set up a name (″host name″) for this Network Utility.

Note: This example assumes you are running with Command Completion disabled. See “Automatic Command Completion” on page 37 to understand how the Network Utility can provide automatic command completion.

First, try the command by itself:

```
Config>set
Command not fully specified
```

This error message reports that the **set** command is backed by a menu of additional keywords, and you need to type more keywords until you form a complete command that will perform an action. Anytime you are at a menu (as you have seen already), you can type **?** to see the available commands or keywords to type. If you are just trying to remember command keywords, it is usually much faster to move around typing **?**, than it is to look up the command in a manual. In this case, the options are:

```
Config>set ?
CONTACT-PERSON
DATA-LINK
DOWN-NOTIFY
GLOBAL-BUFFERS
HOSTNAME
INACTIVITY-TIMER
INPUT-LOW-WATER
LOCATION
PACKET-SIZE
PROMPT
RECEIVE-BUFFERS
SPARE-INTERFACES
```

As you can see, the **set** menu includes a mix of data items: some for system administration, some for node tuning, and so on. In Network Utility, node tuning options are defaulted for you and you should not have to change them.

[7] The **clear dev** command does not perform this function, so you should use it only when you are also clearing protocol information by hand.

Back to the task, the keyword you want is clearly ″hostname″. You can abbreviate any menu item (command name or keyword) to the number of characters needed to make it unique, so shorten ″hostname″ a bit:

```
Config>set host
Host name for this node []? rtp01
Host name updated successfully
rtp01 Config>
```

By default, the system inserts the new host name in front of all command prompts. Many users like this because it enables them to Telnet into a number of routers from a single work station and easily distinguish one router console from another. If you want to choose a different prompt prefix, you can use the **set prompt** command to do so. To reset either the host or the prompt to a null value, use the **clear host** or **clear prompt** command and reboot the Network Utility. To look at the current values, use **list config**.

Note that **set host** is an exception to the normal talk 6 rule in that it took effect immediately and did not require you to issue some sort of ″activate″ command, or to reboot the Network Utility. Very few talk 6 commands behave this way, but this one is useful because you can immediately see its effect on the user prompt.

Example: Typing Ahead

Suppose you do not like the new prompt and want to change the host name from ″rtp01″ to ″RTP01″. You can do this in a single command, as follows:

```
rtp01 Config>set host RTP01
Host name updated successfully
RTP01 Config>
```

The system did not prompt you for the host name because you typed it on the original command line. This illustrates another general rule: when a full command prompts you for input parameters, you have the option of typing them on the original command line and skipping the prompts. If you choose to skip prompts, you need to be careful to type parameters in the right order.

Example: Setting a Port Parameter Using ″net″

Now that you have configured your host name, try something a bit more complex. Suppose you noticed when you rebooted from Config-only mode, that your newly configured Token-Ring adapter port 2 did not come up. You can see what ring speed it is configured for, and change that value. This sort of low-level device-specific configuration parameter is what you use the **net** command for, as shown here:

```
RTP01 Config>li dev           <------ what were those i/f numbers again?
Ifc 0     ESCON Channel                      Slot: 1   Port: 1
Ifc 1     Token-Ring                         Slot: 2   Port: 2
RTP01 Config>       <Enter>
RTP01 Config>net 1            <------ I configure interface 1
Token-Ring interface configuration
RTP01 TKR config>   <Enter>   <------ note the new subprocess prompt
RTP01 TKR config>?            <------ what are the commands here?
EXIT
FRAME
LIST
LLC
MEDIA
SET
PACKET-SIZE bytes
```

```
SOURCE-ROUTING
SPEED Mb/sec
RTP01 TKR config>li           <------ show me what I have now
Token-Ring configuration:

Packet size (INFO field): 2052
Speed:                    4 Mb/sec       <----- It should be 16Mb/sec
Media:                    Shielded

RIF Aging Timer:          120
Source Routing:           Enabled
MAC Address:              000000000000
RTP01 TKR config>speed
Speed (4 or 16) [4]? 16       <------ change the speed here
RTP01 TKR config>li           <------ verify the new value
Token-Ring configuration:

Packet size (INFO field): 2052
Speed:                    16 Mb/sec       <----- looks good now
Media:                    Shielded

RIF Aging Timer:          120
Source Routing:           Enabled
MAC Address:              000000000000
RTP01 TKR config>ex           <------ exit the subprocess
RTP01 Config>                 <------ you are back at the main T 6 menu
```

This change to the ring speed does not take effect immediately, but requires a talk 5 command or reboot to activate it. “Dynamic Reconfiguration” on page 75 covers the basics of activating configuration changes without a reboot. Normally, you use the **net** command immediately after **add dev** to look at the default settings for the new interface and make any necessary changes before activating the port for the first time.

In this example, when you typed **net 1**, you moved into a subprocess for configuring Token-Ring interfaces. The base menu changed and the prompt also changed to let you know you were no longer at the main `Config>` menu, but one level deeper. In order to leave any subprocess menu and return to the next higher one, type **exit**. Remember also that **Ctrl-p** immediately takes you all the way out to the * prompt, and when you return to that process you reenter where you were last:

```
RTP01 Config>       <Enter>     <------ start here
RTP01 Config>net 1              <------ enter a Config subprocess
Token-Ring interface configuration
RTP01 TKR config>   <Ctrl-p>    <------ jump out
RTP01 *             <Enter>
RTP01 *t 6                      <------ go back to Config
                    <Enter>
RTP01 TKR config>   <Enter>     <------ you are back in the subprocess
RTP01 TKR config>ex             <------ exit the subprocess
RTP01 Config>                   <------ You are back where you started
```

Now try two more examples in the Config process, and then move on to the Console process. The first example shows how to reduce the time to reload the box, and the second shows how to change parameters associated with a box protocol.

Example: Enabling ″fast-boot″

From the `Config>` prompt, type **boot** to reach the subsystem for managing configurations, code loads, and boot options. “Chapter 7. Handling Configuration

Files” on page 79 gives the full background on this subsystem, so you do not need to look at all the commands here. Look under the **enable** command and try out the ″fastboot″ option:

```
RTP01 Config>boot                     <----- enter subprocess
Boot configuration
RTP01 Boot config>   <Enter>          <----- note new prompt
RTP01 Boot config>en ?                <----- list "enable" options
AUTO-BOOT-- set Unattended mode
FAST-BOOT-- bypass diags
RTP01 Boot config>en fast             <----- try out "fast-boot"
FastBoot mode is now enabled.

Operation completed successfully.
RTP01 Boot config>ex                  <----- exit the boot subprocess
RTP01 Config>
```

If you watched the console bootup messages when you powered on your Network Utility or typed the **reload** command, you may have noticed that the system runs through a number of power-on diagnostics when it is booting. While this is desirable for a production router that is rebooted infrequently and whose hardware should be validated, it does lengthen the boot time. If you are actively configuring and repeatedly rebooting a given router, you may wish to reduce the boot time by skipping these diagnostics. You have just done this with the **enable fast-boot** command. The next time you do a **reload**, it will proceed more quickly. You should undo this change using **disable fast-boot** before placing the Network Utility into production.

Note that the fast-boot mode can be controlled only by the command line and not from the Configuration Program. The system’s boot mode is stored in nonvolatile memory on the box, and is not part of the configuration file.

Example: Changing an Interface IP Address

The final Config process example uses the menus and commands of the IP protocol subprocess to change an interface IP address. As noted on page 55, this example started with a Network Utility that had an IP address configured on Interface 1 (port 2 on the Token-Ring adapter in slot 2).

```
RTP01 Config>li dev            <----- what are the intfcs again?
Ifc 0     ESCON Channel                      Slot: 1   Port: 1
Ifc 1     Token-Ring                         Slot: 2   Port: 2
RTP01 Config>p ip              <----- short for "protocol ip"
Internet protocol user configuration
RTP01 IP config>   <enter>     <----- now in IP Config subprocess
RTP01 IP config>li addr        <----- list configured IP addresses
IP addresses for each interface:
   intf     0                                     IP disabled on this interface

   intf     1   192.1.1.8        255.255.255.0    Local wire broadcast, fill 1
RTP01 IP config>change addr
Enter the address to be changed []? 192.1.1.8
New address [192.1.1.8]? 192.7.7.7
Address mask [255.255.255.0]?   <enter>
RTP01 IP config>li addr        <----- verify the change
IP addresses for each interface:
   intf     0                                     IP disabled on this interface

   intf     1   192.7.7.7        255.255.255.0    Local wire broadcast, fill 1
RTP01 IP config>ex             <----- exit IP config
RTP01 Config>
```

This is the first example of using the **protocol** command to enter the subprocess for an individual protocol. IP is just one of many protocols you could have selected, and there is a similar list of features you can access using the **feature** command. Type **list config** from `Config>` for a full list of the protocols and features you can configure, or just **p ?** or **f ?** for a quick reminder. All protocols and features work the same way: you enter the subprocess for a protocol or feature, configure it using commands specific to that protocol or feature, then **exit** to the main `Config>` prompt.

For detailed command reference material on configuring any given protocol, refer to the chapter relating to that protocol in one of the two volumes of *MAS Protocol Configuration and Monitoring Reference*. Each of these chapters provides introductory material about the protocol, and a description of each configuration and monitoring console command for that protocol. For the same information about MAS features, refer to *MAS Using and Configuring Features*.

You have now completed the overview of the Config process and its commands. You can move on to talk 5, the Console process. Remember, to leave any process you type **Ctrl-p** to reach the * prompt and then you are ready to use the **talk** command to enter another process:

```
RTP01 Config>   <Ctrl-p>
RTP01 *
```

Operating (Using talk 5, the Console Process)

From the * prompt, type **t 5** or **console** to enter the command line process for monitoring and controlling the Network Utility's active state:

```
RTP01 *   <Enter>
RTP01 *t 5


CGW Operator Console

RTP01 +   <Enter>
RTP01 +
```

Now that you are in the Console process, the command prompt has changed from * to +. The Config and Console processes and their subprocesses have unique prompts that indicate your position at a glance. The status message `CGW Operator Console` shows up only the first time you enter the Console process following a reboot. As explained with talk 6, if the system gives you a blank line when you type **t 5**, that means you have been in talk 5 before and need to press **Enter** to resume wherever you were last.

When you are working in the Console process, you type commands to view and modify the active running state of the Network Utility. You cannot modify the Network Utility's configuration files from this process. Some talk 5 commands allow you to dynamically modify configuration parameters, but these changes are lost when you reboot the Network Utility. If you have made configuration changes under talk 6, however, you can dynamically activate some of them from talk 5 without rebooting the Network Utility.

Command Overview

At the main + prompt, type **?** to see an alphabetical list of the commands available to you:

```
RTP01 +?
ACTIVATE interface
BUFFER statistics
CLEAR statistics
CONFIGURATION of router
DISABLE interface or slot
ENABLE slot
ERROR counts
EVENT logging
FEATURE commands
INTERFACE statistics
MEMORY statistics
NETWORK commands
PERFORMANCE monitor
PROTOCOL commands
QUEUE lengths
RESET interface
STATISTICS of network
TEST network
UPTIME
RTP01 +
```

Some of these commands are for viewing the status of the box and some are operator commands for actively changing that status. In addition, under each protocol and feature there is a Console subprocess containing a mixture of these two command types. The following list groups key talk 5 commands by user task:

- Viewing box status

  buffer Shows interface buffer allocation and in-use counts

  configuration
  Shows software identity, protocols/features, and interface status

  error Shows frame error counts for one or more interfaces

  interface
  Shows the interface number to slot/port mapping (the talk 5 equivalent of talk 6 **list dev**), plus self-test pass/fail counts

  memory
  Shows installed memory and in-use statistics for memory and global (non-interface) buffers

  queue Shows input and output buffer queue counts for one or more interfaces

  statistics
  Shows packet and byte counts for one or more interfaces

  uptime
  Shows elapsed time since the last reboot

- Controlling box status

  activate
  Enables a spare interface that you just configured under talk 6

  clear Resets counters for one or more interfaces

  disable
  Takes offline either a single interface, or all the interfaces in a slot

  enable
  Brings online all the interfaces in a specified slot

  reset Disables an interface and re-enables it using new configuration parameters you changed under talk 6

test Verifies and brings a single interface online

- Accessing other console subprocesses

event Go view counts and temporarily change which ELS messages are being logged

feature *name*
Go view and change status for the specified feature

network *interface number*
Go view and change status for the specified interface

performance
Go view CPU statistics and temporarily change how they are being collected and displayed

protocol *name*
Go view and change status for the specified protocol

Example: Viewing Box Status

As you did from talk 6, try some of these talk 5 commands. Those for viewing box status are all quite simple; you simply type the one-word command and look at the output:

```
RTP01 +mem
Physical installed memory:       256 MB
Total routing (heap) memory:     228 MB
Routing memory in use:             3 %

                  Total  Reserve    Never     Perm     Temp     Prev
                                    Alloc    Alloc    Alloc    Alloc
Heap memory    239390720    26616 232309212  7029792    49828     1888

Number of global buffers: Total = 1000, Free = 1000, Fair = 194, Low = 200
Global buff size: Data = 4478, Hdr = 82, Wrap = 72, Trail = 7, Total = 4644
RTP01 +   <Enter>
RTP01 +buff
                    Input Buffers        Buffer sizes                Bytes
Net   Interface     Req Alloc Low Curr   Hdr  Wrap  Data Trail Total    Alloc
0     ESCON/0       255   255  20    0    86    72  4478     0  4636  1182180
1     TKR/0         250   250   7    0    85    72  2052     7  2216   554000
```

As you can see, **mem** shows box-level status, while **buff** gives interface-level information. For all the commands that give per-interface information (**buff**, **config**, **error**, **int**, **queue**, **stat**), you can specify a list or range of interface numbers you are interested in:

```
RTP01 +int 0-1
                                              Self-Test  Self-Test Maintenance
Net   Net'   Interface    Slot-Port              Passed     Failed      Failed
0     0      ESCON/0      Slot: 1   Port: 1           0          0           0
1     1      TKR/0        Slot: 2   Port: 2           0          0           0
RTP01 +stat 1
Net   Interface      Unicast  Multicast      Bytes    Packets      Bytes
                    Pkts Rcv   Pkts Rcv   Received      Trans      Trans
1     TKR/0                0          0          0          0          0
```

Refer to the *MAS Software User’s Guide* chapter ″The Operating/Monitoring Process″ for a description of the fields in each command’s output.

Example: Viewing Interface Status

The **config** command is particularly important, because at the end of the output is the status of all specified interfaces (this example output is edited to remove blank lines):

```
RTP01 +c
Multiprotocol Access Services
NetU-TX1 Feature 1001 V3.1 Mod 0 PTF 1 RPQ 0 MAS.DE1 netu_38PB

Num Name  Protocol
0   IP     DOD-IP
3   ARP    Address Resolution
11  SNMP   Simple Network Management Protocol
29  NHRP   Next Hop Resolution Protocol

Num Name  Feature
2   MCF    MAC Filtering
7   CMPRS  Data Compression Subsystem
8   NDR    Network Dispatching Router
10  AUTH   Authentication

2 Total Networks:
Net  Interface   MAC/Data-Link       Hardware                     State
0    ESCON/0     ESCON               ESCON Channel                Not present

1    TKR/0       Token-Ring/802.5    Token-Ring                   HW Mismatch
RTP01 +
```

The Network Utility from which this example output was captured in fact has an empty slot 1 and an Ethernet adapter in slot 2. In talk 6, it does not matter if what you configure does not match the installed adapters, but when you reboot with that configuration, talk 5 will show you that your configured interfaces have not come up.

If you had configured correctly, the interface state would start with ″Testing″ then move to ″Up″, and you would be able to use the **net** command to enter an adapter-specific Console subprocess to get more detailed status information. As it is now, you get the following:

```
RTP01 +net 0
 Network interface is not available.
RTP01 +
```

Example: Accessing an Unconfigured Protocol

To view and control what's currently going on with any given protocol, use the **protocol** command to enter the Console subprocess for that protocol. As explained previously, **p ?** will generate a quick list of the protocols supported in a given software load. For example, select Data Link Switching (DLSw):

```
RTP01 +p dls                     <----- short for "protocol dlsw"
 Protocol DLSW is available but not configured
RTP01 +
```

DLSw is ***available*** because it is supported by this software load[8], but it is ***not configured*** because you never went into talk 6 and entered the commands to enable DLSw. Now that you have booted the box without DLSw in the configuration, it is not running and there is no DLSw status to view or modify from talk 5.

[8] If it had not been supported, it would not have shown up under **p ?**, and the system would not have recognized the value ″dls″.

Example: Accessing a Configured Protocol

As noted on page 55, this example started on a Network Utility already booted with an IP configuration. IP is therefore actively running, so you can enter its Console subprocess and see what commands are available:

```
RTP01 +p ip                    <----- short for "protocol ip"
RTP01 IP>?
ACCESS controls
CACHE
COUNTERS
DUMP routing tables
INTERFACE addresses
PACKET-FILTER summary
PARAMETERS
PING dest_addr [src_addr size ttl rate]
REDUNDANT Default Gateways
RESET
RIP
ROUTE given address
ROUTE-TABLE-FILTERING
SIZES
STATIC routes
TRACEROUTE dest_addr [src_addr size probes wait ttl]
UDP-FORWARDING
VRID
VRRP
EXIT
RTP01 IP>
```

If you compare this command list with that generated in talk 6 by typing **?** at the **IP config>** prompt, you see that the talk 5 and talk 6 commands are quite different. In talk 5, for example, you can initiate a **ping** to see if you can reach a given IP address from the Network Utility. Because this is an active command immediately operating on an active network interface, it does not belong in talk 6. Other commands to view active status likewise are talk 5 commands and not talk 6 commands.

Example: Dynamic Reconfiguration

In talk 6 you changed the IP address of Token-Ring port 2 from 192.1.1.8 to 192.7.7.7. Now see what value appears under talk 5:

```
RTP01 IP>int                          <----- short for "interface"
Interface  IP Address(es)   Mask(s)
  TKR/0    192.1.1.8        255.255.255.0
```

The talk 6 change had no effect on the operational state of the Network Utility, because you have not yet activated it either by explicit command or by rebooting. Use the command **reset ip** to reread the current talk 6 IP configuration and dynamically activate it in the running system:

```
RTP01 IP>reset ip
RTP01 IP>int
Interface  IP Address(es)   Mask(s)
  TKR/0    192.7.7.7        255.255.255.0
RTP01 IP>ex
RTP01 +
```

As you can see, the IP address change (and any other IP changes you made under talk 6) are now active. Most protocols have some mechanism for dynamic reconfiguration, but not every protocol has a **reset** command under talk 5. See “Dynamic Reconfiguration” on page 75 for more background on ways to do dynamic reconfiguration.

You have now seen how to issue talk 5 commands to actively query the status of the system. There is another, more passive mechanism available: viewing event messages that the Network Utility generates. To do this you use **talk 2**. As always, type **Ctrl-p** to leave the current process:

```
RTP01 +   <ctrl-p>
RTP01 *
```

Event Logging (Using talk 2, the Monitor Process)

From the * prompt, type **t 2** or **event** to attach your console to the process for viewing the Network Utility’s local message log:

```
RTP01 *   <Enter>
RTP01 *t 2
00:00:50   GW.001:

Copyright 1984 Massachusetts Institute of Technology,
Copyright 1989 The Regents of the University of California


00:00:50   GW.002: Portable CGW RTP01 Rel NetU-TX1 Feature 1001 V3.1 Mod 0 PTF 1
 RPQ 0 MAS.DE1 netu_38PB
 strtd
00:00:50   GW.005: Bffrs: 1000 avail 1000 idle   fair 194 low 200
00:00:50  DOLOG: .....Remote Logging Facility is now available.....
```

In this example, only four messages have been logged since the Network Utility was last booted. Each message has the format:

- Time stamp in the format *HH:MM:SS*

  All 4 of the above messages were logged in the same second, 50 seconds after the clock started.
- Message id in the format *SUBSYSTEM.ID*

  GW.001, GW.002, and GW.005 are ELS messages in the GW (GateWay) subsystem. DOLOG is a nonstandard, unconditional type of message you will see from time to time.
- Message body

  The body of GW.001 is the two copyright statements. The body of GW.002 is the software version statement. To look up the meaning of any particular ELS message, see the *Event Logging System Messages Guide* either on the Web or in CD-ROM format.

Unlike the talk 6 and talk 5 processes, the talk 2 process has no user command prompt. That is because you do not type commands when you are in talk 2; you simply watch messages roll by as the Network Utility generates them. You control what messages appear by enabling or disabling individual or groups of messages under the **event** subprocess of either talk 6 or talk 5. See “Monitoring Event Messages” on page 90 for an introduction to ELS concepts and controlling ELS messages.

Under talk 2, then, the only thing you would normally type is **Ctrl-p**, to return to the * prompt and move to talk 5 or talk 6. If messages are scrolling by too quickly to read, you can use **Ctrl-s** to pause scrolling, and **Ctrl-q** to resume it. Other options for capturing fast-moving event messages include:

- Activating a log file from within a PC terminal emulation program that you are using for your console
- From a UNIX or AIX workstation, Telnetting into the Network Utility to get your console connection, and *tee*ing the Telnet session into a local work station file
- Using the Network Utility's capability to log ELS messages over the network to a remote host, rather than to the local talk 2 process

These options are described in detail in the *MAS Software User's Guide* chapter ″Using the Event Logging System (ELS).″

When you enter talk 2, the system displays all the messages that have been buffered up since the last time you left talk 2. If the message buffer has been overrun or the system is currently generating messages faster than it can display them, you will see lines about ″messages flushed″ interspersed within the talk 2 scrolling output.

If you are about to enter talk 2 and you know that there is a backlog of old messages to be displayed before you can see the current messages you are interested in, use the command **flush 2** from the * prompt before typing **talk 2**. The system discards the entire backlog and talk 2 displays only messages generated after you entered the **flush** command.

Type **Ctrl-p** to exit talk 2 and return to the * prompt.

Saving the Configuration and Rebooting

If you followed through the examples in this tutorial, you have made the following talk 6 configuration changes since you began:

- Added two interfaces
- Set the host name
- Changed an interface Token-Ring speed
- Changed an interface IP address

Note: You also enabled the ″fast-boot″ option, but this change is stored in NVRAM and is not relevant here.

On a Network Utility, talk 6 changes are actually made in a RAM copy of the configuration. If you want these changes to become permanent and be used with the next reboot of the Network Utility, you need to write them to the hard disk. Two different command sequences can accomplish this task:

```
RTP01 *t 6
                <Enter>
RTP01 Config>write
Config Save: Using bank A and config number 3

<boot messages start to appear>

RTP01 Config>   <Ctrl-p>
RTP01 *reload
Are you sure you want to reload the gateway? (Yes or [No]): yes

<boot messages start to appear>
```

..... or .....

```
RTP01 *reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? (Yes or [No] or Abort): yes
Config Save: Using bank A and config number 3

<boot messages start to appear>
```

In the first sequence, the user uses the **write** command to commit changes to disk before **reload**. In the second sequence, the user does not use the **write** command and the system asks whether to save changes to disk before proceeding with the **reload**.

Which method you use is completely up to you. Many users prefer the second method because it involves less thinking and typing, but it may also be easier to forget what talk 6 changes you have made if you do not issue a **write** shortly after making them.

Firmware

Until now, the examples have always booted the Network Utility all the way up to the operational software, at either the `Config (only)>` or `*` prompt. There is one other major console user interface that you have not yet visited, that of the *firmware*. You may not need to interact much with the firmware, but you should be aware of it because it provides an alternative way to load code and configuration files onto the hard disk, and may give you a way out of a difficult problem.

Network Utility firmware is low-level software that drives the power-on and boot logic of the system. It resides in flash memory rather than on the hard disk, so in the event of a failure such as corruption of your operational software load on disk, you can retrieve new software or configuration files and get back up and running.

To reach the firmware user interface, your user console must be through local or dialed-in ASCII terminal emulation. You cannot telnet to the firmware user interface. To reach the main firmware menu, do a **reload** from the * prompt and look for these messages:

```
Starting Boot Sequence...
Strike F1 key now to prematurely terminate Boot
```

Look closely because these only appear for a few seconds each. Press **F1** when prompted, or hold down **Ctrl-c** before and during the messages to interrupt the normal boot sequence and move into the firmware.

After you interrupt the boot sequence, the system may prompt you for a supervisory password before you can see the firmware main menu. This password controls access to sensitive low-level firmware functions. Its initial value from the factory is ″2216″. You can change it only from the firmware itself, under the Utilities menu.

If you dial into the Network Utility via modem to get your console and lose your connection on **reload**, you may not be able to connect back in time to press **F1**. In this case, go to the **boot** subsystem of the Config process and issue the **disable auto-boot** command:

```
*t 6
Gateway user configuration
Config>boot
Boot configuration
Boot config>dis auto            <----- short for "disable auto-boot"
Select the duration to disable autoboot: (once, always) [always] once
```

```
AutoBoot mode is now disabled once.

Operation completed successfully.
Boot config>   <Ctrl-p>
*rel y                         <----- short for "reload, yes"

<boot messages appear>
```

With AutoBoot mode disabled, the system will stop the **reload** process at the firmware, without your having to press **F1**. Then when you connect back in, you will be at the main menu or the request for the supervisory password.

If you disable auto-boot always in talk 6 to reach the firmware or if you were not given the duration (once/always) prompt, remember to re-enable it when you reach the operational code, or you will stop in the firmware for every **reload**.

When you reach the firmware, you see a text menu like the one below at your user console:

```
Nways System Firmware
Version 3.00 built on 04/21/98 at 22:18:42 in cc3:paws_netu6e:cc3_6e
(C)Copyright IBM Corporation, 1996, 1998.  All rights reserved.
                           System Management Services

 Select one:
  1. Manage Configuration
  2. Boot Sequence Selection
  3. Select Device to Test
  4. Utilities



      Enter    -    Esc=Quit   -     F1=Help   -    F3=Reboot  -   F9=Start OS
   -------------   -------------   -------------   -------------   ------------
```

The firmware menu structure and its options are described in the *2216 and Network Utility Service and Maintenance Manual* in the chapter ″Using 2216 Firmware.″ You do not type any commands, but move through a sequence of menus by selecting options. The key tasks you may need to perform from the firmware are:

- Transfer configuration files and operational software to disk

  These functions are equivalent to the **boot** subsystem functions under talk 6. You find them in the firmware menus under ″Utilities,″ then ″Change Management.″
- Upgrade the firmware itself

  To do this, start with ″Utilities″ on the main menu, then ″Update System Firmware.″

You may want to move around the menus a little to get familiar with them. When you have completed any firmware task, press **Esc** to return to the main menu. Use one of the following options to continue:

F3=Reboot - starts the boot process all over. If you have auto-boot disabled, you will just stop in the firmware again. If you are dialed in, you will lose your connection again.

F9=Start OS - continues the boot process past the firmware up into the operational code.

You have reached the end of this Network Utility user interface tutorial. The following chapters cover a number of other important Network Utility concepts and methods, and assume you have the background provided in this chapter.

Chapter 6. Configuration Concepts and Methods

This chapter provides background information about configuring Network Utility, including:

- What it means to configure Network Utility
- The different ways configuration information is stored and transferred
- The different methods available for creating and changing configurations

"Chapter 3. Performing the Initial Configuration" on page 25 introduced the basic methods of configuring Network Utility and provided guidance on choosing between them (see "Choosing Your Configuration Method" on page 25). This chapter gives additional details about each method and discusses using both of them together.

For specific procedures and commands dealing with configuration files, see "Chapter 7. Handling Configuration Files" on page 79. Some common configuration tasks are described in "Chapter 4. Quick Reference to the User Interface" on page 35.

Configuration Basics

A Network Utility configuration is a collection of data items that control how the software operates, including such elements as:

- What interfaces to activate
- What links to bring up
- What protocols and features to make active
- What functions in a given protocol or feature to make active
- What network addresses and names to use

When you boot up a Network Utility, the system reads its configuration from a file on its hard disk, and activates interfaces and protocols according to the information in that file. You create the file in one of two ways:

- Using the command-line interface from a user terminal console

  You type commands to create configuration data items in memory and then write the configuration to the Network Utility hard disk.
- Using a graphical configuration program that runs on a PC or workstation

  You create the configuration on the workstation and then transfer it to the Network Utility hard disk.

Once the system is up and running, you can use the command-line interface to make the following types of configuration changes:

- Changes that take effect in the running system, but are not saved in a file and are therefore lost when you reboot
- Changes that take effect in the running system, are also saved in a file and are therefore maintained when you reboot
- Changes that do not take effect in the running system, but are saved in a file and become active only when you reboot

Configuration Files on Disk

The Network Utility hard disk is organized to contain two logical *banks*, one for each of two operational code (software) loads. This allows you to have the active code load in one bank, transfer a new load to the other bank, test it, and be able to back off to the original load if necessary. The two banks are referred to as Bank A and Bank B.

Each of the two banks has room for four configuration files. You can select to boot up the code load in Bank A with any of the 4 configuration files in Bank A. The same holds true for Bank B. To use a Bank A configuration file with the Bank B code load, you must first copy the Bank A configuration file to one of the four file positions in Bank B.

There are four ways to transfer a configuration file into a bank on the hard disk:

1. Use the talk 6 command **write** to store the current configuration in RAM out as a disk file.

   Use this command if you are configuring your Network Utility with the command-line talk 6 process, rather than with the Configuration Program.

   Note: If the term ″talk 6″ is unfamiliar to you, use “Chapter 5. A Guided Tour through the Command-Line Interface” on page 51 as a tutorial on the command-line interface.

2. Use TFTP or Xmodem to transfer the configuration file from a local server (PC or workstation) directly onto the hard disk.

   You can transfer a configuration file in, whether the file was created from the Configuration Program or was previously transferred from this or another Network Utility.
3. Use SNMP to transfer configuration data from the Configuration Program into RAM, and then onto the hard disk.

   You initiate the file transfer from the Configuration Program. This method is available only from the Configuration Program.
4. Copy a configuration file from one bank to the other.

   You initiate copies and other configuration file management operations from the Network Utility console under talk 6 in the **boot** subprocess.

See the section “Loading New Configuration Files” on page 82 in “Chapter 7. Handling Configuration Files” on page 79 for specifics on these operations.

Configuration Methods

Command-Line Interface

To use the command-line interface, first bring up a local or remote console to a Network Utility. For details on how to do this and reach the * or `Config (only)>` prompt, refer to “Chapter 2. Bringing Up a User Console” on page 15.

If you have an active console at the * prompt, use **talk 6** to access the Config process. If you are at the `Config (only)>`, the Config process is the only process available to you. From the Config process, you navigate menus and issue commands to configure interfaces and protocols, and write these changes to configuration files on the Network Utility hard disk.

In most cases, you use the command-line interface to configure only the Network Utility to which you are attached. But you could easily use a single Network Utility to produce configuration files to be transferred into other Network Utilities. Simply use the **write** command under talk 6 to store a configuration to a disk file, then use **tftp put** under the boot subprocess to transfer the file off the Network Utility. From then on, you have a file that can be loaded into the target Network Utility just as if it had come from the Configuration Program.

One option available only from the command line is Quick Config. As described in step 3 on page 27, Quick Config guides you through an initial configuration of a subset of the protocols in Network Utility. The system asks you questions, instead of the normal mode where it waits for you to type commands.

The ability to dynamically activate configuration changes without rebooting the Network Utility is also exclusive to the command-line interface. “Example: Dynamic Reconfiguration” on page 65 described using talk 5 to activate an IP address change made under talk 6. “Dynamic Reconfiguration” on page 75 gives more background on the dynamic reconfiguration capabilities of Network Utility.

Configuration Program

Network Utility is supported by the same graphical configuration program that you can use to configure the 2216-400. You run this program on a PC or workstation and send the configurations you produce to one or more 2216s or Network Utilities. A version of the 2216/Network Utility Configuration Program is available for each of the following operating systems:

- Microsoft™ Windows 95 or Windows NT
- IBM AIX
- IBM OS/2

IBM distributes major releases of the Configuration Program on CD-ROM and on the Web. Regular maintenance PTFs are available only on the Web. The *Configuration Program User’s Guide* describes system requirements and contains instructions for installing and using the program.

Support for Network Utility and 2216-400

When you start a new configuration with the Configuration Program, it presents a drop-down list for you to select whether the new configuration is for a 2216-400 or for a Network Utility. Your choice affects the following:

- The number of adapter slots you can configure
- The types of adapters you can configure (Network Utility supports a subset of the full list of 2216 adapters)
- The protocols and features you can configure (Network Utility supports a subset of the full MAS function)
- The default value for a variety of tuning parameters (Network Utility is preset for its intended applications)

Configurations for the 2216-400 and Network Utility are not interchangeable.

Configuration File Formats

The Configuration Program deals with three different formats of configuration files:

- .CSF files: contain data in a format that is native to the Configuration Program.

You use this format with the *Configuration* pull-down commands **Open**, **Save**, **Save as**, and **Delete**. Contents are software release-dependent; the Configuration Program automatically migrates data items when you do an **Open**.

- .CFG files: contain data in a format that is native to the router.

  You use this format when you want to create a file to transfer to the router, or when you want to read in a file you have transferred from a router.
- .ACF files: contain data in an ASCII flat file format

  You can write your configuration out into an ASCII flat file, make changes to it with a text editor, and read it back in.

Transferring and Activating Configurations

There are two ways to transfer a configuration from the Configuration Program to a Network Utility:

1. Create a router-format (.CFG) file, transfer it (possibly using FTP) to a server near the Network Utility, then retrieve it with Xmodem or TFTP onto the Network Utility's hard disk. The configuration becomes active when you select it and reboot the Network Utility.
2. Initiate a Configuration Program ″send″ operation. The Configuration Program uses SNMP to send individual data items (not a true file) into the Network Utility. The Network Utility clears the active memory copy of its current configuration, receives these data items, and then writes them to disk in a new file. Before you do the ″send,″ you select at the Configuration Program whether the Network Utility should be rebooted with the new configuration, and if so when. The configuration you sent becomes active only upon reboot.

Note that with each method, you transfer and activate an entire Network Utility configuration. There is no mechanism for the Configuration Program to dynamically send a small configuration change and activate it at the Network Utility without requiring a reboot of the Network Utility. You can only perform this type of dynamic reconfiguration using the command-line interface.

Other Configuration Program Features

Features of the Configuration Program include:

- Timed restart

  When you use the Configuration Program's facility to send a configuration to a router, you can specify the date and time you want the router to restart and use the configuration.
- Multiple router send

  You can create a list of target routers to receive configuration files, with the same or different configuration files, restart times, and so on, for each router.
- Command-line facility

  You can use the workstation operating system command line, from which you start the Configuration Program, to automate configuration operations that are available in the program. You place arguments on the original command line or in an argument file, and the Configuration Program uses them to direct its operation.

  From AIX, it is not necessary to have the operating system graphical environment (for example, Xwindows) installed to use this facility. You start the Configuration Program using the **headless** command.
- ASCII file support

You can use the Configuration Program to create and read configuration files in ASCII format. You can also convert configuration files from one format to another. An ASCII configuration file may be useful if you want to alter many configurations at one time without having to load the configurations into the Configuration Program. This feature is not intended to be used to create new configurations or to make major modifications to existing configurations.

- Online Help

  The Configuration Program supports an extensive set of help files. Press **F1** when you are positioned on any data item, and you will see a pop-up window describing the item and giving its default value and allowable range.

Dynamic Reconfiguration

The ability to dynamically modify configuration parameters without rebooting the Network Utility is available only from the command-line interface. Table 13 summarizes the different ways you can change configuration parameters from the command line, whether a change affects the running system before a reboot, and whether the change is active following a reboot. The column ″Choose Write to Disk″ indicates whether you issued the **write** command from the main talk 6 menu to save the configuration to disk, or requested a disk save after issuing the **reload** command.

Table 13. Dynamic Reconfiguration Options

Method	Choose Write to Disk	Affects Running System	Active After Reboot
Change in talk 6	Yes	No (Note 1)	Yes
	No	No (Note 1)	No
Change in talk 5	Not applicable	Yes	No
Change in talk 6, then activate in talk 5 (Note 3)	Yes	Yes (Note 2)	Yes
	No	Yes (Note 2)	No

Note:

1. The Network Dispatcher feature is an exception to this rule; its talk 6 changes take effect immediately.
2. The change takes effect when you issue the activate command, not when you change the parameter (unlike a direct talk 5 change).
3. The APPN protocol is an exception to this rule; you activate its talk 6 changes from talk 6 instead of talk 5.

As you can see, the general rule is that talk 6 changes become active following reboot or a talk 5 command to activate them. Talk 5 commands become active immediately but are lost upon reboot.

Not every configuration data item can be changed in all of the above ways. It depends on the part of the system (protocol, interface, and so on) to which a given data item belongs. For example, DLSw, SNMP, and ELS configuration all support most of the same commands in talk 6 and talk 5. You can make a change in either place depending on the permanence you want for the change. There is no talk 5 command to activate talk 6 changes, because a talk 5 command exists to make the same change.

In IP, however, there are no talk 5 commands corresponding to talk 6 commands. You use **reset ip** in talk 5 to activate the current talk 6 configuration. Interface reconfiguration is also activated using a single talk 5 command, because it involves taking the interface down and up.

See “Configuring Physical Adapters and Interfaces” on page 39 for a few examples of common dynamic reconfiguration tasks involving adapters and interfaces.

Combining Configuration Methods

If you decide to use only the command-line interface for configuration, you never need to use the Configuration Program. If you use the Configuration Program, you still need to use the command-line Config process for several reasons:

- For some protocols, talk 6 is the only way to view the Configuration Program configuration on an active Network Utility.
- There are a few configuration items, such as ELS messages and the PCMCIA EtherJet addresses, that are accessible only by talk 6 and not from the Configuration Program.
- The command line is the only way to make dynamic configuration changes.

To use a combination of the Configuration Program and talk 6, you must keep the .CSF file at the Configuration Program synchronized with the configuration information at the Network Utility. A typical scenario might be:

1. Do the initial configuration at the Configuration Program.
2. Transfer this configuration to the Network Utility, either using SNMP, or by a creating a .CFG file and transferring it manually.
3. Activate, debug, and tune the configuration at the Network Utility using the command-line interface.
4. Retrieve the configuration back into the Configuration Program either using SNMP or by reading in a .CFG file.
5. Regularly retrieve the configuration from the Network Utility, as you need to make dynamic configuration changes.
6. Make planned network changes from the Configuration Program and send the new configurations to the Network Utility.

See “Chapter 7. Handling Configuration Files” on page 79 for specific procedures to transfer configuration files.

Migrating a Configuration to a New MAS Release

You will occasionally need to migrate your Network Utility to a new release of MAS, either for maintenance purposes or to pick up new function[9]. Because a Network Utility configuration contains release-specific information, you must also upgrade your configuration to the level of the MAS release you are installing.

If you use ***only*** the command-line interface to configure your Network Utility, you simply load and boot the new MAS release using one of the procedures in “Chapter 10. Software Maintenance” on page 105. When the new MAS release boots, it automatically adjusts your configuration to the new release level. These adjustments are made in memory and do not affect the disk copy of the configuration. You can issue the **write** command at the `Config>` prompt to save the

[9] See “Chapter 10. Software Maintenance” on page 105 for background and procedures relating to code upgrade.

upgraded configuration to disk. You can leave a copy of the old release config in the disk bank with the old level of code, in case you need to boot from the older release.

If you use the Configuration Program ***at all***, even just occasionally, you ***must*** use the Configuration Program to upgrade your configuration. All new MAS releases are accompanied by a new release of the Configuration Program. Follow these steps to upgrade your configuration:

1. Using the old release version of the Configuration Program,
   a. If necessary, retrieve the configuration from your Network Utility into the Configuration Program. You need to do this only if you have made command-line changes to the configuration since the last time you sent it from the Configuration Program to the Network Utility.
   b. Save the configuration as a .CSF file (the internal Configuration Program format), using **Save** or **Save as** from the **Configure** drop-down menu.
2. Using the new release version of the Configuration Program,
   a. Open the configuration using **Open** from the **Configure** drop-down menu. The new version of the Configuration Program automatically upgrades the configuration to the new release as it reads it in.
   b. Save the new release version of the configuration.
   c. Transfer the new release version of the configuration to your Network Utility and activate it when you boot the new release of MAS.

Chapter 7. Handling Configuration Files

This chapter describes specific procedures for:

- Viewing and managing configuration files on the hard disk of a Network Utility
- Transferring configuration files from outside Network Utility onto its hard disk
- Transferring configuration files from the Network Utility hard disk

For background information about configuring Network Utility, see “Chapter 3. Performing the Initial Configuration” on page 25 and “Chapter 6. Configuration Concepts and Methods” on page 71.

For details about the individual commands introduced in this chapter, see the following chapters in the *MAS Software User’s Guide*:

- ″Using BOOT Config to Perform Change Management″
- ″Configuring Change Management″

Managing Configuration Files on Disk

All the commands to list and manage configuration files on the Network Utility hard disk are located in the boot Config subprocess. The following example shows how to reach this subprocess and list the available commands:

```
*t 6
              <Enter>
Config>boot
Boot configuration
Boot config>?
ADD description
COPY software
DESCRIBE software VPD
DISABLE boot choices
ENABLE boot choices
ERASE software
LIST software status
LOCK Config File
SET boot information
TFTP software
TIMEDLOAD software
UNLOCK Config File
UPDATE Firmware
EXIT
Boot config>
```

Listing Configurations

The **list** command is the starting point for viewing what configuration files are present in the four positions of each of the two code load banks. This same display is integrated into a number of the commands on the menu.

```
Boot config>li
+------ BankA -----------+--------- Description ----------+------ Date -------+
| IMAGE - ACTIVE         |                                | 03 Aug 1998 10:04 |
| CONFIG 1 - AVAIL       |                                | 04 Aug 1998 13:50 |
| CONFIG 2 - ACTIVE  *   | example config 1               | 04 Aug 1998 13:52 |
| CONFIG 3 - AVAIL       |                                | 04 Aug 1998 06:41 |
| CONFIG 4 - AVAIL       |                                | 04 Aug 1998 09:43 |
+------ BankB -----------+--------- Description ----------+------ Date -------+
| IMAGE - PENDING        |                                | 05 Aug 1998 03:41 |
| CONFIG 1 - PENDING *   |                                | 31 Jul 1998 12:59 |
 | CONFIG 2 - AVAIL       |                                | 31 Jul 1998 09:50 |
 | CONFIG 3 - AVAIL       |                                | 31 Jul 1998 09:52 |
 | CONFIG 4 - AVAIL       |                                | 31 Jul 1998 12:50 |
 +------------------------+--------------------------------+-------------------+
   * - Last Used Config      L - Config File is Locked

  Auto-boot mode is enabled.   Fast-boot mode is enabled.

 Time Activated Load Schedule Information...

 The load timer is not currently activated.
 Boot config>
```

Image (code load) and configuration states are defined as follows:

ACTIVE
The file was used for the current boot of the Network Utility

AVAIL This is a valid file that can be made ACTIVE.

CORRUPT
The file is not usable. Normally, this is because a file transfer to this position did not complete successfully.

LOCAL
The file will be used only on the next load or reset. After the file is used, it will be placed in the AVAIL state.

NONE No file is present in the position (the initial state).

PENDING
The file will be used on the next reload, reset, or power-on of the Network Utility.

To remind yourself of what is in a particular configuration file, use the **add** command to enter a brief description.

Making a Configuration Active

To make a particular configuration file active, you make it the PENDING configuration file in the bank with the ACTIVE or PENDING code load, and then reboot Network Utility. You do this either when the file already exists or when you create it, as follows:

- If the file is already on disk, use the **set** command to designate the bank and configuration file position to be used for the next boot.

  You can specify whether the new setting of source bank and configuration is just for the next boot (the state becomes LOCAL) or for all future boots (the state becomes PENDING).

  You normally use the **set** command after transferring a file to the disk using TFTP or Xmodem.
- If you create a new file using the talk 6 **write** command, it automatically becomes the PENDING configuration in the ACTIVE bank.

  When you do a **write**, the system writes the configuration in active memory to the next unlocked position in the ACTIVE bank, rotating in sequence. You do not pick the file position. If you want to prevent a particular file from being overwritten, use the **lock** command.

  Because the new file becomes PENDING, you can do a **write** followed by a **reload** without paying attention to the particular position used, and without having to issue the **set** command.

- If you create a file implicitly by typing **reload** and choosing to save configuration changes, the new file becomes the PENDING configuration before the reboot proceeds.

  The following sequence works the same way as issuing the **write** command:

  ```
  *rel y
  The configuration has been changed, save it? (Yes or [No] or Abort):yes
  ```

- If you create a file by using the Configuration Program **Communicate** option to directly transfer a configuration, the new file becomes the PENDING configuration.

  This also works the same way as issuing the **write** command. If you request a reboot from the Configuration Program, this configuration becomes active when the reboot occurs.

Delayed Activation

There are two ways to cause a timed, presumably unattended, activation of a configuration:

- If you are using the Configuration Program and transfer your configuration using the **Communicate** option, you can specify the date and time for the Network Utility to reboot and activate the configuration.
- No matter what method you use to create a configuration file on the Network Utility hard disk, you can used the **timedload** command in the boot Config subprocess to schedule a date and time for the Network Utility to reboot and activate a specified code load and configuration.

  If you choose the current code load and configuration, this function simply becomes a scheduled reboot operation.

File Utilities

The boot Config subprocess provides a number of utility commands for managing configuration files (and code loads) on disk:

add to enter a short description of a configuration

copy to copy a configuration between banks and/or file positions

erase to remove a configuration file and return the position status to NONE

lock to prevent the file from being overwritten by one of the file creation methods

unlock
to allow a file position to be used again for a new file

Firmware Change Management

Most of the configuration management functions in the boot Config subprocess are also available from Network Utility firmware menus. To access them, select the following sequence starting from the firmware main menu:

- Option 4, ″Utilities″
- Option 12, ″Change Management″

Loading New Configuration Files

Table 14 summarizes the ways you can transfer a configuration from outside the Network Utility to its hard disk. SNMP involves a direct transfer from the Configuration Program to the Network Utility, while TFTP and Xmodem require the configuration file to be on a workstation that acts as a file server to the Network Utility.

Which method you choose to transfer it into the Network Utility depends on how you can attach to the Network Utility, whether you are using the Configuration Program, what software you have on your workstation, and your own preferences. Network Utility configuration files are typically small enough that transfer times over low-speed modems are reasonable.

Table 14. Loading Configurations

Physical Attachment	Line Protocol	Transfer Protocol	Tool	Default IP Addresses
Service port + null modem Service port + ext modem PCMCIA modem	Async terminal	Xmodem	Firmware	Not applicable
	SLIP	TFTP	Op-code	Network Utility=10.1.1.2 Workstation=10.1.1.3
		SNMP	Cfg pgm	
PCMCIA EtherJet Ethernet LIC (10 Mbps) Token-Ring LIC	IP	TFTP	Op-code Firmware	Network Utility=10.1.0.2 Workstation=10.1.0.3
		SNMP	Cfg pgm	
Any IP network interface	IP	TFTP	Op-code	No defaults
		SNMP	Cfg pgm	

The following sections summarize each of the possible configuration transfer procedures, grouping them by the tool from which you start the transfer.

Using the Configuration Program

There are two ways to transfer a configuration from the Configuration Program to a Network Utility.

1. Create a router configuration file and then use the Network Utility operational code or firmware as the tool from which to do the transfer.
2. Use SNMP to transfer the configuration to Network Utility memory and hard disk.

Exporting a Router Configuration File

After you have started the Configuration Program and created a Network Utility configuration, move to the Navigation Window and:

1. Bring up the **Configure** drop-down menu and select **Create router configuration**.
2. Choose the directory path and filename on the workstation where you are running the Configuration Program, where you want the router configuration file (.cfg) to be stored.
3. Click on **OK.** The Configuration Program writes this file to disk.
4. Select **Save as** under **Configure** so that you also save the configuration in .csf format, the preferred format for archiving.

It is then your responsibility to load the file onto your Network Utility, using either the operational code or firmware to do the loading. You can follow any of the procedures described in “Using the Operational Code” on page 84 or “Using the Firmware” on page 85.

If your Configuration Program PC or workstation cannot be the TFTP or Xmodem server for the file transfer in these procedures, you must first move the .cfg file to a workstation that can be the server. You can use any file transfer method, such as FTP, to move the file between the workstations.

Directly Sending Using SNMP

In order to use SNMP transfer, you must configure the Network Utility with an IP address and enable SNMP with a read-write community name. Each of the sample configurations in “Part 2. Learning About Network Utility” on page 47 shows how to configure an IP address and SNMP for this communication, in both the Configuration Program and from talk 6.

If you want to use SNMP to download a Network Utility’s very first configuration, see “Procedure B: Configuration Program Initial Configuration” on page 29.

If this is not the first configuration, make sure there is at least one unlocked configuration file position (other than the active one) in the currently active code bank on the hard disk. (See “Listing Configurations” on page 79 for more information.)

After you have created a Network Utility configuration at the Configuration Program, use the following procedure to transfer that configuration to Network Utility using SNMP:

1. Bring up the **Configure** drop-down menu and select **Communications**.
2. From the pop-up, select **Single router** if you only want to send the current configuration to one Network Utility, or **Multiple routers** if you want to send any saved configuration to any number of target routers.
3. From the next single-router panel, or multiple-router list panel, select the **Send** option and enter the necessary addressing information for the routers.

   You can also enter a date and time for the router to be restarted with this configuration, if you wish. There are two ways to do this:

   a. Select **Send** and **Restart router**[10]

      The router stores the restart time in volatile memory, so if the Network Utility reboots before the scheduled time, the configuration is activated early.

      If you enter a date or time in the past, the router activates the new configuration immediately.

   b. Select **Timed config**

      The router stores the restart time in nonvolatile memory, so if the Network Utility reboots before the scheduled time it uses its current configuration. The newly downloaded configuration is not activated until the scheduled restart time arrives.

      If you enter a date or time in the past, the router stores the new configuration on disk but does not activate it. If a previous ″timed config″ restart operation is pending, it is cancelled.

[10] You can also do a **Send**, followed by a manual **Restart router** operation at a later time.

When you set the date and time by either of these methods it is not necessary to synchronize this date and time with the Network Utility, or even set a date and time at the Network Utility. The Configuration Program translates the date and time you set to a time interval and sends that value to the Network Utility.

4. Click on **OK** (or **Run** for the multiple-router list), and the Configuration Program starts sending configuration data items to the specified router or routers using SNMP. Sending starts immediately, regardless of whether you specified a later date and time for the target routers to reboot.
5. The Configuration Program provides status and result messages about the transfer. If you have problems and are sending to a single router, you may want to try the **Query router information** button instead of **Send**. This option retrieves a short amount of information from the router. You can use it to see whether you have an SNMP communication path to the router.

When a router begins to receive a configuration through SNMP, that configuration replaces any talk 6 changes made since the last reboot. When the transfer is complete, the Network Utility writes the received configuration to disk and activates it based on what you selected when you initiated the send operation.

Using the Operational Code

You can use the operational code to pull in a configuration file that was created in one of two ways:

- Exported from the Configuration Program using step 1 on page 82
- Previously transferred from this or another Network Utility

As Table 14 on page 82 shows, the configuration transfer procedures you can initiate from the op-code all use TFTP as the file transfer protocol.

Using TFTP

The op-code procedure for using TFTP to transfer a configuration file to the Network Utility hard disk is:

1. Place the configuration file on a workstation that has TFTP server software installed and IP network physical connectivity to the Network Utility.
2. Access the firmware main menu using the procedure described in “Boot Options: Fast Boot and Reaching Firmware” on page 45.
3. Configure the IP addresses you will be using.

   If you are using a standard network interface including an Ethernet or Token-Ring adapter, use the Configuration Program or talk 6 to configure an IP address for the interface in the normal way. (From talk 6, use **add address** in the IP subprocess.) Activate this configuration change before proceeding.

   If you are using the PCMCIA EtherJet card, use **system set ip** to set the following addresses:

   - IP address: the IP address for the EtherJet card
   - Netmask: the mask for the subnet attached to the EtherJet card
   - Gateway address: the IP address for the TFTP server workstation

   If you are using SLIP, you cannot change the IP addresses but must use those given in Table 14 on page 82.
4. Transfer the files

From the * prompt, follow this sequence:

```
*t 6
Config>boot
Boot configuration
Boot config>tftp get config
```

Respond to the prompts as follows:

- Server IP address: Put the address of the TFTP server workstation.
- Remote directory: Put the path name to the directory on the server workstation where the configuration file is. Use slashes in the direction expected by your server. Uppercase versus lowercase matters only if it matters to your server.
- Destination bank: Select bank A or bank B.
- Destination configuration: Select an unlocked position between 1 and 4.

Based on the server IP address and the configured Network Utility interface IP addresses, the Network Utility selects which of its interfaces to use to reach the server. The Network Utility gives success or failure status messages as appropriate.

5. Reboot or schedule a reboot to use the configuration.

   To activate the new configuration immediately, use the following procedure from the `Boot Config>` prompt:

   a. Use the **set** command to select the new configuration so that it will be used for the next reboot.
   b. Press **Ctrl-p** and then enter **reload** to reboot the Network Utility

   To activate the new configuration later, type **timedload activate** from the `Boot config>` prompt to select the bank and new configuration, and to specify the date and time for the Network Utility to reboot. You can answer ″no″ to the questions about loading, because you already did this step.

See the *MAS Software User’s Guide* chapter ″Configuring Change Management″ for more information on the commands in the above procedure.

Using the Firmware

You can use the firmware to pull in a configuration file that was created in one of two ways:

- Exported from the Configuration Program using step 1 on page 82
- Previously transferred from this or another Network Utility

As Table 14 on page 82 shows, the firmware supports both XMODEM and TFTP file transfer protocols.

Using Xmodem

The firmware procedure for using Xmodem to transfer a configuration file to the Network Utility hard disk is:

1. Place the configuration file on the workstation with the terminal emulation software supporting your current user console session.
2. Access the firmware main menu using the procedures described in “Boot Options: Fast Boot and Reaching Firmware” on page 45.

3. Make the following sequence of menu selections:
    a. System Management Services (main menu): Option 4, ″Utilities″
    b. System Management Utilities: Option 12, ″Change Management″
    c. Change Management Software Control: Option 12, ″Xmodem software″
    d. Select Type: ″Config″
    e. Select Bank: choose Bank A or Bank B
    f. Select Config: choose an unlocked position

    The firmware tells you when to start the file transfer.
4. Go to your terminal emulation package and start the transfer of the file from your workstation server, using whatever name you like. When the transfer starts, the status of the file position changes to CORRUPT, to indicate that it does not contain a complete configuration file. When the transfer completes, the status of the file position changes to AVAIL. You can verify that this has happened using option 7, ″List Software″ from the firmware Change Management menu.
5. Boot the Network Utility using the configuration you just loaded.

    Use Option 9 ″Set Boot Information″ to select the current op-code bank and the new configuration. Press **Esc** to reach the main menu, and then **F9** to boot the Network Utility with the new configuration.

Using TFTP

The firmware procedure for using TFTP to transfer a configuration file to a Network Utility hard disk is:

1. Place the configuration file on a workstation that has TFTP server software installed and IP network physical connectivity to the Network Utility.
2. Access the firmware main menu using the procedures described in “Boot Options: Fast Boot and Reaching Firmware” on page 45.
3. Configure the IP addresses you will be using:

    Follow the menu sequence:
    a. System Management Services (main menu): Option 4, ″Utilities″
    b. System Management Utilities: Option 11, ″Remote Initial Program Load Setup″
    c. Network Parameters: Option 1, ″IP Parameters″

    Set the following addresses:
    - Client IP address: an IP address for the Network Utility LAN card. This is a temporary address that need not be related to the Network Utility operational address for that interface.
    - Server IP address: the IP address of the workstation’s LAN adapter
    - Gateway IP address: the IP address of any intermediate router, or repeat the workstation’s IP address if there is none
    - Netmask: the mask for the subnet attached to the Network Utility LAN card
4. Initiate the transfer through these menu selections:
    a. System Management Services (main menu): Option 4, ″Utilities″
    b. System Management Utilities: Option 12, ″Change Management″
    c. Change Management Software Control: Option 10, ″TFTP software″
    d. Select Type: ″Config″
    e. Select Bank: choose Bank A or Bank B

5. Enter the path and filename of the configuration file on your workstation
6. If prompted, select the interface through which you want the firmware to do the file transfer.

   The firmware transfers the configuration file and gives status messages. On completion, you will be back at the Change Management menu.
7. Boot the Network Utility using the configuration you just loaded

   Use Option 9 ″Set Boot Information″ to select the current op-code bank and the new configuration. Press **Esc** to reach the main menu, and then **F9** to boot the Network Utility with the new configuration.

Transferring Configuration Files from Network Utility

You may want to transfer a configuration file ***from*** a Network Utility for any of the following reasons:

- You are using command-line configuration and you want to back up your configuration somewhere other than on the Network Utility hard disk.
- You are using command-line configuration and you want to export the configuration file to another Network Utility.
- You are using both the Configuration Program and command-line configuration and you want to update the Configuration Program file with recent talk 6 changes (for example, dynamic reconfiguration changes).

For the operational code procedures to transfer a configuration to a Network Utility, there is a reverse procedure for transferring a configuration from a Network Utility. The steps are virtually identical, so the following procedure lists only the essential differences.

1. Import a .CFG file into the Configuration Program.

   Transfer the .CFG file to the Configuration Program workstation. Do a **Read router configuration** instead of a **Create router configuration**.
2. Use SNMP to transfer a configuration into the Configuration Program. Do a **Retrieve configuration** instead of a **Send configuration**.
3. Use operational code TFTP to send a configuration from the Network Utility. Type **tftp put config** instead of **tftp get config**.

There are no firmware-based procedures to transfer a configuration from a Network Utility.

Chapter 8. Management Concepts and Methods

This publication uses the term *managing* to mean all the ways you can monitor and control what is going on with an active Network Utility. These ways include:

- Typing commands on a local or remote console to query status and change the state of interfaces and protocols
- Monitoring a running log of event messages, either through the same console or at a server for remote logging
- Using an SNMP MIB browser to query the status of interfaces and the functions of the box that have associated SNMP MIB support
- Using an SNMP-based management product and its applications to monitor and control interfaces and the functions of the box that have associated SNMP MIB support
- Using SNMP-based topology applications to monitor a protocol-specific (for example, APPN or DLSw) view of your network and its resources
- Using an SNMP-based management product to monitor SNMP traps sent by the box to report error conditions
- Using an SNA alert focal point product (such as NetView/390) to monitor SNA alerts sent by the box to report error conditions
- Using an SNA management product (such as NetView/390) to control SNA resources

This chapter gives an overview of these methods and introduces some of the other products you can use to manage your Network Utility.

Console Commands

To enter commands to query and change box status, you must first bring up a local or remote console attachment to an active Network Utility. For details on how to do this and reach the * prompt, see “Chapter 2. Bringing Up a User Console” on page 15.

Once you have an active console, use talk 5 to access the Console process. [11] From there, you navigate menus and issue commands to query the status of interfaces and protocols, and to make dynamic operator changes such as:

- Disabling and enabling interfaces
- Recycling connections
- Activating configuration changes

See “Operating (Using talk 5, the Console Process)” on page 61 for an overview of talk 5 commands and the types of status you can view and change from the operator console. Full details on the top-level talk 5 commands are provided in the *MAS Software User’s Guide* chapter ″The Operating/Monitoring Process (GWCON - Talk 5) and Commands″.

By using the talk 5 commands **net**, **protocol**, and **feature**, you can move down in the menu structure and use commands for monitoring and controlling interfaces and

[11] If you attach to a Network Utility that has never been configured before, you will be in Config-only mode and will not be able to go into the talk 5 Console process. Follow the instructions in “Chapter 3. Performing the Initial Configuration” on page 25 to configure your Network Utility for the first time and boot into normal operating mode.

particular protocols and features. Interface-level talk 5 commands are documented in the chapters of the *MAS Software User's Guide* devoted to the different interface types. Protocol and feature talk 5 commands are described in various chapters of the two-volume *MAS Protocol Configuration and Monitoring Reference*, and in *MAS Using and Configuring Features*.

Monitoring Event Messages

Why Monitor Events?

Talk 5 commands provide a snapshot of Network Utility status, but cannot produce a log or trace of events happening inside the box. For this you use ELS, the Event Logging System. By activating the right ELS messages and monitoring the event log, you can follow in real time such events as:

- Interfaces going through test phases, coming up, and going down
- Packets for a particular protocol being sent and received
- DLC links coming up and going down
- CPU utilization changing in response to network activity
- Higher level protocol connections (for example, DLSw partner and circuit connections) coming up and going down

By monitoring ELS messages, you can start to answer some basic questions, like:

- Is anything happening?
- Why is the link not coming up?
- Is my protocol seeing the traffic that a workstation is sending?

The Event Logging System is a powerful tool for debugging basic configuration problems.

Specifying Which Events to Log

To use ELS messages, you first tell the system which of its thousands of predefined events you want it to report to you. You can specify the set of active messages using the following criteria:

Subsystem name
Using the short predefined name of a software component such as IP, TKR, or DLS, you can refer to all possible messages from that component.

Event number
You can turn individual messages on or off, or specify a range of event numbers. It is sometimes useful to activate all the messages in a subsystem and then turn off a few particularly frequent ones within that subsystem, to avoid obscuring more critical messages.

Logging level
You can specify the severity level of the messages you want to see. For example, you may want to see only unusual error messages, or only trace messages, or include simple informational messages.

Group name
You can specify the name you chose previously when you defined a group of messages.

In addition, you can set up filters on a logical interface number, so that for any active set of messages, only those relating to a particular interface appear in the log.

Specifying Where to Log Events

When you activate messages, you choose one of the following destinations for the message:

1. The Monitor process

   You view messages sent to this process using the **talk 2** command from the * prompt. See “Event Logging (Using talk 2, the Monitor Process)” on page 66 for an introduction to using the Monitor process.

2. A remote logging server

   You can set up any PC or workstation that supports a standard *syslog* facility to receive a flow of event message packets and save them in a file. The Network Utility sends each message in a UDP/IP packet out through a standard network interface. Because log message flow can be heavy, a log server is normally LAN-attached to the Network Utility.

3. An SNMP trap, sent to an SNMP management station

   The Network Utility packages the event message in an IBM enterprise-specific SNMP trap, and sends it in a UDP/IP packet out through a standard network interface.

Activating Event Logging

From the command line, you can use either talk 6 or talk 5 to select the events you want to log and where to log them. From either process, enter the **event** subprocess to proceed. If you activate events under talk 6, the changes do not take effect until you write them to disk and reboot the Network Utility. The messages for those events will be continuously active from the first reboot on.

If you activate events from talk 5, the system immediately begins to deliver messages for those events to the destination you specify (talk 2, the log server, or SNMP management station). When you reboot the Network Utility, those messages cease to be active. Using talk 5 to activate events is a good way to debug an immediate problem you may be having. You turn on the events, quickly jump to talk 2 to see what is happening, and so on. When you reboot later, the events are deactivated without you having to enter any new commands.

Another useful debug technique is to use the talk 5 event subprocess to view statistics of how many times any event has been encountered. These statistics are available even for events that have not been activated.

There is no talk 5 command to activate the current talk 6 ELS configuration. If you want immediate activation, you must repeat the same commands in talk 5 that you entered in talk 6.

From the Configuration Program, you can only set up the Network Utility to do remote logging to a host. You cannot configure which ELS events are active or direct ELS events to a particular destination. The Configuration Program does preserve this configuration information, however, if you retrieve a configuration from a Network Utility, modify other parts of the configuration using the Configuration Program, and write the configuration back out.

From an SNMP management station, you can use SETs to control most ELS configuration functions using an enterprise-specific ELS MIB.

For an introduction to some of the key commands to activate and control ELS events, see “Monitoring Events” on page 101. For a detailed explanation of ELS concepts and the associated talk 6 and talk 5 commands, see the *MAS Software User’s Guide* chapter ″Configuring and Monitoring the Event Logging System (ELS).″ For a description of every individual ELS message, see the *Event Logging System Messages Guide* either on CD-ROM or on the Web.

Simple Network Management Protocol (SNMP) Support

SNMP is an industry-standard protocol that management stations use to query and set configuration, control, and status information in a managed node. In the Network Utility context, the management station would normally be a PC or workstation with an SNMP management software product installed on it. The managed node would be the Network Utility.

SNMP requests and replies flow inside UDP packets through an IP network between the management station and the managed node. In general, the management station initiates communication by sending requests for information and requests to set data items to new values. The managed node simply carries out these requests and replies to them. A managed node can, however, send an unsolicited message called a *trap* to report an event. A Network Utility might send a trap to report such events as a box reboot or an interface going down.

A *Management Information Base (MIB)* is a virtual information store defining the data items in the managed node that can be accessed from the management station. MIBs are defined in strictly formatted description files which can be read both by people and by management station software.

A managed node product *supports* a MIB when its software can field SNMP requests for the data items documented in the MIB, and retrieve or set its corresponding internal data items. The MIB description file defines for each data item whether the management station can only read it, or can modify its value. Sometimes, a product chooses only to allow read-access to a data item that the MIB documents as write-able. You should consult product documentation to understand the level of access a particular product has implemented.

Most industry-standard protocols and interface types have an associated IETF standard MIB with an RFC number. Standard MIBs define data items that are common to most implementations of the associated protocol or interface type. Vendors cannot always wait for a MIB to reach standard RFC status within the IETF, and sometimes ship support for a pre-standard *Internet Draft* version of the MIB.

In addition to the standard MIBs, many product vendors develop their own MIBs to define data items that are unique to their products. For example, Network Utility supports MIBs that give access to memory and CPU utilization information, for which there is no standard MIB. In SNMP parlance, these vendor MIBs are called *enterprise-specific* MIBs.

MIB Support

IBM Network Utility supports a comprehensive set of standard and enterprise-specific MIBs for monitoring and managing resources. The current list numbers somewhere between 40 and 50 MIBs.

You can find a ″README″ file documenting Network Utility MIB support by accessing the appropriate software release directory on the World Wide Web at:

```
ftp://ftp.networking.raleigh.ibm.com/pub/netmgmt/netu
```

In the same directory, you can find the MIB description files themselves, ready to be retrieved using FTP and loaded into a management station. Whenever possible the files are compiled into SNMP Version 1 format, to make them compatible with the widest possible variety of management station software.

For the standard and Internet Draft MIBs, the compilation process strips out introductory explanatory text and page formatting that helps make a MIB more readable. To get the full pre-compiled version of an RFC or Internet Draft MIB, retrieve it from an IETF FTP site as you would any RFC or Internet Draft. You can start at the following URL and follows links to the RFC or Internet Draft repository:

```
http://www.ietf.org
```

The following MIBs are new for this release:

- **Layer 2 Tunneling MIB**

  The Layer 2 Tunneling MIB is an IBM Enterprise MIB which allows you to view information about active layer 2 tunnels and the statistics associated with them. There are traps for tunnel starts and stops and authentication failures. You can also issue sets to a test MIB which will test whether the tunnel can be established, based on the configuration information for that tunnel. There is a test MIB which allows you to determine the response time for a tunnel.
- **Policy MIB**

  The Policy MIB is an IBM Enterprise MIB which consists mostly of the policy information loaded into the policy database of the router. You can determine where the policies were loaded from, either local configuration, from an LDAP server, or both. The MIB keeps track of the number of times a policy has been hit and any active negotiations that are taking place for IPSec and IKE. The negotiation information can be used to index back into the IPSec/IKE MIB for specific information about the negotiations. You may also look at both the administrative and operational LDAP configuration parameters. The MIB allows you to perform sets, thus changing the configuration, for some of the LDAP parameters. It has an object which causes the policy database to refresh when set. There is also a policy test MIB which allows you to set the selectors for a policy query (Source and Destination IP Address, protocol and port number and DS byte) and determine which policies and actions would result from a policy query with those selectors.
- **IPSec/IKE MIB**

  The IPSec/IKE MIB is an IBM Enterprise MIB which allows you to view the active negotiation information for phase I and phase II of IKE. It also provides tables for the IPSec statistics for encryption and decryption as well as errors. There are traps for tunnel starts and stops and authentication and decryption failures. The MIB also displays information about which subnets and applications are being protected and the local and remote identification information for the security gateways.

|

Getting Started

At the Network Utility

Before an SNMP management station can communicate with your Network Utility, you must first configure SNMP in the Network Utility with the appropriate access enabled. You can use either the Configuration Program, talk 6, or talk 5 to enable SNMP and set up a *community name* that grants access to one or more management stations. From talk 6 or talk 5, use **protocol snmp** to access the Config and Console subprocesses for working with SNMP. As shown in step 3 on page 27, you can also use Quick Config to enable SNMP and set up a read or a read-write community name.

See the *MAS Protocol Configuration and Monitoring Reference Volume 1* chapters ″Using SNMP″ and ″Configuring and Monitoring SNMP″ for more background information, and for a description of the SNMP talk 6 and talk 5 commands.

At the Management Station

Before a management station can provide any significant support of a managed node, it must know what MIBs that managed node supports. If you are using any of the IBM products described in “IBM Nways Manager Products” on page 96, you do not have to do anything to set this up. Each of them has the MIBs that Network Utility supports already compiled in.

If you are using some other management product, you may have to set up this knowledge. Management stations typically provide a facility for loading compiled MIB modules into the station. When you are preparing a management station to manage Network Utility, set it to read in all the MIBs from the appropriate directory under the URL given in “MIB Support” on page 93.

If you intend to send traps from Network Utility to the management station, you may also need to set up the management station to issue messages or take specific actions on receipt of a trap.

SNA Alert Support

IBM’s Systems Network Architecture (SNA) defines a rich set of protocol flows for the purpose of managing network products. A key part of that architecture is the ability for the managed node to send an unsolicited error or event report, called an *alert*, to an SNA management station. An alert contains a sequence of submessages that enable the management product to report to an operator such things as:

- The identity of the node that built the alert
- The error or event that prompted the alert
- Several possible causes for the problem
- Possible corrective actions

The SNA management product most commonly used to receive alerts is NetView/390. In SNA architecture, such a product is called an alert *focal point*. A product in the network that can receive and forward alerts on behalf of other products is called an *entry point*.

When you are using Network Utility as an APPN network node, it has the capability to establish LU6.2 sessions with alert focal points and send native SNA alerts to report error conditions in the box and in the network. The following are some of roughly 30 predefined events that trigger an alert from Network Utility's APPN function:

- Session setup failure
- Invalid XID received, XID protocol error
- HPR or DLUR configuration or protocol error
- CP-CP session failure
- Out of resources
- Subcomponent protocol error

If one of these events occurs and Network Utility has no current focal point session on which to send the alert, it queues the alert for later transmission. You can configure the depth of this ″held alert″ queue. You cannot configure which of these events will trigger an alert.

The LU6.2 session on which alerts flow can be established either by the focal point or by the Network Utility. You do not have to configure any special parameters at your APPN Network Utility to enable it to accept a session from an alert focal point and send alerts. If you want the Network Utility to actively set up sessions and forward alerts, you configure the name of one or more *implicit* focal points as part of your APPN configuration. If the primary focal point cannot be reached, Network Utility attempts to reach the other configured names.

In addition to sending alerts for events it detected, Network Utility can serve as an SNA entry point and forward alerts on behalf of other SNA nodes with which it has sessions. No configuration is required to enable this function.

Getting Started

You can use the Configuration Program or talk 6 to configure focal point names if you want the Network Utility to activate the focal point sessions. From talk 6, use **protocol appn** to access the Config subprocess for working with APPN, and then use the **add focal-point** command.

For more background, refer to the *MAS Protocol Configuration and Monitoring Reference Volume 2* APPN sections ″Entry Point Capabilities for APPN-related Alerts,″ ″Configurable Held Alert Queue,″ and ″Implicit Focal Point.″ The command names are given in the ″Router Configuration Process″ section in the same chapter.

Network Management Products

Both SNMP and SNA management flows require a product separate from Network Utility, to manage a display of the network and the Network Utility, to query the Network Utility for status, or to receive unsolicited event reports from the Network Utility. This section lists some of the products you might use to perform these tasks.

SNMP MIB Browsers

A *MIB browser* is a small PC or workstation application that can load MIB definitions, query or set data items in a managed node, and decode returned values and results into a easily readable form. In SNMP terms it is a management station, but a MIB browser lacks the power and sophistication of a full-fledged SNMP management platform such as those described in the following section. MIB browsers are frequently packaged as part of such platforms, but can also be stand-alone products.

IBM Nways Manager Products

The following IBM SNMP network management products are specifically intended to manage Network Utility and a wide variety of other IBM and non-IBM networking products. Each of them provides a graphical topology view of your network resources, with a color-coded status of resources and overall status of each network. Each supports automatic discovery of network resources, and automatic updates to a network map in response to network changes.

IBM Nways Manager for AIX

Designed for managing medium to large network environments, this product runs on a workstation running AIX, IBM’s version of UNIX. Nways Manager for AIX runs on top of Tivoli TME 10 NetView, which was formerly known as ″NetView for AIX″ and ″NetView/6000.″ Tivoli TME 10 NetView provides general network management platform capabilities such as the management of LAN topologies, fault and event recording, and error logging. When combined with IBM’s SNA Server for AIX, Tivoli TME 10 NetView can also map SNMP traps to SNA alerts. For Network Utility, this permits an SNA alert to flow for virtually any defined ELS event.

Nways Manager for AIX provides the following capabilities on top of the base Tivoli TME 10 NetView functions:

- A Network Utility-specific management application

  When you select a Network Utility from the network topology view, you see a graphic of the front panel of the Network Utility, with color-coded interface status. A navigation window on the side allows you to access all SNMP MIB information provided by Network Utility, either in graphical or tabular form. This application enables you to:

  – View or change adapter and interface status
  – Display statistics at a component or interface level
  – Receive real-time, color-coded status at a glance
  – Define and monitor performance thresholds
  – Define and monitor real-time and historical statistics
  – Monitor real-time events

  From the Network Utility application, you can launch:

  – The 2216/Network Utility graphical Configuration Program, to configure the box
  – A Telnet session to the Network Utility, so you can use the command-line interface to configure, monitor, and control the Network Utility

  Because the Network Utility management application is Java-based, you need not be at the workstation running Nways Manager to use it. You can bring up the application from a PC or workstation running a JDK-compliant Web browser, connected over your intranet or the Internet to the main Nways Manager workstation. For details on which Web browsers and versions of JDK are required, see the Nways Manager product prerequisites at:

  ```
  http://www.networking.ibm.com/netmgt
  ```

Java management support includes viewing real-time status of the Network Utility and the ability to do performance management. For security reasons, you cannot launch the Configuration Program from a Java web browser.

- Distributed Intelligent Agents

  To provide support for larger networks, you can use boxes other than the Nways Manager workstation to poll the managed nodes in your network. Offloading polling from the manager workstation frees its processor to do other tasks, and it frees network bandwidth because you place the polling closer to the devices being polled. These ″agents″ of the manager can be configured to notify Nways Manager when thresholds are exceeded.

  The intelligent agent software is Java-based and is downloaded from Nways Manager. The agents can be placed in any Java-enabled (Java virtual machine) workstations in the network. Nways Manager can also use the distributed polling capabilities provided by the TME 10 Mid-Level Manager.

- APPN topology support

  Nways Manager for AIX provides an APPN-level view of the topology of your network. You can discover participating APPN resources, view them, and view their status as color-coded icons. APPN protocol performance and error events (data and graph) are also provided. This application does not present Branch Extender or Extended Border Node topologies.

- DLSw topology support

  Nways Manager for AIX can also show you a DLSw topology view of your network, including DLSw connectivity, resources, and color-coded status. The topology is refreshed as new nodes are discovered. This application does not present the topology of DLSw IP multicast groups.

- VLAN, ATM, and RMON support

  Nways Manager for AIX has comprehensive support for products implementing virtual LANs, for ATM networks, and for collecting, correlating, and displaying data from RMON and ECAM probes.

- VPN Management Application

  The Nways Virtual Private Network (VPN) Management Application provides a rich set of Monitoring, Event Reporting, Troubleshooting, Operational Control and Application Launching functions. The initial version is specifically targeted to provide monitoring and operational control of the VPN capabilities for the IBM 22xx Router and uses private MIB Objects in providing it’s functions as standard MIB Objects do not currently exist.

  The Nways VPN Management Application concentrates on three VPN Components:

  – VPN Tunnels

  – VPN Clients

  – Policies

  The Monitoring function allows you to view active and previous VPN Tunnels, view active and previous VPN Clients and view defined and active VPN Policies. The Event Reporting function informs you when a VPN Tunnel starts and when VPN Device experiences a Security Attack. The Operational Control function allows you to disable/inactivate a VPN Tunnel, disable/inactivate a VPN Client and refresh VPN Policies. The Troubleshooting function allows you to Proxy-Ping a VPN device and view VPN Event Failure Logs. The Application Launching function will provide the ability to launch a variety of related network management applications such as the device’s PSM/JMA.

The first version of Nways Manager for AIX with specific support for Network Utility is Version 1.2.2.

For more information about Nways Manager for AIX including specifications and system requirements, go to:

```
http://www.networking.ibm.com/cma/cmaprod.html
```

The pages at this site describe the separately priced components of Nways Manager for AIX, and which components perform which of the above functions.

IBM Nways Workgroup Manager for Windows NT

Designed for managing small-to-medium network environments, Workgroup Manager is a 32-bit native Windows NT application that operates on NT Version 4.0. Unlike Nways Manager for AIX, Workgroup manager is self-contained and does not use an underlying network management platform. It must therefore provide a number of platform functions itself.

Key features of Nways Workgroup Manager for Windows NT include:

- Automatic discovery of your IP network
- Real-time, graphical views of network topology
- Ability to browse, update, and compile MIBs
- Color-coded and aggregated network and device real-time status
- Trouble ticketing
- Trap management, including specifying trap severities
- Trap compiler
- Polling configuration and notification
- Performance threshold configuration and notification
- Inventory management
- Collection and presentation of real-time and historical statistics
- VPN Management Application

Nways Workgroup Manager for Windows NT supports exactly the same Network Utility-specific Java management application as Nways Manager for AIX. You can run the Network Utility management application from a Java-capable web browser. Nways Workgroup Manager for Windows NT also supports Distributed Intelligent Agents.

Nways Workgroup Manager for Windows NT does not support the APPN and DLSw topology applications that Nways Manager for AIX does. The Nways Workgroup Manager for Windows NT topology display is based on IP connectivity between the managed nodes.

The first version of Nways Workgroup Manager for Windows NT with specific support for Network Utility is Version 1.1.2.

IBM Nways Manager for HP-UX

Designed for managing medium-to-large network environments, this product runs on a workstation running HP-UX, Hewlett Packard’s version of UNIX. Nways Manager for HP-UX runs on top of HP’s *Network Node Manager* management platform software, previously known as ″HP OpenView.″

In this environment, network node manager provides the base management platform functions, including topology display, trap management, and so on. Unlike Nways Manager for AIX, it enables you to associate IBM network devices with the appropriate Nways Manager for HP-UX management application.

From Nways Manager for HP-UX, you can launch the same Network Utility-specific Java management applications as you can from Nways Manager for AIX. Nways Manager for HP-UX also supports Distributed Intelligent Agents.

Nways Manager for HP-UX does not support the APPN and DLSw topology applications that Nways Manager for AIX does.

The first version of Nways Manager for HP-UX with specific support for Network Utility is Version 1.2.

NetView/390

NetView/390 is a host-based management product for managing medium-to-large SNA networks. There are several ways you can use NetView/390 to manage a Network Utility and the SNA products it can connect to the host:

- Controlling SNA resources (activating and deactivating links, PUs, and LUs)
  - When Network Utility is running DLSw, NetView/390 can control the links that DLSw is representing, and the PUs and LUs in remote SNA end stations.
  - When Network Utility is running TN3270 server support, NetView/390 can control the local PUs and LUs represented in the Network Utility.
  - When Network Utility is running DLUR for downstream nodes, NetView/390 can control the PUs and LUs the Network Utility is serving, and the links between Network Utility and those nodes.
  - When Network Utility is bridging SNA end-station traffic, NetView/390 can control the end-station PUs and LUs.
  - When Network Utility is running APPN, DLSw, or bridging SNA traffic, NetView/390 can control adjacent links between the host and Network Utility.
  - When Network Utility is running LSA direct gateway function, NetView/390 can control the LAN links that appear to be local to VTAM, as well as the PUs and LUs of the attached SNA end stations.
- Monitoring network errors and topology
  - NetView/390 can be the alert focal point when Network Utility is serving as an APPN node, both for the alerts that Network Utility generates and those it forwards from other nodes.
  - When Network Utility is running DLSw, DLUR, or bridging SNA traffic, NetView/390 can receive alerts, response time information, or any other SSCP-PU flow from a downstream PU.
  - NetView/390 can be the alert focal point for Network Utility traps that have been converted to alerts by Tivoli TME 10 NetView and SNA Server for AIX.
  - Through related products *SNA Topology Manager*, *APPN Accounting Manager*, and *APPN Topology Integrator*, NetView/390 can acquire and monitor the topology of an APPN network including Network Utility and other SNMP-capable APPN products.

Chapter 9. General Management Tasks

This chapter gives procedures and commands for important Network Utility operations. It serves as a supplement to some of the concept presentations in previous chapters.

Monitoring Events

This section supplements the background information on event logging and viewing provided in “Event Logging (Using talk 2, the Monitor Process)” on page 66 and “Monitoring Event Messages” on page 90. It introduces the commands that control what events are logged, and where they are logged.

Accessing the Event Logging System

You must use the command-line interface to activate event logging. From the Configuration Program, you can configure only general remote logging parameters.

From either the main talk 5 or talk 6 prompt, type **event** to enter the ELS Console or Config subprocess, respectively. You see essentially the same commands whether you are working under talk 5 or talk 6. Talk 5 ELS commands take effect immediately and are quite useful for turning on messages to debug a particular flow in a running system. From talk 6, you configure the events you want to be logged all the time, so you do not have to activate them each time you reboot the Network Utility.

Commands to Control Event Logging

There are six basic commands for activating and deactivating event logging, two for each of the three possible destinations of log messages:

- **disp** and **nodisp** control which events are locally logged to talk 2
- **trap** and **notrap** control which events generate SNMP traps
- **remote** and **noremote** control which events are remotely logged to a syslogd-capable host

All of these commands use the same method for specifying which events are to be activated or deactivated. Following the name of the command on the command line, you normally type one of the following (there are other options):

- **event** *subsystem.event#*, to specify a single predefined event

  subsystem is the name of a functional component as known to ELS, such as ″dls″ for DLSw or ″esc″ for ESCON. You can type **li sub** to get a list of ELS subsystem names.

  event# is the number of a predefined event, typed with leading zeros. You can type **li sub** *subsystem* to get a quick list of the events in a particular subsystem.

- **sub** *subsystem logging_level*, to specify some set of the predefined events in an ELS subsystem

  subsystem is the ELS subsystem name described above. The value ″all″ selects all subsystems.

  logging_level is optional and defaults to ″standard″, which includes all error and unusual informational messages. The value ″all″ selects all messages in the subsystem.

The following list gives a few examples of these commands:

disp sub all
Enables logging to talk 2 of all error and unusual informational messages in all ELS subsystems. This is a good general setting to configure in talk 6.

rem sub dls
Enables remote logging of all error and unusual informational messages in the DLS subsystem. Separately, you need to configure the destination host for remote logging.

disp sub sdlc all
Enables logging to talk 2 of all messages in the SDLC subsystem. You might enable all messages when trying to trace an error situation.

nodisp ev sdlc.008
Disables logging to talk 2 of a particularly chatty SDLC message, which may be interfering with seeing more important messages in the error log.

trap ev dls.475
Enables sending an SNMP trap when a particular DLSw QLLC error event occurs.

For detailed information about these commands, how to configure remote logging, what the logging levels are, and more, refer to ″Using the Event Logging System (ELS)″ in the *MAS Software User's Guide*.

Monitoring Memory Utilization

This section describes how Network Utility memory is used, and how you can monitor its status.

Network Utility Memory Usage

A Network Utility ships with either 256 MB or 512 MB of main memory. When you boot the system, it loads operational code from disk into this memory, taking a certain amount of memory space for each load module. Once the operational code is loaded, the system splits up the remaining memory between APPN/TN3270 (if configured) and the routing function. The routing function includes IP, DLSw, TCP, channel gateway; in short, every function except APPN and TN3270 server.

When you configure APPN either from the Configuration Program or the command line, you can specify the amount of memory to be reserved for APPN. In Network Utility, this value is preset to the memory required for a maximum TN3270E server configuration[12].This value should be reasonable for non-TN3270 APPN applications as well, so you should not need to change it. If your configuration does not enable APPN, Network Utility ignores the configured value and does not reserve memory for APPN. If your configuration enables APPN, Network Utility allocates the specified amount of memory to APPN and then allocates all remaining memory to the routing function.

You can monitor memory utilization in a running Network Utility either from a command-line console or from an SNMP management station. Either way, you look

[12] With the introduction of support for 512 MB, the Configuration Program defaults to assume a target Network Utility will have 512 MB of memory. If you load this configuration onto a Network Utility with 256 MB, it will automatically adjust the memory setting downward to the default value for boxes with 256 MB of memory. You do not need to change the Configuration Program default.

separately at the status of APPN memory and the status of routing function memory. Once the system is loaded, these memory partitions are fixed and are managed independently.

Monitoring Memory from the Command Line

To monitor routing function memory from the command line:

1. From the * prompt, type **talk 5** and press **Enter** to reach the + prompt.
2. Type **mem** to see summary and detailed statistics on current memory utilization. The output uses the term *heap* to refer to the memory being used by the routing function.

To monitor APPN/TN3270 memory from the command line:

1. From the * prompt, type **talk 5** and press **Enter** to reach the + prompt.
2. From the + prompt, type **p appn** and press **Enter** to reach the APPN Console subprocess.
3. Type **mem** and press **Enter** to see summary and detailed statistics on APPN memory utilization. The output breaks down APPN memory into various pieces and shows the state of each piece.

Monitoring Memory using SNMP

Network Utility supports IBM enterprise-specific MIBs that provide access to memory utilization information both for the routing function and for APPN/TN3270.

The Nways Manager products discussed in "IBM Nways Manager Products" on page 96 provide full statistical support for both the APPN and routing function memory partitions. For either partition, you can view real-time and historical utilization information. You can set up alarm thresholds for either utilization percentage, so you can be notified when memory utilization reaches a certain level.

You can also configure Network Utility from the command line to send an SNMP trap when available routing function memory falls below a given threshold. From the talk 6 prompt `Config>`, type the command **patch mosheap-lowmark** and give the percentage value if you want to change it from the default value of 10%.

Monitoring CPU Utilization

This section describes how to control CPU monitoring, and get reports from talk 5 or direct periodic messages to the talk 2 log.

Accessing Performance Monitoring

From either the main talk 5 or talk 6 prompt, type **perf** to enter the Performance Monitoring Console or Config subprocess, respectively. From talk 6 and from the Configuration Program, you can enable or disable CPU utilization monitoring and set its operating parameters as part of your Network Utility configuration. From talk 5, you can make the same changes take effect immediately, and you can get reports on the CPU utilization in a running Network Utility.

Monitoring CPU Utilization from the Command Line

Once you are at the `PERF Console>` prompt, the following commands are available to you:

report Give a summary of current CPU utilization, high water marks, and historical distribution of values.

enable cpu, disable cpu
Control the overall gathering of CPU utilization information. By default, Network Utility runs with CPU utilization enabled, with negligible impact on system performance. If you are running TN3270 server functions with Network Dispatcher, it is particularly important to leave CPU utilization enabled.

enable t2, disable t2
Control the generation of a periodic ELS message in talk 2 showing current CPU utilization. If you enable this message, you can avoid having to repeatedly type the **report** command to monitor how CPU utilization is changing.

set, list, clear
Set the time window for statistics gathering. View the current values of all settings. Reset statistics.

All the same commands or parameters are available from talk 6 and the Configuration Program, except for **clear** and **report**.

For more information on these commands and examples of their output, see ″Configuring and Monitoring Performance″ in the *MAS Software User's Guide*.

Monitoring CPU Utilization using SNMP

Network Utility supports an IBM enterprise-specific MIB that provides access to current and historical CPU utilization information.

The Nways Manager products discussed in “IBM Nways Manager Products” on page 96 provide full statistical support for Network Utility CPU utilization. You can view both real-time and historical utilization information. You can set up alarm thresholds for the utilization percentage, so you can be notified when it reaches a certain level.

Chapter 10. Software Maintenance

This chapter describes what you need to know to receive and install fixes for Network Utility software problems, and to upgrade to new software releases containing new function.

This information includes:

- How the software is named and packaged
- How to download new software versions from the World Wide Web
- How to load software onto Network Utility
- How to call for product service and support

Software Versions and Packaging

Version Naming

The software that operates Network Utility is called *Multiprotocol Access Services*, or MAS. MAS also operates the IBM 2216-400, but there are different, separate packages of MAS for each product. The MAS packages for Network Utility are characterized by:

- Preset configuration defaults, to tune Network Utility for its intended applications.
- Specialized function packaging oriented toward the key uses of Network Utility. For example, some of the general multiprotocol routing functions of the 2216-400 such as IPX, Appletalk, Banyan Vines, and DECNet, are not available in the Network Utility packages.

Specific levels of MAS are identified by the following numbers:

Version
A new function release occasionally requires a new version number. Sometimes this is related to a price increase, but it could also be related to a shift in how IBM is distributing the software. A new version number does not mean that the release has any more new function than a release that only has a new release number.

Release
This number changes with every new function release.

Modifier
This number signals a new function release that is a small change to a larger base new function release. It follows the decimal point in the format ″MAS Vv.r Mod m PTF p″.

PTF This number represents a maintenance level, described below.

The initial code base for Network Utility is: MAS V3R1.0 PTF 1. Because IBM is using the same release numbering as for the 2216-400 packages of MAS, you can easily correlate the function and maintenance level of software on the two products.

To see the software level of the code that is actively running in your Network Utility, move to the base talk 5 menu and type **c** (for ″configuration″). The software version part of the output of this command uses the format ″MAS Vv.r Mod m PTF p″.

To see the software level of the code loads on the Network Utility hard disk, move to the base talk 6 menu, type **boot** to enter the boot Config subprocess and then type **describe**.

Maintenance Levels

When you access the World Wide Web pages that contain recent versions of Network Utility software, you see some of the following terms for different maintenance levels of the Network Utility packages:

GA Level
The software level first made ″generally available″ to IBM customers. This is the level initially shipped on the hard disk of new Network Utility boxes. GA level software undergoes an extensive product-level and system-level test before it is released. General availability normally corresponds to a new Version or Release of the software (the initial release of Network Utility in a PTF is an exception to this rule).

PTF A major maintenance release (″program temporary fix″) that accumulates a large number of fixes and undergoes a regression test of most major software functions. After a Release has been deployed for some time, IBM will typically start to ship a stable PTF on the hard disk of new products.

EPTF A small maintenance release (″emergency PTF″) that comes out on a more frequent basis, involves fewer fixes, and undergoes a regression test of the specific areas affected by the fixes.

PTFs and EPTFs are cumulative, in that each supersedes all previous PTFs and EPTFs. You only need to install the latest PTF or EPTF to obtain all previous fixes.

Feature Packaging

There are two feature packages of Network Utility software, corresponding to the two different models of Network Utility:

Model	Description
TX1	Base code, including DLSw, APPN, IP and VPN function
TN1	Base code plus TN3720E server function

Based on the model you purchased, your Network Utility comes preloaded with the proper software package in both banks of the hard disk. When you load a new maintenance level of software, you load the same package that is already on the Network Utility.

Note that there is only one version of the Configuration Program, and it supports the software functions in all the software packages. If you configure functions that are not supported in the particular software package you have on the router, the router software ignores that part of the configuration.

From the command line, you cannot configure or monitor software functions that are not present in the software load you are running.

Getting Web Access to the Software

To update your Network Utility software, you must first download the appropriate maintenance level from the World Wide Web. To find the new software, start with the main Network Utility product page at:

```
http://www.networking.ibm.com/networkutility
```

Click on **Support** and then **Downloads** to reach the following information and links:

- General information about accessing the software
- Detailed procedures for downloading and installing the software
- Links to the latest maintenance levels of the Configuration Program, with associated README files
- Links to the latest maintenance levels of MAS, with associated PTF or EPTF content files

When you follow the links to a particular maintenance level of the Configuration Program, you can access packed binary versions of the 2216/Network Utility Configuration Program for each of its supported operating systems. Anyone can download these files. The associated README file gives instructions for unpacking and installing the new version of the Configuration Program.

When you follow the links to a particular maintenance level of MAS, you can access compressed packed binary versions of each of the Network Utility software features listed above.

You need an IBM Networking customer id and password to be able to download these files. You create the id and password yourself by registering on the Web, and you can immediately use it to download files. This id and password spans multiple IBM Networking products, and allows you to subscribe for e-mail notification of product updates. If you do not have one, the Web pages will take you to the registration page the first time you download a Network Utility code package.

Downloading and Unpacking Files

The Web pages for downloading a particular MAS maintenance release contain files for each of the supported software features. Each file contains a complete set of software for Network Utility. When you install a maintenance level of Network Utility software, you completely replace all the existing software with the new level.

To download the software in a particular file and transfer it to the router, you:

1. Use your Web browser to download the complete file in binary to your workstation.
2. Transfer the file to the workstation from which you will load it into the router. This is called the *server workstation* because it acts as a file server to the router. You can use FTP or any other file transfer method for this step.
3. At the server workstation, unpack the single downloaded file into a number of router software files. These files are called *load modules* and have the file extension ″ld″ (or ″LD″ if your file system does not support mixed case).
4. Using TFTP or Xmodem, you transfer the load modules into the router.

Depending on the MAS release, the Web page may contain two files for each software feature, each constructed by a different packing utility. Choose the version that your server workstation software can unpack. Normally you would choose as follows:

Server Operating System	File Format	Unpack Command
DOS, Windows, or OS/2	.zip	pkunzip
UNIX or AIX	.tar	tar -xvf

When you unpack the router software, make sure that all ″.ld″ files are in the same directory, and have file system permissions to give appropriate read access. Do not change the names of any of the .ld files. Do not mix files between different Network Utility feature packages, or between different maintenance levels of the same package. Keep each package distinct and separate with a different path name on your server workstation.

Loading New Operational Code

Operational code (op-code, for short) is the software that runs the normal packet forwarding and system services functions of Network Utility. Op-code includes the base operating system, protocols, features, diagnostics, and the command-line interface code. The vast majority of software changes in PTFs and EPTFs are changes to the operational code.

To load and activate new operational code, you must:

1. Transfer the unpacked load modules from your server workstation into one of the two op-code banks on the Network Utility hard disk.
2. Set the router to boot from the bank with the new op-code.
3. Reboot the router, or schedule it to reboot at a later date and time.

Table 15 summarizes the different ways you can transfer operational code from a server workstation to a Network Utility hard disk. Which method you choose depends on how you can attach the workstation to the router, what software you have on your workstation, and your own preferences. Here are some important points to consider:

- The size of all the .ld files combined is over 10 MB. If you can possibly use a LAN or network interface instead of the service port or modem, you should do so to avoid hours of file transfer time.
- The TFTP-based methods from the op-code and firmware automatically transfer all .ld files in a single operation. With Xmodem, you must manually specify the name of each of the roughly 20 .ld files that make up a software load.

Table 15. Loading Operational Code

Physical Attachment	Line Protocol	Transfer Protocol	Tool	Default IP Addresses
Service port + null modem Service port + ext modem PCMCIA modem	Async termimal	Xmodem	Firmware	Not applicable
	SLIP	TFTP	Op-code	Network Utility=10.1.1.2 Workstation=10.1.1.3
PCMCIA EtherJet Ethernet LIC (10 Mbps) Token-Ring LIC	IP	TFTP	Op-code Firmware	Network Utility=10.1.0.2 Workstation=10.1.0.3
Any IP network interface	IP	TFTP	Op-code	No defaults

Using the Operational Code

As Table 15 on page 108 shows, the transfer procedures you can initiate from the op-code all use TFTP as the file transfer protocol.

Using TFTP

The op-code procedure for using TFTP to transfer op-code and firmware files to a Network Utility hard disk is:

1. Configure the IP addresses you will be using.

   If you are using a standard network interface including an Ethernet or Token-Ring LIC, use the Configuration Program or talk 6 to configure an IP address for the interface in the normal way. (From talk 6, you use **add address** in the IP subprocess.) Activate this configuration change before proceeding.

   If you are using the PCMCIA EtherJet card, use **system set ip** to set the following addresses:

   - IP address: the IP address for the EtherJet card
   - Netmask: the mask for the subnet attached to the EtherJet card
   - Gateway address: the IP address of any intermediate router to reach the TFTP server workstation, or the IP address of the workstation itself if there is no intermediate router.

   If you are using SLIP, you cannot change the IP addresses but must use those given in Table 15 on page 108.

2. Transfer the opcode and firmware files.

   From the * prompt, follow this sequence:

   ```
   *t 6
   Config>boot
   Boot configuration
   Boot config>tftp get load mod
   ```

   Respond to the prompts as follows:

   - Server IP address: Put the address of the TFTP server workstation.
   - Remote directory: Put the path name to the directory on the server workstation where all the .ld files are. Use slashes in the direction expected by your server. Uppercase versus lowercase matters only if it matters to your server.
   - Destination bank: Select bank A or bank B. You cannot select the currently active bank.

   Based on the server IP address and the configured Network Utility interface IP addresses, the router selects which of its interfaces to use to reach the server. The router gives success or failure status messages as appropriate.

3. Place your configuration file into the target bank

   Transfer the configuration file you want into a position in the bank where you just placed the new code load. If the new code load is a new MAS release, see “Migrating a Configuration to a New MAS Release” on page 76 for important background on this step.

   - If your new code load is not a new MAS release, or you use only the command-line interface to configure your Network Utility, use the **copy config** command to copy your current configuration where the new load can pick it up.

- If your new code load is a new MAS release and you use the Configuration Program at all, use the Configuration Program to upgrade your configuration. Then use the command **tftp get config** (or any of the other methods described in “Loading New Configuration Files” on page 82) to transfer the upgraded configuration to the target bank.

4. Reboot or schedule a reboot

   To activate the new load immediately, use the following procedure, starting from the **Boot config>** prompt:

   a. Use the **set** command to select the bank you just loaded to boot next, and to select the configuration you just copied or transferred.

   b. Press **Ctrl-p** and then type **reload** to reboot the router

   To activate the new load later, type **timedload activate** from the **Boot config>** prompt to select the bank and configuration and to specify the date and time for the router to reboot. You can answer ″no″ to the questions about loading the bank, because you already did this step.

See the *MAS Software User’s Guide* chapter ″Configuring Change Management″ for more information on the commands in the above procedure.

Using the Firmware

As Table 15 on page 108 shows, you can use either Xmodem or TFTP from the firmware to transfer op-code to the Network Utility hard disk. Xmodem is not recommended because modem speeds are too slow for these large op-code files and Xmodem requires regular interaction. TFTP over LAN interfaces is the preferred transfer method when you are working from the firmware. Nevertheless, this section summarizes all the possible procedures in case you need to use them.

Using Xmodem

The firmware procedure for using Xmodem to transfer op-code and firmware files to a Network Utility hard disk is:

1. Access the firmware main menu using the procedures described in “Boot Options: Fast Boot and Reaching Firmware” on page 45.

2. Make the following sequence of menu selections:

   a. System Management Services (main menu): Option 4, ″Utilities″

   b. System Management Utilities: Option 12, ″Change Management″

   c. Change Management Software Control: Option 12, ″XMODEM software″

   d. Select type: ″Load Image″

   e. Select bank: choose Bank A or Bank B

   The firmware tells you when to start the file transfer.

3. Go to your terminal emulation package and start the transfer of the file LML.ld from your workstation server.

4. After transferring LML.ld, you must transfer every other ″.ld″ module on your workstation server, one by one. LML.ld must be first, but after that the order does not matter. You must include Firm.ld.

   When the file transfer begins, the status of the bank changes to CORRUPT, to indicate that it does not contain a complete valid code load. When the Network Utility has received the last load module, the status of the bank changes to

AVAIL. You can verify that this has happened using option 7, ″List Software″, from the firmware Change Management menu.

5. Boot the router using the op-code you just loaded .

   Use Option 9 ″Set Boot Information″ to select the new op-code bank (and configuration) to boot from. Press **Esc** to reach the main menu, then **F9** to boot the Network Utility with the new op-code.

Using TFTP

The firmware procedure for using TFTP to transfer op-code and firmware files to a Network Utility hard disk is:

1. Access the firmware main menu using the procedures described in “Boot Options: Fast Boot and Reaching Firmware” on page 45.
2. Configure the IP addresses you will be using:

   Follow the menu sequence:

   a. System Management Services (main menu): Option 4, ″Utilities″

   b. System Management Utilities: Option 11, ″Remote Initial Program Load Setup″

   c. Network Parameters: Option 1, ″IP Parameters″

   Set the following addresses:

   - Client IP address: an IP address for the Network Utility LAN card. This is a temporary address that need not be related to the router operational address for that interface.
   - Server IP address: the IP address of the workstation’s LAN adapter
   - Gateway IP address: the IP address of any intermediate router, or repeat the workstation’s IP address if there is none
   - Netmask: the mask for the subnet attached to the Network Utility LAN card
3. Initiate the transfer through these menu selections:

   a. System Management Services (main menu): Option 4, ″Utilities″

   b. System Management Utilities: Option 12, ″Change Management″

   c. Change Management Software Control: Option 10, ″TFTP software″

   d. Select Type: ″Load Image″

   e. Select Bank: choose Bank A or Bank B

   f. Select Load Type: ″Modules″
4. Enter the path on your workstation to the directory with all the load modules.
5. If prompted, select the interface through which you want the firmware to do the file transfer.

   The firmware transfers each load module in turn and gives status messages. On completion, you will be back at the Change Management menu.
6. Boot the router using the op-code you just loaded .

   Use Option 9 ″Set Boot Information″ to select the new op-code bank (and configuration) to boot from. Press **Esc** to reach the main menu, then **F9** to boot the Network Utility with the new op-code.

Upgrading Firmware

Introduction

Firmware is low-level software that drives the power-on and boot logic of Network Utility. It resides in nonvolatile flash memory rather than on the hard disk, so in the event of a failure such as corruption of your operational software load on disk, you can retrieve new software or configuration files and get back up and running. To *upgrade* the firmware means to write a new version of it to flash, replacing the old version.

You need to upgrade the firmware under two conditions:

1. IBM ships a PTF or EPTF that you need to fix a problem, and that PTF or EPTF requires a firmware upgrade. The documentation associated with each PTF or EPTF states whether firmware upgrade is required or not.
2. You want to install a new MAS functional release. Moving to a new release almost always requires a firmware upgrade.

On the Network Utility code download Web pages, there are no separate files containing new versions of the firmware. Instead, the firmware is one of the load modules packed inside the .zip and .tar files along with the operational code load modules. The firmware load module has the file name ″Firm.ld″. Every PTF and EPTF contains a new Firm.ld file, even if that file’s contents are the same as for an older maintenance level.

When you follow the procedures described in “Downloading and Unpacking Files” on page 107 and in “Loading New Operational Code” on page 108, you are downloading a new version of firmware from the web and transferring it to Bank A or Bank B of your hard disk. Placing Firm.ld into a disk bank and rebooting from that bank has absolutely no effect on the active firmware, which is running from flash memory. In order to upgrade to new firmware, you must write the new firmware to flash memory.

Procedure Overview

There are two general methods for downloading new firmware from the Web and getting it into flash memory on Network Utility. The recommended method is to do the firmware upgrade in conjunction with installing new operational code, as follows:

1. Download the new maintenance level of both op-code and firmware from the Web to a local server, as described in “Downloading and Unpacking Files” on page 107.
2. Transfer the new op-code and firmware ″.ld″s to a bank in the Network Utility hard disk, using one of the TFTP or Xmodem procedures described in “Loading New Operational Code” on page 108.
3. Write the copy of Firm.ld that is now already on disk to flash memory, using one of the procedures described in “Local Disk Procedures” on page 113.

In addition to the recommended method, you can also independently transfer only the firmware into the Network Utility, and write it to flash without also transferring and activating operational code. You do this as follows:

1. Download the new maintenance level of both op-code and firmware from the Web to a local server, as described in “Downloading and Unpacking Files” on page 107

page 107. There is no way to download only the firmware from the Web, because it is packaged with operational code.

2. Transfer only Firm.ld to a non-bank location on the Network Utility hard disk, and write it to flash in the same procedure. You can use either Xmodem or TFTP for the file transfer, as described in “File Transfer Procedures” on page 114.

Transferring the firmware independently is not the recommended upgrade method, simply because it duplicates the Firm.ld file transfer you already did when you installed new operational code into bank A or B of the hard disk. The local disk procedures are faster and simpler.

Local Disk Procedures

Follow either of these procedures after you have already transferred a new set of operational code and firmware to hard disk bank A or B, to activate the firmware in that disk bank.

Using the Operational Code

Note: This procedure is available only when you are running MAS V3.2 or later operational code. If you are installing such a level for the first time, you must reboot to the new operational code before you can use this procedure to upgrade the firmware to the same level.

1. Type **talk 6**, then **boot** to reach the Boot Config subprocess.
2. Type **update** to start the firmware upgrade
3. When prompted, select the bank (A or B) into which you transferred the new level of operational code and firmware.

   There is also a ″P″ option, which you can use to rewrite flash with a valid firmware level previously saved on disk (not in bank A or B). You can select this if your flash becomes corrupted (perhaps the system lost power during a flash write) and you want to return to the previous firmware level.
4. The system writes flash memory with the new firmware level from the source location you specified, and creates a new ″recovery image″ (the one selected by ″P″) automatically as appropriate. Do not turn off Network Utility power while the firmware is being updated in flash memory.

The **update** command writes the new firmware level to flash memory, but the updated firmware does not begin to run until your next reboot. The easiest way to install a new maintenance level that requires firmware upgrade from the `Boot config>` prompt is therefore:

1. Use **tftp get load m** to transfer the new op-code and firmware to disk
2. Use **update** to write the new firmware to flash
3. Use **copy** to copy your configuration file to the new code bank
4. Use **set** to select the new code bank for the next boot
5. Type **Ctrl-p** then **reload** to reboot the Network Utility to use the new firmware and the new operational code at the same time.

Using The Firmware

Once you have transferred a new level of operational code and firmware to disk bank A or B, reboot to use the new operational code but stop in the old firmware to write flash memory with the new firmware as follows:

1. Access the firmware main menu using the procedures described in “Boot Options: Fast Boot and Reaching Firmware” on page 45.
2. Make the following sequence of menu selections:
   a. System Management Services (main menu): Option 4, ″Utilities″
   b. System Management Utilities: Option 7, ″Update System Firmware″
   c. F/W Update Options: Option 3, ″Use a Local Image File″

   The firmware asks for a local file name. Enter one of:

   c:\sys0\firm.ld for Bank A

   c:\sys1\firm.ld for Bank B
3. Respond ″yes″ to the question ″Do you want to continue?″ The firmware then starts writing the new firmware to flash memory.
4. Wait and do not turn off the system while the update proceeds.
5. On completion, press **Enter** to restart the system. The new firmware boots up to the new operational code if you enabled autoboot in step 1 of this procedure.

File Transfer Procedures

Follow either of these procedures to transfer only the firmware from a local Xmodem or TFTP server to the Network Utility, and to activate that firmware. As shown in Table 10-2, you use the old firmware user interface in both procedures to initiate the file transfer over any of a number of connection types. As described in “Procedure Overview” on page 112, the local disk procedures may be faster than using these procedures.

Table 16. Loading Firmware

Physical Attachment	Line Protocol	Transfer Protocol	Tool	Default IP Addresses
Service port + null modem Service port + ext modem PCMCIA modem	Async terminal	Xmodem	Firmware	Not applicable
PCMCIA EtherJet Ethernet LIC (10 Mbps) Token-Ring LIC	IP	TFTP	Firmware	Network Utility=10.1.0.2 Workstation=10.1.0.3

Using Xmodem

The firmware procedure for using Xmodem to transfer op-code and firmware files to a Network Utility hard disk is:

1. Access the firmware main menu using the procedures described in “Boot Options: Fast Boot and Reaching Firmware” on page 45.
2. Make the following sequence of menu selections:
   a. System Management Services (main menu): Option 4, ″Utilities″
   b. System Management Utilities: Option 12, ″Change Management″
   c. Change Management Software Control: Option 12, ″XMODEM software″
   d. Select type: ″Load Image″
   e. Select bank: choose Bank A or Bank B

   The firmware tells you when to start the file transfer.
3. Go to your terminal emulation package and start the transfer of the file LML.ld from your workstation server.

4. After transferring LML.ld, you must transfer every other ″.ld″ module on your workstation server, one by one. LML.ld must be first, but after that the order does not matter. You must include Firm.ld.

   When the file transfer begins, the status of the bank changes to CORRUPT, to indicate that it does not contain a complete valid code load. When the Network Utility has received the last load module, the status of the bank changes to AVAIL. You can verify that this has happened using option 7, ″List Software″, from the firmware Change Management menu.
5. Boot the router using the op-code you just loaded .

   Use Option 9 ″Set Boot Information″ to select the new op-code bank (and configuration) to boot from. Press **Esc** to reach the main menu, then **F9** to boot the Network Utility with the new op-code.

Using TFTP

The firmware file transfer and upgradeprocedure using TFTP is:

1. Access the firmware main menu using the procedures described in “Boot Options: Fast Boot and Reaching Firmware” on page 45.
2. Configure the IP addresses you will be using:

   Follow the menu sequence:
   a. System Management Services (main menu): Option 4, ″Utilities″
   b. System Management Utilities: Option 11, ″Remote Initial Program Load Setup″
   c. Network Parameters: Option 1, ″IP Parameters″

   Set the following addresses:
   - Client IP address: an IP address for the Network Utility LAN card. This is a temporary address that need not be related to the router operational address for that interface.
   - Server IP address: the IP address of the workstation’s LAN adapter
   - Gateway IP address: the IP address of any intermediate router, or repeat the workstation’s IP address if there is none
   - Netmask: the mask for the subnet attached to the Network Utility LAN card
3. Initiate the transfer with the following sequence of menu selections:
   a. System Management Services (main menu): Option 4, ″Utilities″
   b. System Management Utilities: Option 7, ″Update System Firmware″
   c. F/W Update Options: Option 1, ″TFTP a Remote Image File″

   Enter the following file names:
   - Local file name: choose a name for a temporary file to be stored in the root directory of the Network Utility hard disk. Do not give a path name. Use a file name extension of 3 characters or fewer.
   - Remote file name: the path and file name (which must be ″Firm.ld″) of the firmware load module on your workstation. It should be in the directory where you unpacked the .zip or .tar file you downloaded from the Web.

   After you select the adapter and port the firmware should use, the router initiates the TFTP get operation.
4. After TFTP completes, respond ″yes″ to the question ″Do you want to continue?″ at the firmware console. The firmware then starts writing the new firmware to flash memory.

5. Wait and do not turn off the system while the update proceeds.
6. On completion, press **Enter** to restart the system. The new firmware boots up to the current operational code.

How to Call for Service and Support

If you bought your Network Utility from an IBM Business Partner or Reseller, contact that party to find out how to receive service and support.

If you bought your Network Utility from IBM, the following forms of assistance are available:

- Hardware or code problem service and support

  For telephone support:

  – Inside the U.S. - call 1 800 IBM-SERV (1 800 426–7378).

  – Outside the U.S. - contact your local IBM service representative for the phone number in your country.

  Before you call, have the machine type, model, and serial number from the backplate of the Network Utility available. If you have a software problem, you may need to have a TFTP server and Internet connection available to transfer a memory dump from the Network Utility and send it to IBM support personnel.

  You can also access IBM Service and Support via the World Wide Web at:

  ```
  http://www.networking.ibm.com/support/networkutility
  ```

  Select the Network Utility product to reach product technical hints, tips, FAQs, and code updates. In addition, you can subscribe to receive notification of future code updates.

- Configuration help and how-to questions for initial installations

  – Inside the U.S. - call 1 800 IBM-SERV (1 800 426-7378). This is a free service.

  – Outside the U.S. - contact your local IBM service representative. This service may not be free outside the U.S.

- Service and support contracts for network design, planning, or problem determination

  – Inside the U.S. - call 1-800-IBM-SERV (1 800 426-7378).

  – Outside the U.S. - contact your local IBM service representative.

Part 3. Configuration and Management Specifics

Chapter 11. Overview

This chapter is an introduction to the part of the book titled “Part 3. Configuration and Management Specifics” on page 117. It gives an overview of possible applications for Network Utility and describes how the other chapters document some of these applications.

Major Network Utility Functions

Using IBM’s Multiprotocol Access Services software technology, Network Utility supports a variety of networking functions. The Network Utility is specifically designed for CPU and memory-intensive functions at network positions requiring a small number of physical interfaces.

Key applications of Network Utility by model include:

Model TN1 - Network Utility TN3270E Server

TN3270E Server

The TN3270E Server function provides SNA host application access to IP desktop users.

One or more Network Utilities can be positioned at a regional office or host data center to provide access for medium-to-large numbers of TN3270 clients distributed throughout an IP network.

Network Utility Model TN1 also supports all the functions of Model TX1.

Model TX1 - Network Utility Transport

Data Link Switching (DLSw)

DLSw provides native SNA end-station (workstation, controller, FEP, or host) connectivity across IP backbone networks. It also performs DLC type conversion like that done in FRAD and X.25 PAD products.

One or more Network Utilities can be positioned at a regional office or host data center to terminate TCP connections from smaller DLSw routers in many branch offices.

Advanced Peer to Peer Networking (APPN)

APPN provides native SNA end-station (workstation, controller, FEP, or host) connectivity across SNA backbone networks. The *Enterprise Extender* feature allows this same connectivity across IP backbone networks.

Network Utilities can be positioned wherever a high-capacity APPN network node is required. You can place one at the edge of an IP network to receive traffic from other Enterprise Extender products. A Network Utility could also provide extended border node function when connecting two different APPN networks.

Channel Gateway

Network Utility supports both ESCON (fiber-optic cable) and Parallel Channel (bus and tag cable) adapters. Using one of these adapters, a Network Utility can serve as a gateway routing both SNA and IP traffic from a S/390 host to local LANs, an ATM network, or to a high-speed serial line.

Network Dispatcher

This function allows a number of IP-based application servers (for example, TN3270 servers, HTTP web servers, or FTP servers) to present a single IP address appearance to client workstations on an intranet or on the Internet. The network dispatcher function fields TCP connection requests from these clients and routes them to an available server. It provides both load balancing among the servers, and high ″logical server″ availability by bypassing failed physical servers.

The Network Utility can be placed at a host data center in front of hosts providing these server functions, or in front of multiple Network Utility Model-TN1s that are providing TN3270E Server function.

High-speed media conversion

Network Utility can serve as a high-speed bridge between interfaces on its supported adapters.

Virtual Private Networking (VPN)

The VPN function consists of a suite of tunneling and security protocols: L2TP, L2F, PPTP, IPSEC, IKE, PKI, Diffserv and LDAP. Taken together, these protocols allow a Network Utility to be configured to use the public Internet as an extension of an enterprise's own private network instead of private WANs and LANs. When so configured, the Network Utility can act as a tunnel termination point for network traffic to an enterprise's remote offices, suppliers and customers. Security policies configured on the router determine dynamically if network traffic needs to be authenticated and/or encrypted, or can flow in the clear. The security policies ensure that the enterprise data can move through the public network as securely, reliably, and flexibly as if they were traversing private lines, and at a significant cost savings.

The Network Utility can be positioned at the boundary point between the Internet and an enterprise's intranet to terminate a large number of either Layer 2 or IPSEC tunnels. These tunnels become an extension of the enterprise network, leveraging the economies and ubiquity of the public Internet.

In this book, we have selected a key subset of the above functions for expanded discussion and example configurations. The chapters that follow cover:

- TN3270E Server, optionally with Network Dispatcher in front of multiple servers
- Channel Gateway, for both SNA and IP traffic
- Data Link Switching, with both TCP termination and local DLC conversion
- Virtual Private Networks

For help in understanding and configuring Network Utility functions other than these, consult the core software publications:

- *MAS Protocol Configuration and Monitoring Reference Volume 1*
- *MAS Protocol Configuration and Monitoring Reference Volume 2*
- *MAS Using and Configuring Features*
- *MAS Software User's Guide*

You may also find configuration help in the following IBM Redbooks. Although they are specific to the IBM 2216 Model 400, some of the configuration scenarios may apply.

- *IBM 2216 Nways Multiaccess Connector Description and Configuration Scenarios Volume 1* (SG24-4957)

- *IBM 2210 Nways Multiprotocol Router and IBM 2216 Nways Multiaccess Connector Description and Configuration Scenarios Volume 2* (SG24-4956)

Chapter Layout and Conventions

Chapters 12 through 20 of this book are organized as follows.

Chapter Layout

Each of the four key functions (TN3270E Server, Channel Gateway, Data Link Switching and VPN) is covered by two chapters:

- An introductory chapter that:
  - Summarizes the supported function
  - Discusses example network configurations
  - Introduces how to manage the function
- An ″Example Configuration Details″ chapter containing:
  - Labelled diagrams of key example configurations
  - Matching tables with configuration parameters for both Configuration Program users and command-line users

  The configurations shown and described in the four ″Example Configuration ″ chapters are actual working configurations. Binary configuration files matching these configurations are downloadable from the World Wide Web. To access these files, follow the `Support` and `Downloads` links from:

  ```
  http://www.networking.ibm.com/networkutility
  ```

In addition, “Chapter 18. Sample Host Definitions” on page 259 provides detailed examples for configuring IBM host software products to match Network Utility configurations.

Example Configuration Table Conventions

The configuration parameter tables used in the four ″Example Configuration″ chapters all follow the same format. Table columns and conventions are as follows:

Configuration Program Navigation
The sequence of folder and panel names to follow until you reach the panel where you enter parameter values.

Configuration Program Values
Parameter names and their values.

If the configuration program panel shows parameters not listed in the table, we used their default values. ***Your configuration must be for a Network Utility and not a 2216-400 to have the correct default values.***

Command-Line Commands
The commands you type to configure the same parameters using the command-line interface, as follows:

- Command sequences start from the talk 6 prompt `Config>`. Where needed, the initial command shows how to get to the right place in the menu system and the resulting command prompt.

- Commands without parameters specified will either ask for the input values or have no parameters. Parameter prompts from the system are shown in `this font`.
- Where the value prompts and values you type are self-explanatory, the details are not shown.
- ″Accept other defaults″ means that there are other parameter prompts for which you should accept the default values (by pressing **Enter**).

Notes Numbers that reference comments at the bottom of each table.

Chapter 12. TN3270E Server

Overview

This section introduces TN3270 and summarizes the TN3270E server function implemented in Network Utility.

What is TN3270?

Many companies today are considering the consolidation of their WAN traffic onto a single IP-only backbone. At the same time, other companies are simplifying their workstation configurations and attempting to run only the TCP/IP protocol stack at the desktop. However, most of these companies still require access to SNA application hosts.

TN3270 meets these requirements by allowing you to run IP from the desktop over the network and attach to your SNA host through a TN3270 server. The clients connect to the server using a TCP connection. The server provides a gateway function for the downstream TN3270 clients by mapping the client sessions to SNA dependent LU-LU sessions that the server maintains with the SNA host. The TN3270 server handles the conversion between the TN3270 data stream and an SNA 3270 data stream.

To deploy a TN3270 solution, you install TN3270 client software on desktop workstations[13] and TN3270 server software in one of several places discussed below. Client software is available from IBM and many other vendors, and runs on top of the TCP/IP stack in the workstation. A given client product provides one of two possible levels of standards support:

- Base TN3270 client

  These clients conform to RFC 1576 (TN3270 Current Practices) and/or RFC 1646 (TN3270 Extensions for LU name and Printer Selection).
- TN3270E client

  These clients conform to RFC 1647 (TN3270 Enhancements), and RFC 2355 (TN3270 Enhancements).

A server implementation that can support TN3270E clients is called a TN3270E server.

Placement of the TN3270 Server Function

The TN3270 server function can be placed in a variety of products and positions within a network, including:

- In the SNA host itself

  IBM and several other vendors provide host TN3270 server software that sits on top of the host TCP/IP stack and connects within the host to VTAM.
- In a router or Network Utility in the network

  IBM and other vendors provide TN3270 server function in networking hardware products. You can place these products directly adjacent to the SNA host, or at any position in the network where you have SNA connectivity to the host. If you

[13] You can also find small, dedicated TN3270 client products that represent printers.

are using IBM routers (2210 or 2216) or Network Utilities, and your host is running APPN, you can use Enterprise Extender technology to place the server at any position where you have IP connectivity to the host.

- In a software product in the network

  IBM and other vendors provide TN3270 server software products that you install on mid-range servers that use operating systems such as AIX, OS/2, or Windows/NT. You can place these products at any position in the network where you have SNA connectivity to the application host.

The choice of TN3270 server product and network position is a complex one, involving such factors as:

- Host capacity and cycle impact
- Price for performance and capacity
- Availability
- Impact of server failure
- Scalability

Network Utility provides a high-performing TN3270E server implementation that scales to large networks. By combining it with the Network Dispatcher feature, you can implement server redundancy and load sharing in large TN3270 installations. You can also place a Network Utility out into an SNA or IP network away from the data center and get the same advantages of scalability, incremental addition, and reduced impact of server failure.

Network Utility TN3270E Server Function

Standards Compliance

The Network Utility implementation of TN3270E server supports these RFCs:

RFC 1576 - TN3270 Current Practices

RFC 1646 - TN3270 Extensions for LU names and Printers

RFC 1647 - TN3270 Enhancements

RFC 2355 - TN3270 Enhancements (obsoletes RFC 1647)

It can handle both base TN3270 and TN3270E clients at the same time.

Host Connectivity

As mentioned above, the path from a TN3270 client to the SNA host consists of two pieces:

- A TCP connection over IP from the client to the server
- An SNA LU-LU session from the server to the host

The form of the SNA connection from the server to the host depends on how the server represents PUs and dependent LUs. When you are using Network Utility as your TN3270 server, you can configure either of two different ways to establish links and represent PUs and LUs to VTAM:

- Using SNA subarea links

  You set up Network Utility this way when you are not running APPN at the host. You configure a separate DLC-layer link to the host for every PU (maximum of 253 LUs). Multiple PUs require multiple parallel host links. SNA frames arriving at Network Utility on one of these links flow directly to the corresponding internal PU.

Subarea host links must be a single DLC-layer hop to the product providing the SNA subarea boundary function. Typically, this product is either NCP running in a FEP, or is VTAM itself in the host. The subarea link from the Network Utility can traverse bridges or other DLC-layer forwarding mechanisms (such as protocol converters or external DLSw routers). Network Utility supports the following link types for subarea host attachment:

  - Token-Ring: physical, ATM LAN emulation, or channel LSA
  - Ethernet: physical, ATM LAN emulation, or channel LSA
  - FDDI: physical only
  - Frame relay PVCs: bridged or routed RFC 1490/2427 formats
  - DLSw

- Using an APPN Dependent LU Requester (DLUR) link

  You set up Network Utility this way when you are running APPN with its Dependent LU Server (DLUS) function at the host. You configure one DLC-layer link to the host to carry the DLUR-DLUS ″pipe,″ even if you are defining multiple local PUs. SNA frames arriving at Network Utility on this link flow to the DLUR function, which redirects them to the correct internal PU.

  When you are using DLUR, you can route through an APPN network using either ISR or HPR routing to reach the host. Network Utility supports the following link types as the ″first hop″ APPN link to the host:

  - Token-Ring: physical, ATM LAN emulation, or channel LSA
  - Ethernet: physical, ATM LAN emulation, or channel LSA
  - FDDI: physical only
  - Frame relay PVCs: bridged or routed RFC 1490/2427 formats
  - ATM (native, not LAN emulation): HPR only
  - Channel MPC+: HPR only
  - PPP
  - SDLC: ISR only
  - X.25: ISR only
  - DLSw: ISR only
  - IP (Enterprise Extender): HPR only

  Note especially that when using DLUR and HPR routing, you can place a Network Utility TN3270E server across an IP network from the SNA application host. Enterprise Extender maintains session-level class of service and transmission priority across the IP network.

General TN3270E Server Configuration

This section covers general information about configuring Network Utility TN3270 server support. For specific example configurations, see page 132.

Configuring TN3270 Subarea under the APPN Protocol

In the Network Utility implementation of TN3270 server, all SNA functions are bundled within the APPN protocol. This means that *even when you are configuring SNA subarea attachment and your SNA host is not running APPN*, you must use the configuration and console services of the APPN protocol. In particular:

- You must go through the APPN protocol at the command line and the Configuration Program to configure ports, links, and TN3270 server functions

- You must go through the APPN protocol at the command line to use TN3270 monitoring commands
- You must still configure APPN at the node level

When you configure SNA subarea support, Network Utility does in fact still function as an APPN network node, but only on links to other APPN nodes. If the *only* ports and links you configure are those for SNA subarea host attachment, then the APPN function serves no purpose.

Configuring in the APPN Environment

APPN and TN3270 server are fully configurable both from the Configuration Program and from the command line. From the Configuration Program, the TN3270 configuration parameters are always available. If you create a TN3270 configuration and download it to a Network Utility Model TX1, which does not support TN3270 server function, the Network Utility ignores the TN3270 part of the configuration. If you are working from the command line on a Model TX1, the commands for configuring and monitoring TN3270 simply do not appear on the APPN menus.

To change an APPN/TN3270 configuration from the Configuration Program, you make the change, transfer the configuration to the Network Utility, and reboot for it to take effect.

To change an APPN/TN3270 configuration from the command line, you move to talk 6, type **p appn**, and then issue the commands to make the change. You have two options for activating the change:

- Write the configuration to disk and reboot Network Utility to activate it.
- Issue the talk 6 APPN **activate** command to dynamically activate the modified APPN/TN3270 configuration.

  Depending on the configuration items you changed, APPN either makes the change immediately, or restarts APPN (but not the entire Network Utility) to activate the change. For the latter case, if you move to talk 5 and type **p appn** while APPN is restarting, you get the message `APPN is not currently active`. You can poll with talk 5 commands to see when the restart is complete.

Implicit and Explicit LU Naming and Mapping

When you configure Network Utility’s TN3270 server function, you create a local LU name for every one of the potential concurrent client sessions the Network Utility is intended to support. The LU name you define in the Network Utility need not have any relation to LU names in VTAM.

When a TN3270 client connects to a server over TCP, it can request a specific LU name, or it can place a generic request for any LU of a certain type. If you are configuring a client to request a specific name, you specify one of the local names defined at the server (Network Utility), not a VTAM LU name.

Because a single Network Utility can support thousands of LUs with similar characteristics, it does not require you to individually configure each LU. Rather, you can create a large pool of *implicit* LUs to satisfy clients that do not request a particular LU name. You then add a small number of *explicit* LUs to satisfy the clients that do request a particular name[14].

As you will see in the example configurations, you define implicit LUs in groups as you define each local PU. You specify an LU name mask, number of LUs and to which pool the LUs belong. To configure an explicit LU, you specify an LU name and an NAU address (2-254). When the Network Utility activates the configuration, it reserves the NAU addresses for explicit LUs, and then generates names for the implicit LUs using the group name mask and one of the available NAU addresses.

MAS V3.2 PTF01 introduced a number of significant functional enhancements in the area of LU definition and client mapping:

- You can define named pools of LUs.

  LU pooling is an enhancement to the TN3270E server function that makes it easier for you to configure some TN3270E networks. This function allows you to group SNA LUs into named “pools”. TN3270E clients can then request a connection using the pool’s name as an LU name. The TN3270E server will then choose an LU from the specified pool to serve the client’s request.

- You can configure mappings between client IP addresses and LU or LU pool names.

  The TN3270E server Client IP Address to LU Name Mapping function provides a mechanism for administrators to control client access to the TN3270E server’s LUs.

  Mapping enhances central administration by allowing the administrator to configure which SNA resources (LUs or Pool) client IP address or subnets will map to and use without modifying client configurations.

- The server can send VTAM a list of dependent LU addresses for each PU, so that VTAM can dynamically create its own LU definitions.

  The dynamic definition of dependent LUs (DDDLU) is a VTAM facility that allows the logical units to be known by VTAM when they connect to VTAM, rather than during major node activation of the related PU.

  If prompted by VTAM, the TN3270E server function will use DDDLU to create its local LUs in VTAM. Instead of sending all of the LU definition requests when the ACTPU is received, the server will wait until the LU actually needs to be defined. The LU definition will occur when a TN3270 client connects in and needs an LU that has not been defined to VTAM.

- You can configure multiple local TCP ports for the TN3270 server function.

  This enhancement allows you to define multiple TCP ports for the TN3270E server to “listen” on. This support allows clients to specify the SNA resource they want using a port number.

- You can disable TN3270E negotiation.

  This enhancement allows you to specify whether the added port will negotiate to be a TN3270E server, as opposed to conforming only to the base TN3270E support. This is needed by some base TN3270 clients, who do not properly handle receiving initial TN3270 Extended negotiations.

Refer to the MAS V3.2 or later *MAS Protocol Configuration and Monitoring Reference Volume 2* for more information on configuring these functions.

[14] The implicit/explicit distinction is solely within the Network Utility. A client can request an implicit LU name, and the Network Utility will satisfy the request if the LU is available. The key point is that the server function will never assign an explicit LU to a client unless the client specifically requests that LU name.

MAS V3.3 introduced Host Initiated DDLU:

- Host Initiated DDLU removes the requirement to redundantly define the LUs to the TN3270E server, if the LUs have already been defined to VTAM. TN3270E server will dynamically define each of the LUs as they are activated on VTAM.

Example Configurations

Network Utility as a TN3270E server can be deployed in several configurations. For example, it can be placed either in the remote branch or in the data center. It can attach to the host via a traditional SNA subarea connection or it can use APPN. In the data center, it can be placed in a channel-attached configuration or it can be a stand-alone server that resides on the campus LAN (or ATM cloud) using the channel-attached connection provided by an existing IBM 3745/46 Communication Controller, 2216-400, 3172 Interconnect Controller, an OSA adapter, or an OEM gateway.

One of the most important elements of a TN3270 implementation is scalability. The Network Utility solution can scale to very large configurations while providing high availability and redundancy.

The following scenarios show you how to effectively utilize the Network Utility as a TN3270E server.

TN3270 via a Subarea Connection to an NCP

This scenario (shown Figure 6 on page 133) shows a traditional SNA subarea network with all host access occurring through an IBM 3745/46 Communication Controller with the IBM Network Control Program (NCP). The Network Utility is installed to provide TN3270 server support for downstream workstations both in the local campus and in the remote sites. The Network Utility attaches to the host through the FEP via a normal subarea connection.

Up to 20 000 TN3270 sessions can be handled with a single Network Utility installed as shown in Figure 6 on page 133. As your network grows, the solution can be scaled simply by adding more TN3270E server capacity via additional Network Utilities. You can also set up automatic load balancing among your TN3270E servers by installing a separate IBM router or Network Utility to serve as a Network Dispatcher. (See “Highly Scalable, Fault-Tolerant TN3270E” on page 136 for an example of how to scale the network.)

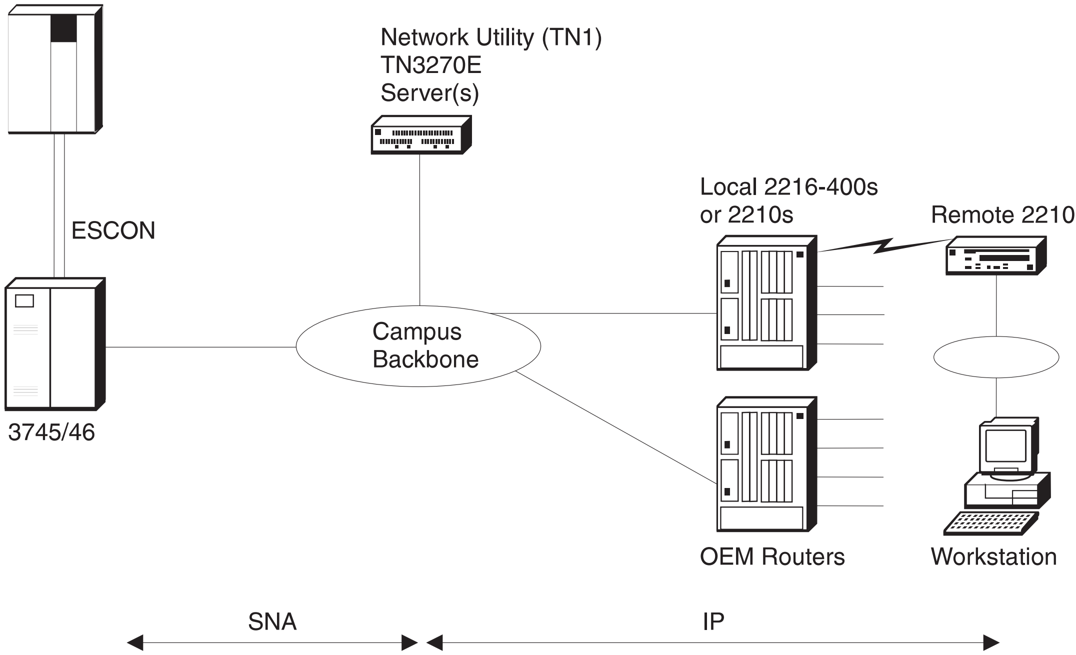


Figure 6. TN3270 via a Subarea Connection through a 37xx

Keys to Configuration

The configuration of the TN3270E server function is very straightforward in this scenario. However, the following points are worth noting:

- There is both an APPN and a subarea implementation of the TN3270E server. Both require APPN support to be installed on the Network Utility and both are configured within the APPN configuration process. This is true even though a pure subarea configuration does not use the APPN function. This is an implementation statement as the TN3270E server function uses the APPN SNA stack for both subarea and APPN connections to the host.

  Note also these additional points relating to APPN and TN3270E server configuration:

  – APPN support must be enabled.

  – You must define a port and one or more link stations to define the connection to VTAM.

  – For subarea configurations, defining a link station and specifying to solicit an SSCP session implicitly defines a PU on the Network Utility. This PU will support up to 253 downstream LUs. If you need more than 253 LUs, then you need to define more than one link station. Each link station needs to use a different service access point (SAP) and a different local node ID (IDNUM).

- When configuring the parameters for the TN3270E server, you can set the IP address of the server to either the internal box IP address or to one of the interface IP addresses. Keep in mind that the address you select for TN3270 may be unavailable for using normal IP Telnet to manage the box.[15]
- The downstream LUs can be defined either as explicit or implicit.

[15] If you need to use Telnet at this same address, you can configure the TN3270E server to use another port (24 for example) so that telnet can use port number 23. This requires that the TN3270 client workstations be configured to use this same port.

- – Use explicit definitions when you need to ensure that the device will always use the same LU name. (For example, printers would normally use explicit definitions.)
- – Use implicit definitions when you have a large group of devices that can use a common pool of available LUs and do not need to use the same LU name every time.

For a complete look at the configuration parameters needed for this scenario, see Table 19 on page 152.

TN3270 via a Subarea Connection through a Channel Gateway

This scenario, shown in Figure 7, is similar to the previous scenario except that here, the Network Utility attaches to the host through a LAN channel gateway such as an IBM 3172, an IBM 2216, an IBM 3746 with the Multiaccess Enclosure (MAE) or an OEM device. These gateways use External Communications Adapter (XCA) pass-through and do not provide the SNA boundary function normally provided by an NCP. With a gateway, this function is provided by VTAM.

If you have an existing gateway with a TN3270 server configured, you can use the Network Utility to offload the existing TN3270 workload or to provide additional TN3270 capacity as your network requirements grow.

An existing 2216 or a 3746 allows you to have multiple channel connections to the host while you can incrementally install Network Utilities for your TN3270E server requirements. The dynamic load balancing features of network dispatcher can be used to optimize efficiency.

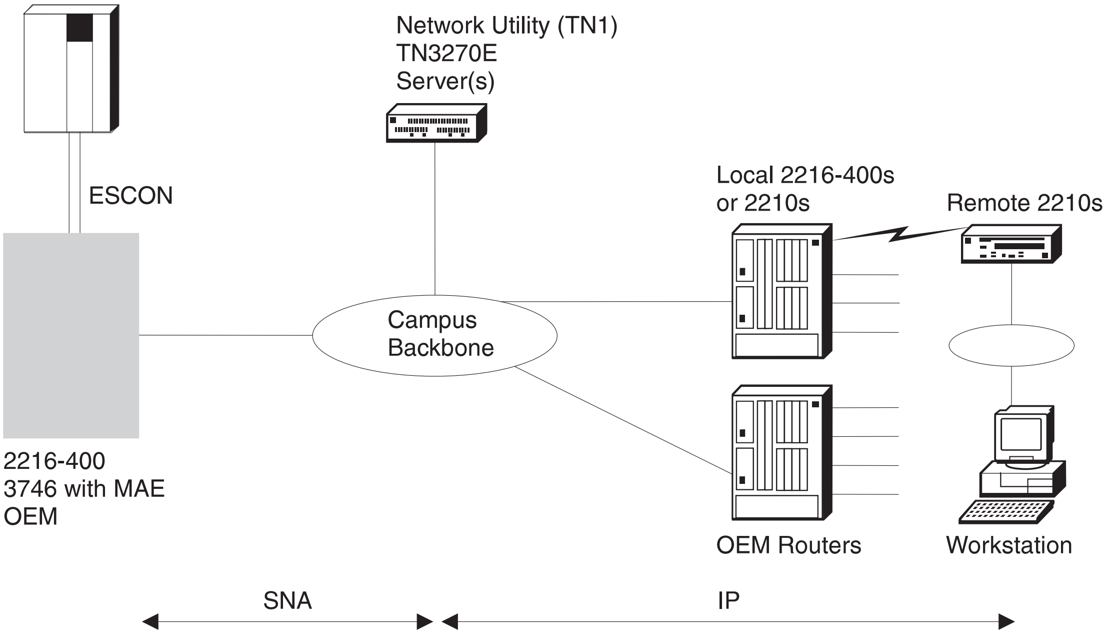


Figure 7. TN3270 via a Subarea Connection through a LAN Gateway

Keys to Configuration

From the Network Utility perspective, the configuration of this scenario is identical to the previous one. The host definitions are also identical. For both scenarios, you just have to define the switched major nodes for the PUs in the TN3270E server.

TN3270 through an OSA Adapter

This scenario is shown in Figure 8. Here, the Network Utility attaches to the host through the S/390 Open Systems Adapter (OSA). Like the previous gateway scenario, the SNA boundary function is in the host.

While the TN3270 server function can reside on the host itself, many customers prefer to offload this function externally to another platform. The Network Utility meets this requirement well by providing scalable, cost-effective TN3270E server function without changing your method of host attachment. This allows you to leverage your existing investments.

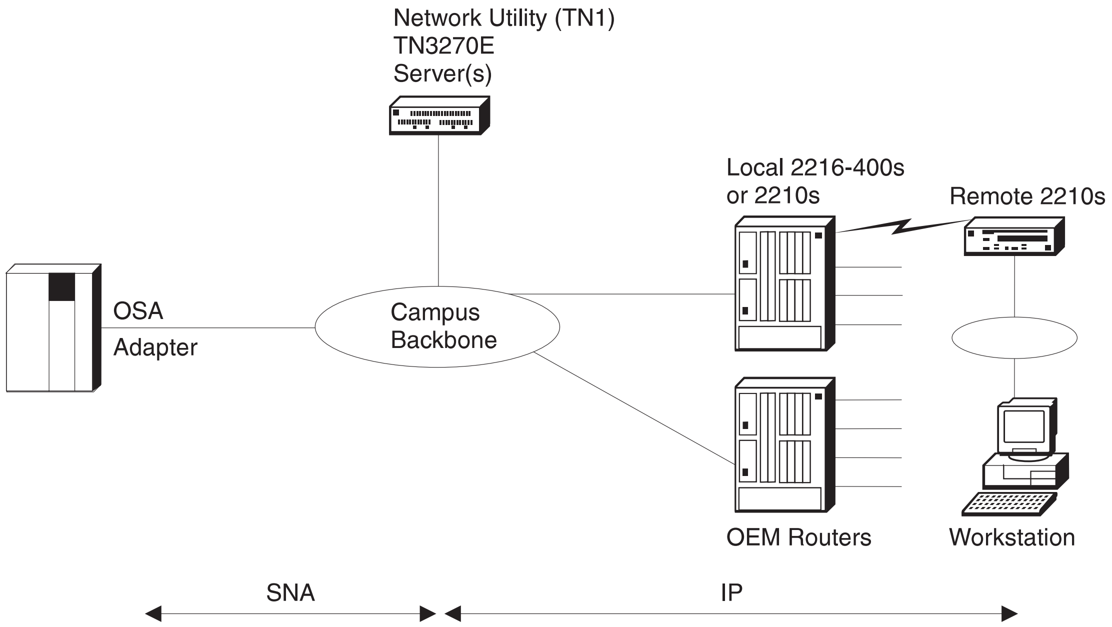


Figure 8. TN3270 via an OSA adapter

Keys to Configuration

From the Network Utility perspective, the configuration of this scenario is identical to the previous two.

TN3270 Subarea SNA over DLSw

TN3270E over subarea SNA over DLSw connection is used for eliminating a second router requirement in the remote nodes or branches. Without this function, you would need two routers in a branch to be able to run TN3270E Server over IP, as shown in Figure 9 on page 136. With the TN3270E Server support on DLSw Subarea, two Network Utility boxes are not needed since DLSw and TN3270E supports are merged in a single Network Utility.

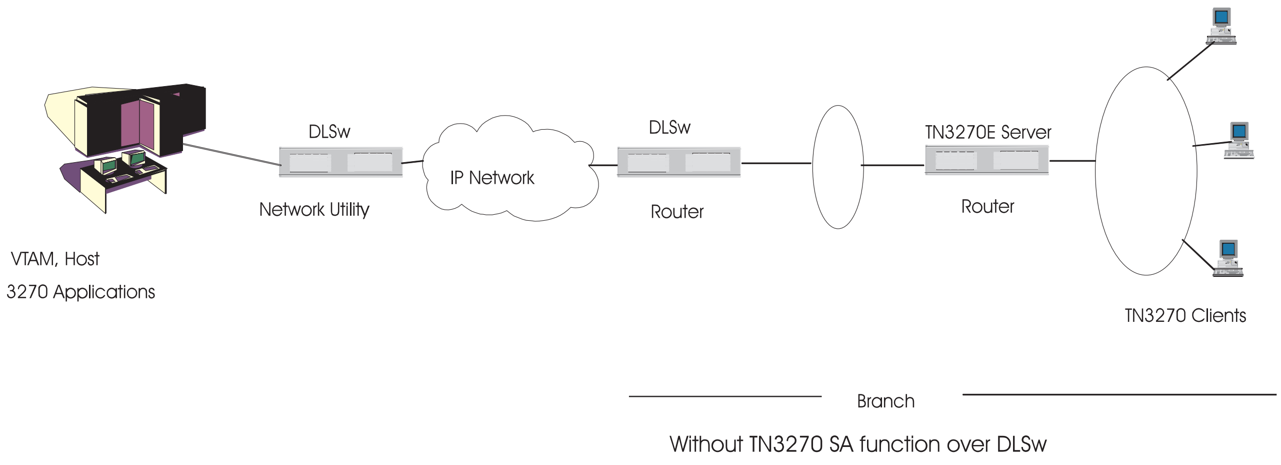


Figure 9. A Typical Branch Configuration ***Without*** *TN3270E Over DLSw Subarea Support on Network Utility*

As shown in Figure 10, TN3270E Server and DLSw are supported in a single Network Utility with Multiple PU's Subarea function. In this function, there is an APPN/DLSw interface within Network Utility. It is possible to run 58 link stations over this interface—in other words, you can run 58 SNA PU Type 2's. The new multiple PU's subarea function can run over Local or Remote DLSw. Local DLSw supports LSA ESCON connection, X.25 QLLC and SDLC links.

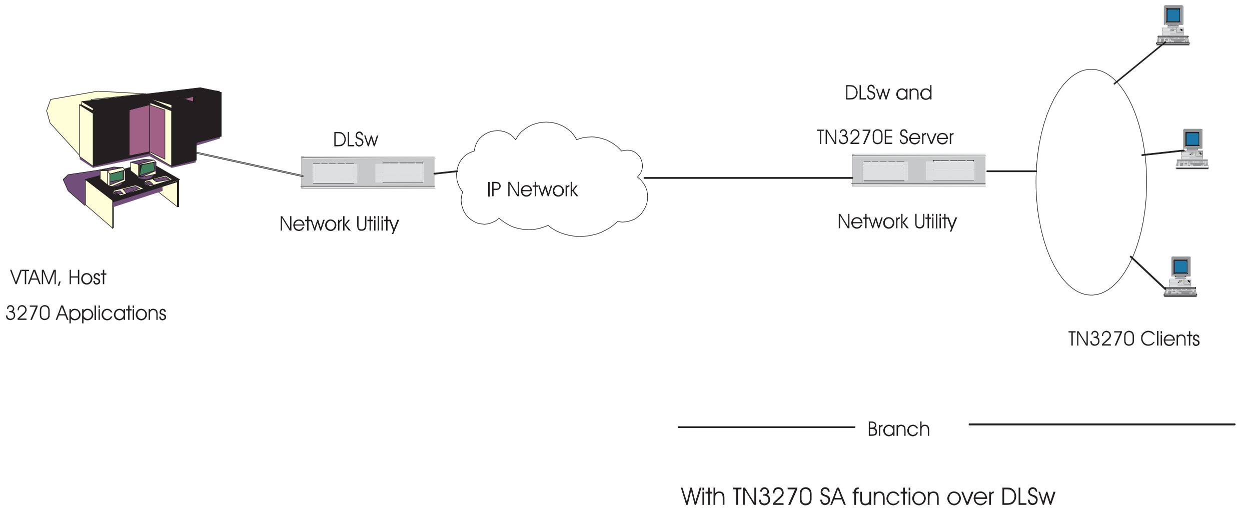


Figure 10. A Typical Branch Configuration ***With*** *TN3270E Over DLSw Subarea Support on Network Utility*

Highly Scalable, Fault-Tolerant TN3270E

This scenario, shown in Figure 11 on page 137, is an extension of the one discussed in "TN3270 via a Subarea Connection to an NCP" on page 132. Here, the solution is scaled with multiple Network Utility devices to provide TN3270E server support for large 3270 environments. Also, a separate Network Utility is configured as a network dispatcher and deployed to provide load balancing[16].The new Network Dispatcher Advisor for TN3270 allows the Network Dispatcher to

[16] As of MAS V3.2, the Network Dispatcher function can also dispatch client sessions to the TN3270 server function running in the same Network Utility.

collect load statistics from each Network Utility TN3270E server in real time to achieve the best possible distribution among the TN3270 servers.

The solution provides high availability in the event of a failure in one of the TN3270E servers. The server that the client session is dispatched to is transparent to the user. If a failure occurs, the sessions through that server are lost but the users simply log back on to the host through another Network Utility using the same destination IP address for the TN3270E server.

The Network Dispatcher function can also utilize redundant hardware, with a second Network Utility configured as a Network Dispatcher and serving as a backup to the primary one.

With this configuration, you can scale your TN3270E support to any size simply by adding additional TN3270E server capacity. You can do this incrementally and non-disruptively as your network requirements grow.

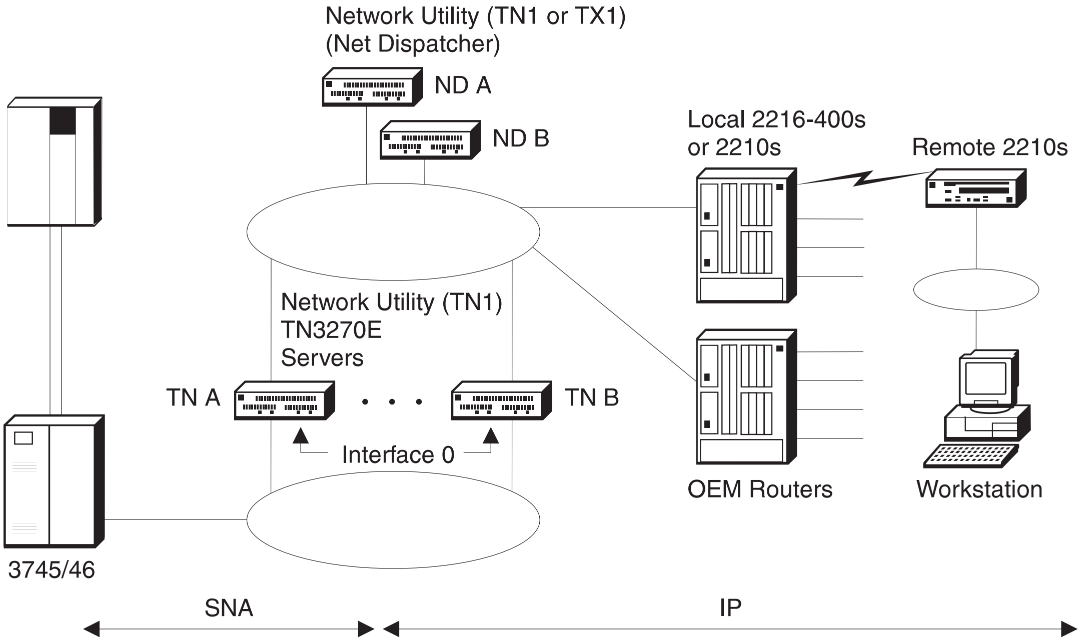


Figure 11. Highly Scalable, Fault-Tolerant TN3270E

Keys to Configuration

As far as the TN3270E server is concerned, the configuration is the same whether you have a Network Dispatcher or not. In fact, the TN3270E server is unaware that the traffic from the clients is being dispatched through another machine. See “TN3270 via a Subarea Connection to an NCP” on page 132 for the basic configuration points for a TN3270E server. See Table 20 on page 158 for the complete set of configuration parameters for the TN3270E servers for this scenario.

However, the IP addressing needs special attention in this configuration for high availability. In “TN3270 via a Subarea Connection to an NCP” on page 132, the TN3270E server was configured with the same address as the router ID (also the same address as the LAN interface). In a Network Dispatcher environment, the IP addressing is somewhat different.

A Network Dispatcher and one or more TN3270E servers form what is called a cluster. An IP address is defined for the cluster and workstations send their TN3270 packets to this IP address. The Network Dispatcher receives these packets and forwards them on to a server in the cluster for processing.

Because the Network Dispatcher does not alter the destination IP address of these packets, each TN3270E server also needs to be configured with this same IP address. However, you have to make sure that the TN3270E servers do not broadcast this address via OSPF or RIP to the network because you do not want these servers to respond to the cluster address. Only the Network Dispatcher should respond to the cluster address[17].

The router must know the TN3270E server's IP address in order to forward packets to the server function. One way to make this address known to the router is to specify it to an interface as a secondary address. Figure 12 shows an example of this IP addressing scheme for the highly available, fault-tolerant TN3270 configuration depicted in Figure 11 on page 137.

```
TN3270E Server #1 (TNA):
    Internal address  172.128.252.3
    Interface 0       172.128.2.3      (2nd address: 172.128.1.100)
    Interface 1       172.128.1.3
    OSPF Router ID    172.128.1.3
    TN3270E Server    172.128.1.100    (same as cluster address)

TN3270E Server #2 (TNB):
    Internal address  172.128.252.4
    Interface 0       172.128.2.4      (2nd address: 172.128.1.100)
    Interface 1       172.128.1.4
    OSPF Router ID    172.128.1.4
    TN3270E Server    172.128.1.100    (same as cluster address)

Network Dispatcher #1 (NDA):
    Internal address  172.128.252.1
    Interface 0 addrs 172.128.1.1
    OSPF Router ID    172.128.1.1
    Cluster address   172.128.1.100
       Port 23
         Server 1     172.128.1.3
         Server 2     172.128.1.4

Network Dispatcher #2 (NDB):
    Internal address  172.128.252.2
    Interface 0 addrs 172.128.1.2
    OSPF Router ID    172.128.1.2
    Cluster address   172.128.1.100
       Port 23
         Server 1     172.128.1.3
         Server 2     172.128.1.4
```

Figure 12. IP addressing for the Highly Scalable, Fault-Tolerant TN3270 Scenario

Note that the cluster address is assigned as a second IP address on interface 0 of the Network Utility machines. In this scenario, the LAN segment that interface 0 attaches to does not carry any IP traffic – only the SNA subarea traffic from the TN3270E server to the host.

[17] The cluster address cannot be pinged. The Network Dispatcher does not respond to pings to the cluster address. It only processes TCP and UDP packets.

The configuration of the Network Dispatchers is standard. For the complete set of configuration parameters needed for this scenario, see Table 21 on page 163 for the primary network dispatcher. For differences from this configuration for the backup network dispatcher, see Table 22 on page 167.

TN3270 Via DLUR over APPN

This scenario, shown in Figure 13, uses APPN to communicate with the host. The Network Utility uses APPN High Performance Routing (HPR) and establishes a Rapid Transport Protocol (RTP) session with the host. HPR is used all the way from the TN3270E server to VTAM. In case of a failure, this assures nondisruptive session switching to an alternate path if you have parallel gateways. This is especially important in Parallel Sysplex environments.

In addition, HPR is supported over IP through the Enterprise Extender feature of the Network Utility. This is important if you want to place your TN3270E server in a remote location and use IP to transport the APPN traffic back to your data center.

The channel gateway is an APPN network node performing APPN Automatic Network routing (ANR) for the RTP session between the Network Utility and the host.

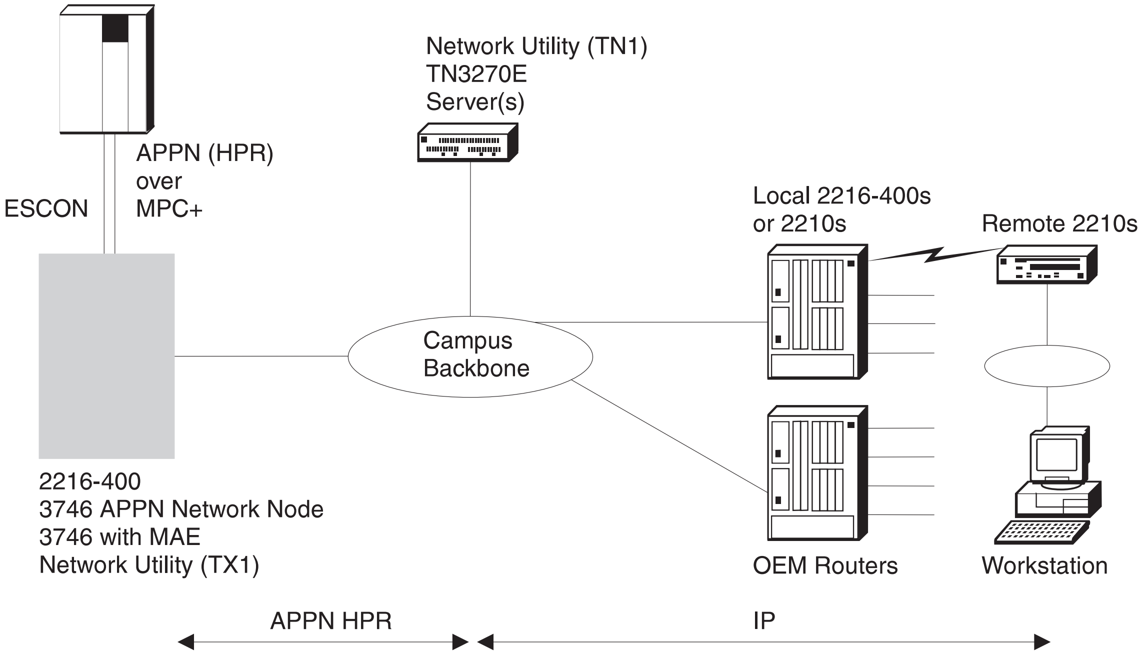


Figure 13. TN3270 Via DLUR over APPN

When connecting a TN3270E server to the host via APPN, you must configure DLUR support on the Network Utility. The DLUR feature extends to APPN nodes the support of T2.0 or T2.1 devices containing dependent LUs. The DLUR function on an APPN network node works in conjunction with a DLUS. The DLUS function is usually provided by VTAM, although it may reside in any part of a mixed APPN/subarea network.

The dependent LU flows (SSCP-PU and SSCP-LU) are encapsulated in a LU 6.2 pipe (CP-SVR) established between the DLUR APPN node and the DLUS SSCP. The CP-SVR pipe is made up of a pair of LU 6.2 sessions using a new CPSVRMGR mode between the DLUR and the DLUS. This pipe brings the SSCP function (in the DLUS) to the DLUR APPN node where it can be made available to attach T2.0/T2.1 nodes containing dependent LUs.

Keys to Configuration

From a downstream workstation perspective, the TN3270E server appears the same whether that server is using SNA subarea or APPN to communicate with the host on the uplink. At the Network Utility, you configure the base TN3270 server parameters the same way as in the SNA subarea scenarios, but the way you configure the local PUs differs. Instead of associating each PU with a subarea link, you configure local PUs without any link association. The DLUR function is responsible for routing traffic on the DLUS-DLUR pipe to and from these local PUs.

APPN requires DLUR support to be configured in the Network Utility. DLUR is quite simple to configure with the only required parameter being the CP name of the DLUS, which is VTAM.

You have to make some additional host definitions for APPN and DLUR support. See “Chapter 18. Sample Host Definitions” on page 259 for an example of these commands.

For a complete look at the configuration parameters needed for this scenario, see Table 23 on page 169.

Distributed TN3270E Server

The previous configurations showed how the Network Utility can be deployed in the data center to centralize the TN3270E server function in your network. This configuration, shown in Figure 14 on page 141, shows just one example of how the Network Utility can also be placed in a remote location to provide distributed TN3270E server capability.

In this configuration, the Network Utility is providing TN3270E server service to workstations in the remote location. As always with a TN3270 configuration, the workstations are using IP to communicate with the TN3270E server. The TN3270E server is using DLUR over an APPN connection back to the host in the data center.

In this example, the corporate WAN is a public Frame Relay network that carries IP traffic only. Therefore, the Network Utility is configured to use the Enterprise Extender feature which allows the APPN HPR traffic to be carried over the IP-only WAN.

The Enterprise Extender traffic is terminated at the host gateway, which decapsulates the HPR traffic and then passes the APPN traffic through the network node onto the MPC+ path to the host. This is a very fast, low-overhead packet-forwarding function, so a single gateway can handle a large amount of traffic.

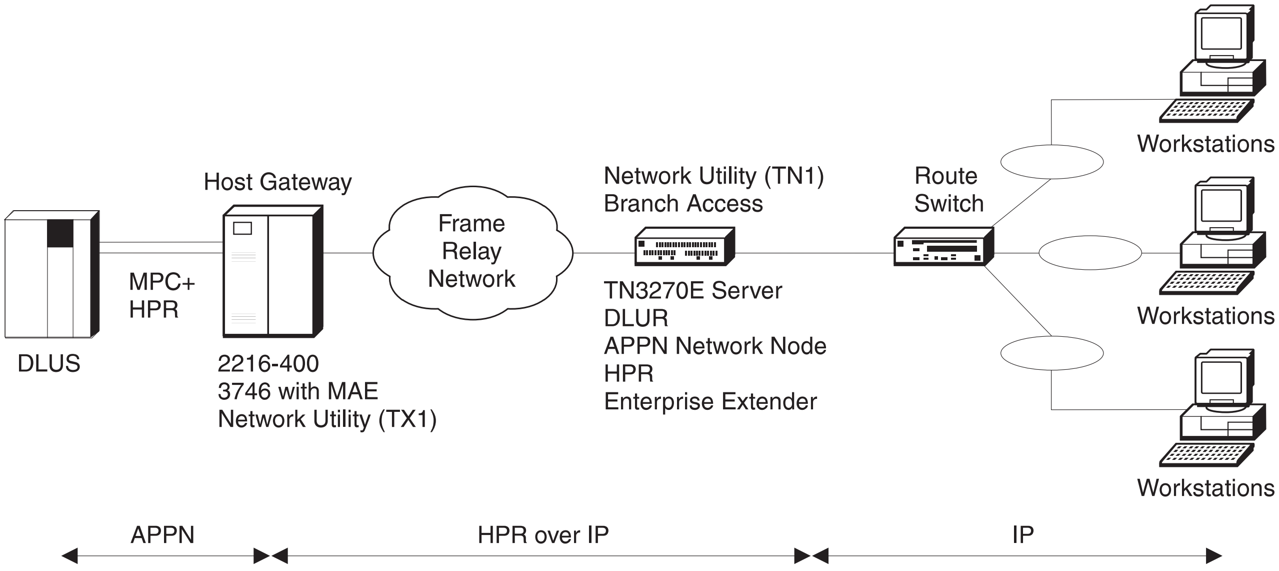


Figure 14. Distributed TN3270E Server

Keys to Configuration

From a downstream workstation perspective, the TN3270E server appears the same whether it is in the remote branch or in the data center regardless of whether the upstream connection to the host is using SNA subarea or APPN. Therefore, the TN3270E server function in the Network Utility is configured exactly the same as in the previous scenarios.

APPN and DLUR are configured the same as in “TN3270 Via DLUR over APPN” on page 139 with one exception, which is the port definition for APPN over the frame relay IP link. When configuring APPN to use HPR over IP (the Enterprise Extender feature), you specify a port type of IP. Then when adding the link station for this port, instead of specifying the MAC address of the adjacent FEP as was done in “TN3270 Via DLUR over APPN” on page 139, you specify the IP address of the other end of the HPR over IP network, which is the host gateway in this example[18]. The IP network is responsible for the delivery of the traffic to the host gateway over the best path available. You are assured a reliable transport because the connection between the TN3270E server and the host uses an RTP session.

Managing the TN3270E Server

This section introduces some of the ways in which you can monitor and manage the TN3270E server function.

Note: The monitoring functions described in this section assume that you are running MAS V3.2 or later operational code. MAS V3.2 introduced several new TN3270 monitoring commands, as well as a TN3270E submenu.

[18] The host gateway must also be configured with an HPR over IP port in much the same manner as described here.

Command-Line Monitoring

To view currently running TN3270 server status from the command line, move first to talk 5 and enter **p appn**. If you get the message `Protocol APPN is available but not configured`, you need to complete your base APPN configuration and reboot Network Utility to activate APPN. As discussed in “Configuring TN3270 Subarea under the APPN Protocol” on page 129, you need APPN to be active even if you are using only TN3270 subarea connectivity.

Once you have reached the APPN monitoring prompt `APPN >`, type **tn** (short for ″TN3270E″) to reach the submenu for monitoring TN3270E server status.

The following commands are then available at the `TN3270E >` monitoring prompt:

list status

If the system responds *TN3270E is not configured or not active*, you did not enable the TN3270 server function adequately in the currently active APPN configuration. If you get this error and you did configure the function, perhaps the TN3270 server IP address you chose is not active as an interface address or as the internal IP address. Consult the examples of TN3270 configurations in Chapter 13 for other possible reasons, then change your APPN/TN3270 configuration and activate it as described in “Configuring in the APPN Environment” on page 130.

If the server function is active, this command provides the following information:

- Configuration information currently in use

  TN3270E IP Address

  The server IP address to which clients connect, also the cluster address if you are using Network Dispatcher

  NetDisp Advisor Port Number

  The TCP port to which Network Dispatchers can connect to retrieve load information

  Keepalive type

  Whether and how the server polls clients to see if they are still active. Possible values are:

  None Server does not poll clients, and will discover client absence only when trying to send data.

  NOP Server polls clients at the TCP level, client software need not have capability to respond.

  Timing mark

  Server polls clients at the TN3270 level, and client software must respond within a certain time window.

  Automatic Logoff

  Whether or not the server disconnects clients after a period of inactivity (with no data flowing in either direction)

- Summary statistics

  Number of connections

  The current number of active TCP connections from TN3270 clients

Number of available Logical Unit Application (LUA) LUs
The number of LUs that have been activated from VTAM or are dynamic LUs and the PU has been activated from VTAM.

Number of defined LUs
The number of LUs defined to a TN3270E server.

Number of LUA LUs pending termination
LUs that have been terminated from 3270 but have not been entirely cleaned up from VTAM.

Number of connections in SSCP-LU state
The number of currently active client TCP connections that have an associated LU in this state (received an ACTLU but not yet a BIND)

Number of connections in LU-LU state
The number of currently active client TCP connections that have an associated LU in this state (received BIND, fully active)

list connections
You can type this command with or without modifiers:

- **list connections**

  Displays all currently active client connections (those with an active TCP connection).
- **list connections** *client ip address*

  Displays all the currently active connections that originated from the specified IP address.
- **list connections** *resource name*

  Displays all currently active connections that are associated with the specified LU or pool name.

For each of the **list connection** commands, the following information is displayed for each session:

Local LU
The LU name, configured at Network Utility, to which the server function has mapped this client TCP connection

Class The type of LU, as follows:

IW Implicit workstation

EW Explicit workstation

IP Implicit printer

EP Explicit printer

Assoc LU
For a workstation LU, the name of any associated printer LU

Client Addr
The IP address of the client

Status
Whether the connection is in SSCP-LU state or LU-LU state

Prim LU
The primary LU name as known to VTAM

Sec LU
The secondary LU name as known to VTAM

Idle Min
The number of minutes since this connection carried any user data

list port
Shows the additional TN3270 ports and defined parameters.

list mapping
Lists all the LU name mapping entries.

list pools
Lists all the TN3270E implicit pools.

Besides the above list commands, a TN3270 server user needs to be able to query the status of other APPN or SNA resources on which the function depends. The following APPN monitoring commands are of general use:

aping - to test connectivity to a remote LU

li port - to show interface status

li link - to show the status of logical links

If you are using DLUR for your host connection, the following commands are particularly useful:

li appc - to check the status of the DLUS-DLUR pipe

li local - to show the status of internal PUs used by the TN3270 server function

li dlur - to show the status of DLUR PUs

To review your APPN configuration, move to talk 6 and type **list all**.

Event Logging Support

In general, APPN/TN3270 ELS messages are intended to capture debug and trace information for IBM support personnel. These functions have extensive logging and trace support, but the ELS messages themselves are tightly packed with low-level information.

Normally, you activate APPN/TN3270 tracing and logging under the direction of IBM support personnel. The general procedure is to enable some of a large list of possible traces as part of your APPN configuration. From the Configuration Program, see the APPN Node Services folder. From talk 6, use the **set trace** command. After you activate this configuration change, the output of these traces flows into a trace table in APPN memory, and also to ELS if you have APPN ELS messages active. Should you have a problem that requires activating traces, IBM support will provide detailed procedures to guide you in capturing debug information.

SNA Management Support

APPN generates SNA alerts for a variety of error conditions, and can forward alerts from other SNA devices. This support is described in “SNA Alert Support” on page 94. There are no alerts specific to the TN3270 server function, but alerts that a Network Utility itself generates may relate to SNA resources involved with TN3270.

From a VTAM or NetView/390 operator console, you can control the links, PUs, and LUs involved with TN3270 as described in “NetView/390” on page 99.

SNMP MIB and Trap Support

Network Utility supports an Internet Draft version of both of the forthcoming standard MIBs for TN3270 server function:

TN3270 Base MIB
TN3270 Response Time MIB

Network Utility support for these MIBs includes the ability to:

- View server configuration, status, and statistics
- Set up client groups for response time collection
- View the mapping of LU names from VTAM name to local name to client IP address
- View the mapping of client IP addresses to VTAM LU names
- Collect response time data for current client groups

In addition, Network Utility supports the following IETF MIBs relating to APPN and SNA functions:

RFC 2155, APPN
RFC 2051, APPC
RFC 2232, DLUR
RFC 2238, HPR
RFC 1666, SNA NAU
Internet Draft, Extended Border Node

Network Utility supports the following IBM Enterprise-Specific MIBs relating to APPN functions:

APPN Memory
APPN Accounting
APPN HPR NCL
APPN HPR Route Test
APPN Peripheral Access Node (Branch Extender)

These MIBs provide a comprehensive view of APPN and SNA resources within Network Utility, including those being used for TN3270.

Network Management Application Support

The Nways Manager products discussed in “IBM Nways Manager Products” on page 96 provide specialized statistical support for TN3270 response time monitoring, as well as the ability to view TN3270 server resources. To initiate response time monitoring, you select a group of one or more clients using an IP address and mask. For each group you define, the manager collects response time statistics into predefined time buckets (less than 1 second, 1 to 2 seconds, and so on). Using the collected information, you can view aggregate historical response time by group, or create custom reports that present the data in different graphical formats.

To view TN3270 resources and their status, you use specific panels that combine information from different tables within the base TN3270 MIB. To view APPN and SNA resources in general, you use specific panels that access information from the APPN MIBs. You can also use integrated browser support to view the information in any of these MIBs.

Nways Manager for AIX provides an APPN-level view of the topology of your network. You can discover participating APPN resources, view them, and view their status as color-coded icons. APPN protocol performance and error events (data and graph) are also provided. This application does not represent Branch Extender or Extended Border Node topologies.

TN3270 Server Enhancements

Dynamic Definition of Dependent LUs

You may use Dynamic Definition of Dependent LUs (DDDLU) to avoid duplicate definition of LUs in both VTAM and the TN3270E. DDDLU allows LUs to be defined in one place only—the Network Utility. In VTAM, you only need to define one or more PUs depending on the number of LUs you need. Implementation of DDDLU also eliminates the efforts of definitions and maintenance in VTAM for future LU definition requirements.

When a TN3270E client requests a connection using one of the LUs defined in the router, the router sends a Reply/PSID NMVT command to VTAM. In this command, the local address of the LU and device type information (3270) is sent to VTAM using the SSCP-PU session. VTAM then sees from the PU definition that there is no definition for the LU in question. At this time, VTAM creates the LU definition using the LUGROUP model statement for the parameter values and the LUSEED value for the dynamic name generation for the LU.

LUs which require specific LU names and 3270 printers on specific ports can also be defined under the same switched major node. See this sample below.

Table 17. DDDPU Sample

```
DDDPU          VBUILD TYPE=SWNET
DDPU           PU  ADDR=02,      x
               IDBLK=077,          x
               IDNUM=22160,        x
                 PUTYPE=2,           x
                USSTAB=US327X,       x
                LUGROUP=GROUP1,      x
               LUSEED=DDLU###,         x
               DLOGMOD=D4C32XX3        x
SALE01       LU  LOCADDR=98,           x     1
                 DLOGMOD=D4C32XX3,       x
               LOGAPPL=CICSA
SALEPRT      LU   LOCADDR=99,           x      2
               LOGMODE=SAL3287,
               LOGAPPL=CICSA
```

1. In this sample definition, the LU ’SALE01’ was requested to be on LOCADDR=98 because of specific requirements. Therefore, this specific LU is defined under this ’DDDPU’ to meet the requirements.
2. In this definition, the printer must also be on a specific port. This especially happens for some SNA applications (e.g. CICS application). The application for the sales department needs a printer on port 99, with LOGMODE=SAL3287, and it needs to be connected to application CICSA when it is activated.

The name specified for the LU in TN3270E Server (local LU) does not need to match the name generated by VTAM for the same LU. If the client wants to have a specific LU, instead of just selecting any LU from a pool, it must use the LU name that is specified in TN3270E. The host application, however, works with the LU name that VTAM generates dynamically. These two names are tied to each other via the local address of the LU.

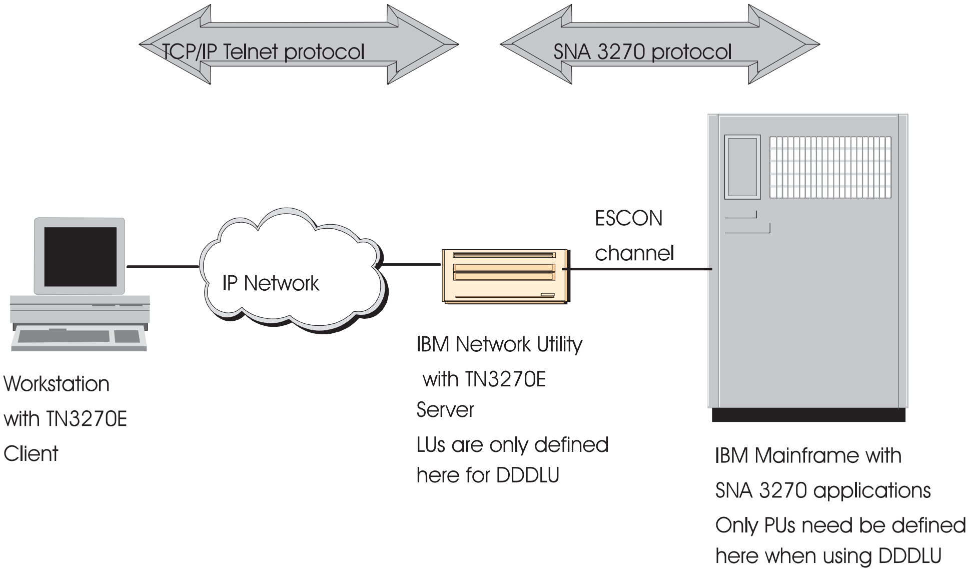

TCP/IP Telnet protocol
SNA 3270 protocol
ESCON channel
IP Network
Workstation with TN3270E Client
IBM Network Utility with TN3270E Server
LUs are only defined here for DDDLU
IBM Mainframe with SNA 3270 applications
Only PUs need be defined here when using DDDLU

Figure 15. TN3270E Server Running on ESCON Attached Network Utility, Using DDDLU

The dynamic definition of LUs is done in VTAM exit routine Selection of Definitions for Dependent LUs (SDDLU). If you use the IBM supplied SDDLU exit program, then you need to specify the LUSEED parameter for name construction in the PU definition, in addition to the LUGROUP model major node. If you use an exit program of your own, you should follow its practices.

This concept is described in detail in SC31-8370, *VTAM Network Implementation Guide*, under the section entitled “Defining Dependent LUs Dynamically”.

The LU can be explicit (as defined locally in TN3270E), in which case the client must specify an exact LU name in the workstation. The LU requested by the user (TN3270 client) can also be an implicit one, in which case it belongs to a pool of LUs.

IP address to LU name mapping is also supported for DDDLUs. In addition, you can have other explicit LUs, defined in different ways, under a different PU to be used for IP address to LU mapping.

TN3270 Host Initiated Dynamic LU Definitions

In addition to DDDLU, another way to avoid duplicate LU definitions is Host Initiated Dynamic LU (HIDLU). HIDLU allows LUs to be defined in VTAM only. In Network Utility (or 2216) you only define a PU, or as many PUs you need, but no LUs for these PUs.

When a client requests to use such an LU, TN3270E sends VTAM a request to activate the PU and its LUs. When the VTAM-defined LUs are activated, the LU names will be conveyed to Network Utility in the ACTLU commands in Control Vector 0E.

LUs defined in this manner have the same name in both VTAM and the Network Utility.

To use HIDLU, you have to use parameter **INCLUD0E=YES** in the PU definition in VTAM. This function requires VTAM V4R4 with APAR OW25501 and OW31805. With HIDLU, you can only define display terminals. Printers are not supported. HIDLU definitions can be used at the same time with other locally (in the Network Utility) defined LUs, which can be implicit, explicit or DDDLU defined LUs.

TN3270 Host On-Demand Client Caching

Host On-Demand (HOD) allows Web browser clients to connect to SNA 3270 and 5250 host applications. The terminal emulation (TN3270 or TN5250) runs as a Java applet in the client's browser. Connection to a host application is made via a TN3270 (or TN5250) server.

Java applets are usually retrieved from a HOD server, which runs as a Web server.

Host On-Demand Client Caching allows an IBM 2216 or 2212 or Network Utility, acting as a TN3270 Server, to cache the HOD applets and serve them to client browsers upon request.

HOD Client Cache can offload the HOD Server and, if placed strategically, can load HOD pages and applets quicker to client workstations. Another advantage of using HOD Client Cache function is to distribute the load on specific lines/bandwidths within the network to eliminate the congestion. The function uses and is defined under Network Dispatcher feature, using either talk 6 or the configuration program. First, a cluster address is defined to Network Dispatcher, and then port number(s) and HOD Server addresses are defined under that cluster address.

The basic operating principle of HOD Client Cache is the following: The clients use the ND cluster address in their browsers, instead of the actual address of the HOD Server. When the request for HOD Server comes to the cluster address, port 80 (HTTP port number), Java applets required to establish the session will be transferred from the HOD cache. If the applets or other required pages are not in the cache of Network Utility, the router will connect to the HOD Server, download the items, store them in the cache and provide them to the client. Now that the pages and applets are in the cache, the next user will make a hit and get them directly from the cache. Therefore, this HOD Client Cache function on Network Utility helps to better utilize the network by distributing the Java applets on Network Utilities so that clients do not have to load these applets from the HOD Server. When using this function, no extra load is created on the HOD Server either, since the requests of Java applets are delivered by Network Utility.

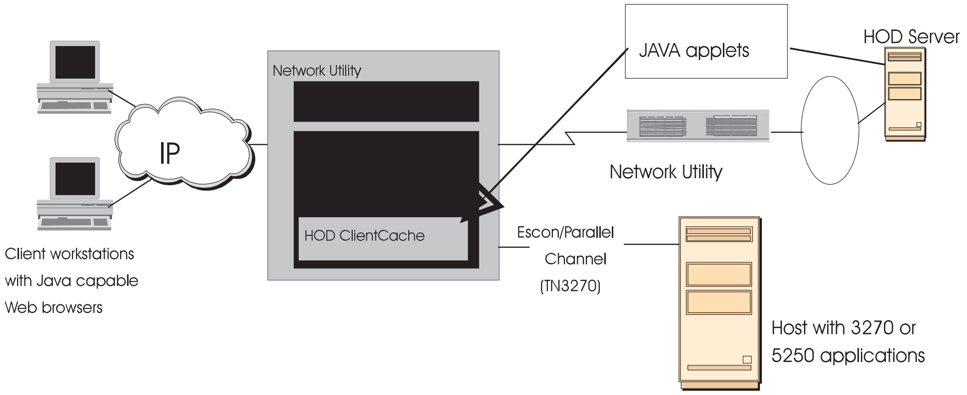


Figure 16. Scenario With TN3270E Server and HOD Cache

HOD Client Caching is only available together with TN3270E Server function.

Chapter 13. TN3270E Server Example Configuration Details

This chapter contains diagrams and configuration parameter tables for several of the examples of TN3270E server network configurations in "Chapter 12. TN3270E Server" on page 127. The parameter values shown are from real working test configurations.

For an explanation of the columns and conventions in the configuration parameter tables, see "Example Configuration Table Conventions" on page 125.

The Network Utility World Wide Web pages contain binary configuration files that match these configuration parameter tables. To access these files, follow the `Download` link from:

```
http://www.networking.ibm.com/networkutility
```

The configurations documented in this chapter are:

Table 18. Cross-Reference of Example Configuration Information

Configuration Description	Parameter Table
"TN3270 via a Subarea Connection to an NCP" on page 132	Table 19 on page 152
"Highly Scalable, Fault-Tolerant TN3270E" on page 136, for the TN3270 server TN A	Table 20 on page 158
"Highly Scalable, Fault-Tolerant TN3270E" on page 136, for the Network Dispatcher ND A	Table 21 on page 163
"TN3270 Via DLUR over APPN" on page 139	Table 23 on page 169
"Dynamic Definition of Dependent LUs" on page 146	"Dynamic Definition of Dependent LUs" on page 172
"TN3270 Host Initiated Dynamic LU Definitions" on page 147	"Host Initiated Dynamic LU Definition" on page 179
"TN3270 Host On-Demand Client Caching" on page 148	"TN3270E Host On-Demand (HOD) Client Cache" on page 185
"TN3270 Subarea SNA over DLSw" on page 135	"TN3270E Subarea SNA over DLSw" on page 192

TN3270 via LAN Subarea, via DLUR, and using Network Dispatcher

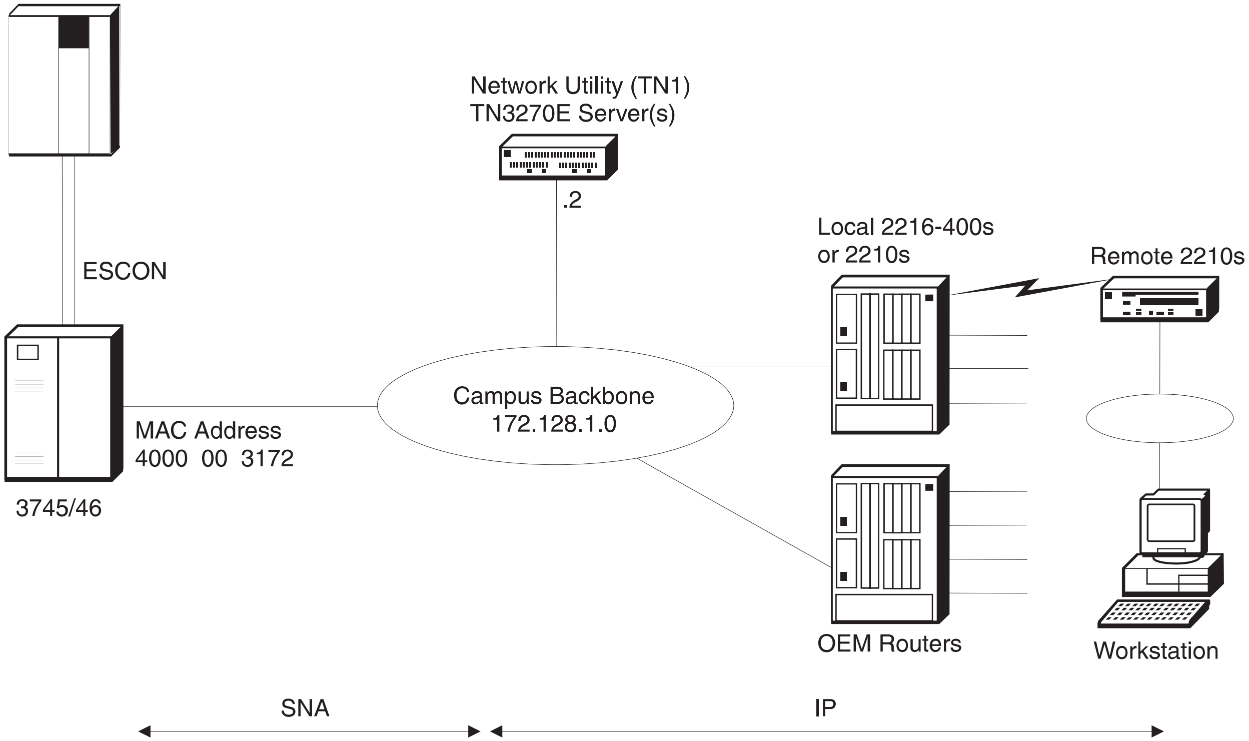


Figure 17. TN3270E Subarea

Table 19. TN3270E Subarea. See page 132 for a description and 152 for a diagram of this configuration.

Configuration Program Navigation	**Configuration Program Values**	**Command-Line Commands**	**Notes**
Devices Adapters Slots	Slot1: 2-Port TR	See ″add dev″ on next row	1
Devices Adapters Ports	Slot 1/Port 1: Interface 0: TR	`Config>`**`add dev tok`**	2

Devices
Interfaces
	Interface 0
	MAC Address 400022AA0001
		`Config>`**`net 0`**
		`TKR Config>`**`set phy 40:00:22:AA:00:01`**	

Table 19. TN3270E Subarea (continued). See page 132 for a description and 152 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes
System General	System name: NU_A Location: XYZ Contact: Administrator	Config>**set host** Config>**set location** Config>**set contact**	
System SNMP Config General	SNMP (checked)	Config>**p snmp** SNMP Config>**enable snmp**	
System SNMP Config Communities General	Community name: admin Access type: Read-write trap Community view: All	SNMP Config>**add community** SNMP Config>**set comm access write**	3
Protocols IP General	Internal address: 172.128.252.2 Router ID: 172.128.1.2	Config>**p ip** IP Config>**set internal 172.128.252.2** IP Config>**set router-id 172.128.1.2**	
Protocols IP Interfaces	Interface 0 (TR slot 1 port 1) IP address: 172.128.1.2 Subnet mask: 255.255.255.0	IP Config>**add address**	
Protocols IP OSPF General	OSPF (checked)	Config>**p ospf** OSPF Config>**enable ospf** (Accept other defaults)	
Protocols IP OSPF Area Configuration General	Area number: 0.0.0.0 Stub area (not checked)	OSPF Config>**set area**	
Protocols IP OSPF Interfaces	Interface 0 OSPF (checked)	OSPF Config>**set interface** Interface IP address: **172.128.1.2** Attaches to area: **0.0.0.0** (Accept other defaults)	
Protocols APPN General	APPN network node (checked to enable) Network ID: NUBNODE Control point name: CPNU	Config>**p appn** APPN config> **set node** Enable APPN Network ID: **NUBNODE** Control point name: **CPNU** (Accept other defaults)	4

Table 19. TN3270E Subarea (continued). See page 132 for a description and 152 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
APPN
Interfaces
	(highlight Interface 0 Token Ring)
	(click on the configure tab)
	Define APPN port (checked to enable)
	Port name: TR3270
	High performance routing (HPR) supported (unchecked to disable)
	Support multiple PUs (checked to enable)
		`APPN config>`**`add port`**
		`APPN Port Link Type:` **`TOKEN RING`**
		`Port name:` **`TR3270`**
		`Enable APPN`
		`Support multiple PUs`
		`High performance routing:` **`No`**
		(Accept other defaults)
			5

Protocols
APPN
Interfaces
	(highlight Interface 0 Token Ring)
	(click on the Link stations tab)
	STAT001 (new definition)
	General-1 Tab:
	Link station name: STAT001
	Solicit SSCP session (checked)
	Link support APPN functions (unchecked)
	General-2 Tab:
	MAC address of adjacent node: 400000003172
	Node ID: 12244
	Local SAP address: 04
	(click on **Add** to create the Link station)
	
	STAT002 (new definition)
	General-1 Tab:
	Link station name: STAT002
	Solicit SSCP session (checked)
	Link support APPN functions (unchecked)
	General-2 Tab:
	MAC address of adjacent node: 400000003172
	Node ID: 12245
	Local SAP address: 08
	(click on **Add** to create the Link station)
		`APPN config>`**`add link`**
		`Port name for the link station:` **`TR3270`**
		`Station name:` **`STAT001`**
		`MAC address of adjacent node:` **`400000003172`**
		`Solicit SSCP Session:` **`Yes`**
		`Local Node ID:` **`12244`**
		`Local SAP address:` **`4`**
		`Does link support APPN function?:` **`No`**
		(Accept other defaults)
		`APPN config>`**`add link`**
		`Port name for the link station:` **`TR3270`**
		`Station name:` **`STAT002`**
		`MAC address of adjacent node:`**`400000003172`**
		`Solicit SSCP Session:` **`Yes`**
		`Local Node ID:` **`12245`**
		`Local SAP address:` **`8`**
		`Does link support APPN function?:` **`No`**
		(Accept other defaults)
			6

Table 19. TN3270E Subarea (continued). See page 132 for a description and 152 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
  APPN
    TN3270E Server
      General
	TN3270E (checked to enable)
	  IP address : 172.128.1.2
	  Automatic logoff (checked to enable)
		`APPN config>`**`tn`**
		`TN3270E config>`**`set`**
		`  Enable TN3270E Server`
		`  TN3270E Server IP Address:` **`172.128.1.2`**
		`  Automatic logoff:` **`Yes`**
		    (Accept other defaults)
			7

Protocols
  APPN
    TN3270E Server
      LUs
	Local PU Name: STAT001
	    (click on **Implicit Pool**)
	      LU name mask: @LU1A
	      Number of implicit workstation definitions: 10
	Local PU Name: STAT002
	    (click on **Implicit Pool**)
	      LU name mask: @LU2A
	      Number of implicit workstation definitions: 10
	    (click on **LUs** to define explicit LUs)
	      LU name: PC03A
	      NAU address: 5
	    (click on **Add**)
		`TN3270E config>`**`add imp`**
		`  Station Name:` **`STAT001`**
		`  LU name mask:` **`@LU1A`**
		`  Number of Implicit LUs in Pool:` **`10`**
		
		`TN3270E config>`**`add imp`**
		`  Station Name:` **`STAT002`**
		`  LU name mask:` **`@LU2A`**
		`  Number of Implicit LUs in Pool:` **`10`**
		
		`TN3270E config>`**`add lu`**
		`  Station Name:` **`STAT002`**
		`  LU name:` **`PC03A`**
		`  NAU address:` **`5`**
			8

Notes:

1. **add dev** defines a single port, not an adapter.
2. The configuration program assigns an interface number to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the **add dev** command for each port you want to use, and the interface number (also known as ″net number″) is the output of the command.
3. You need a write-capable SNMP community only if you want to download configuration files from the configuration program directly into the router. SNMP is not required to TFTP a configuration file to the router.
4. If you have a pure SNA subarea network with no APPN, then the Network ID can be any value. If you have APPN in your network, then the Network ID should conform to your APPN network naming conventions.
5. APPN must be enabled even though this example uses SNA subarea for the TN3270E server connection to the host. This is because the TN3270E server code uses the APPN SNA stack both for APPN and subarea communications to the host.
6. When you create the link stations, you are also implicitly creating PUs. These PUs are assigned a ″Local Node ID″ here. This must match the ″IDNUM″ in VTAM's SW Major Node definition. The ID Block is always 077 for a Network Utility. If you need to define multiple link stations (PUs), then each link station has to have a different Local SAP address.

   Setting Solicit SSCP session to yes defines the link as a subarea connection.
7. Beginning with MAS V3.2, TN3270E Server has its own command-line submenu.
8. For implicit LUs, you just have to define the pools. The @LU1A is a template that will be used to create the actual LU names in the pool. In this example, with 10 LUs in the pool, the LU names generated are @LU1A2, @LU1A3, @LU1A4,...@LU1A11 which correspond to LOCADDRs 2-11 for the PU defined in VTAM. Similarly, @LU2A will generate @LU2A2, @LU2A3, @LU2A4. Note that the LU name @LU2A5 is not used because the NAU address of 5 has been reserved for the explicit definition. Therefore, the remaining LUs in the pool are @LU2A6 through @LU2A12.

   For explicit LUs, the LU name given here must match the name defined in the workstation's 3270 emulation configuration. The NAU address points to the LOCADDR in the appropriate PU definition in the Switched Major node in VTAM.

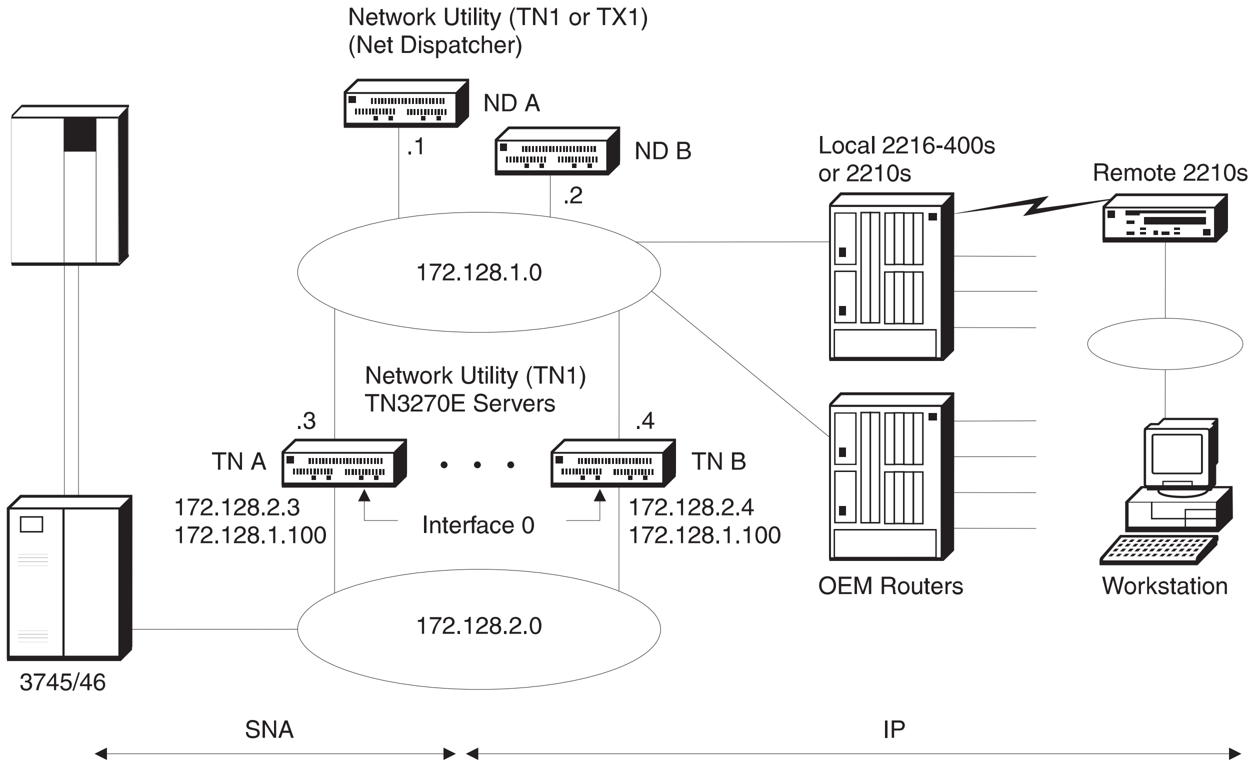


Figure 18. TN3270E Server Config -Highly Scalable, Fault-Tolerant TN3270

Table 20. TN3270E Server Config -Highly Scalable, Fault-Tolerant TN3270. See page 136 for a description and 157 for a diagram of this configuration.

This table provides the configuration for the TN A server. *See Table 21 on page 163 and Table 22 on page 167 for the configuration of the Network Dispatchers for this example.*

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes
Devices Adapters Slots	Slot1: 2-Port TR	See ″add dev″ on next row	1
Devices Adapters Ports	Slot 1/Port 1: Interface 0: TR Slot 1/Port 2: Interface 1: TR	Config>**add dev tok** (once for each interface)	2
Devices Interfaces	Interface 0 Mac Address 400022AA0053 Interface 1 Mac Address 400022AA0003	Config>**net 0** TKR config>**set phy 40:00:22:AA:00:53** TKR config>**exit** Config>**net 1** TKR config>**set phy 40:00:22:AA:00:03**	
System General	System name: NU_A Location: XYZ Contact: Administrator	Config>**set host** Config>**set location** Config>**set contact**	
System SNMP Config General	SNMP (checked)	Config>**p snmp** SNMP Config>**enable snmp**	
System SNMP Config Communities General	Community name: admin Access type: Read-write trap Community view: All	SNMP Config>**add community** SNMP Config>**set comm access write**	3
Protocols IP General	Internal address: 172.128.252.3 Router ID: 172.128.1.3 Same Subnet (checked)	Config>**p ip** IP config>**set internal 172.128.252.3** IP config>**set router-id 172.128.1.3** IP config>**enable same-subnet**	4

Table 20. TN3270E Server Config -Highly Scalable, Fault-Tolerant TN3270 (continued). See page 136 for a description and 157 for a diagram of this configuration.

This table provides the configuration for the TN A server. *See Table 21 on page 163 and Table 22 on page 167 for the configuration of the Network Dispatchers for this example.*

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
IP
Interfaces
	Interface 0 (TR slot 1 port 1)
	IP address: 172.128.2.3
	Subnet mask: 255.255.255.0
	IP address: 172.128.1.100
	Subnet mask: 255.255.255.0
	Interface 1 (TR slot 1 port 2)
	IP address: 172.128.1.3
	Subnet mask: 255.255.255.0
		`IP config>`**`add address 0 172.128.2.3 255.255.255.0`**
		`IP config>`**`add address 0 172.128.1.100 255.255.255.0`**
		`IP config>`**`add address 1 172.128.1.3 255.255.255.0`**
			5,6

Protocols
IP
OSPF
General
	OSPF (checked)	`Config>`**`p ospf`**
		`OSPF Config>`**`enable ospf`**	

Protocols IP OSPF Area Configuration General	Area number: 0.0.0.0 Stub area (not checked)	`OSPF Config>`**`set area`**	

Protocols
IP
OSPF
Interfaces
	Interface 1
	OSPF (checked)
		`OSPF Config>`**`set interface`**
		`Interface IP address:` **`172.128.1.3`**
		`Attaches to area:` **`0.0.0.0`**
		(Accept other defaults)
			7

Protocols
APPN
General
	APPN network node (checked to enable)
	Network ID: NUBNODE
	Control point name: CPNU
		`Config>`**`p appn`**
		`APPN config>` **`set node`**
		`Enable APPN`
		`Network ID:` **`NUBNODE`**
		`Control point name:` **`CPNU`**
		(Accept other defaults)
			8

Table 20. TN3270E Server Config -Highly Scalable, Fault-Tolerant TN3270 (continued). See page 136 for a description and 157 for a diagram of this configuration.

This table provides the configuration for the TN A server. *See Table 21 on page 163 and Table 22 on page 167 for the configuration of the Network Dispatchers for this example.*

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
APPN
Interfaces
	(highlight Interface 0 Token Ring)
	(click on the **configure** tab)
	Define APPN port (checked to enable)
	Port name: TR3270
	High performance routing (HPR) supported (unchecked to disable)
	Support multiple PUs (checked to enable)
		`APPN config>`**`add port`**
		`APPN Port Link Type:` **`TOKEN RING`**
		`Port name:` **`TR3270`**
		`Enable APPN`
		`Support multiple PUs`
		`High performance routing:` **`No`**
		(Accept other defaults)
			9

Protocols
APPN
Interfaces
	(highlight Interface 0 Token Ring)
	(click on the **Link stations** tab)
	STAT001 (new definition)
	General-1 Tab:
	Link station name: STAT001
	Solicit SSCP session (checked)
	Link support APPN functions (unchecked)
	General-2 Tab:
	MAC address of adjacent node: 400000003172
	Node ID: 12244
	Local SAP address: 04
	(click on **Add** to create the Link station)
	
	STAT002 (new definition)
	General-1 Tab:
	Link station name: STAT002
	Solicit SSCP session (checked)
	Link support APPN functions (unchecked)
	General-2 Tab:
	MAC address of adjacent node: 400000003172
	Node ID: 12245
	Local SAP address: 08
	(click on **Add** to create the Link station)
		`APPN config>`**`add link`**
		`Port name for the link station:` **`TR3270`**
		`Station name:` **`STAT001`**
		`MAC address of adjacent node:` **`400000003172`**
		`Solicit SSCP Session:` **`Yes`**
		`Local Node ID:` **`12244`**
		`Local SAP address:` **`4`**
		`Does link support APPN function?:` **`No`**
		
		(Accept other defaults)
		
		`APPN config>`**`add link`**
		`Port name for the link station:` **`TR3270`**
		`Station name:` **`STAT002`**
		`MAC address of adjacent node:`**`400000003172`**
		`Solicit SSCP Session:` **`Yes`**
		`Local Node ID:` **`12245`**
		`Local SAP address:` **`8`**
		`Does link support APPN function?:` **`No`**
		(Accept other defaults)
			10

Table 20. TN3270E Server Config -Highly Scalable, Fault-Tolerant TN3270 (continued). See page 136 for a description and 157 for a diagram of this configuration.

This table provides the configuration for the TN A server. *See Table 21 on page 163 and Table 22 on page 167 for the configuration of the Network Dispatchers for this example.*

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
APPN
TN3270E Server
General
	TN3270E (checked to enable)
	IP address : 172.128.1.100
	Automatic logoff (checked to enable)
		`APPN config>`**`tn`**
		`TN3270E config>`**`set`**
		`Enable TN3270E Server`
		`TN3270E Server IP Address:` **`172.128.1.100`**
		`Automatic logoff:` **`Yes`**
		(Accept other defaults)
			11

Protocols
APPN
TN3270E Server
LUs
	Local PU Name: STAT001
	(click on **Implicit Pool**)
	LU name mask: @LU1A
	Number of implicit workstation definitions: 10
	Local PU Name: STAT002
	(click on **Implicit Pool**)
	LU name mask: @LU2A
	Number of implicit workstation definitions: 10
		`TN3270E config>`**`add imp`**
		`Station Name:` **`STAT001`**
		`LU name mask:` **`@LU1A`**
		`Number of Implicit LUs in Pool:` **`10`**
		
		`TN3270E config>`**`add imp`**
		`Station Name:` **`STAT002`**
		`LU name mask:` **`@LU2A`**
		`Number of Implicit LUs in Pool:` **`10`**
			12

Notes:

1. **add dev** defines a single port, not an adapter.
2. The configuration program assigns an interface number to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the **add dev** command for each port you want to use, and the interface number (also known as ″net number″) is the output of the command.
3. You need a write-capable SNMP community only if you want to download configuration files from the configuration program directly into the router. SNMP is not required to TFTP a configuration file to the router.
4. You must enable the ″same-subnet function″ because you are using two interfaces with an IP address within the same subnet. (172.128.1.3 is assigned to TR 1 and 172.128.1.100 (cluster address) is assigned as a 2nd address to TR 0.)
5. Note that Interface 0 has been assigned two IP addresses, one of which is the cluster address used by the Network dispatcher. The TN3270E Server will be configured for the same address in a subsequent step. All TN3270 traffic will be sent to this address through the Network Dispatcher. In order for this traffic to reach the Network Utility's internal IP queue, this address needs to be assigned to either an interface address or the internal address. In this example, it has been assigned to an interface as the second address of that interface.
6. Note that Interface 0 is on the LAN segment which is connected to the SNA gateway. This segment carries the LLC traffic from the TN3270 server to the gateway. Depending on the rest of the configuration of your Network Utility, this segment may not have any IP traffic on it. However, since all the TN3270E servers will have the same IP address assigned to the interface on this segment, it has been assigned a subnet address (172.128.2) and all the TN3270E servers will have an address on this subnet also (in this case 172.128.2.3) in order to an IP addressing conflict.
7. It is very important ***not*** to enable OSPF on the Network Dispatcher cluster address. If you do, the cluster address will be broadcast to the network as being on the TN3270E server (in addition to the Network Dispatcher machine).
8. If you have a pure SNA subarea network with no APPN, then the Network ID can be any value. If you have APPN in your network, then the Network ID should conform to your APPN network naming conventions.
9. APPN must be enabled even though the example uses SNA subarea for the TN3270E server connection to the host. This is because the TN3270E server code uses the APPN SNA stack both for APPN and subarea communications to the host.
10. When you create the link stations, you are also implicitly creating PUs. These PUs are assigned a ″Local Node ID″ here. This must match the ″IDNUM″ in VTAM's SW Major Node definition. The ID Block is always 077 for a Network Utility. If you need to define multiple link stations (PUs), then each link station has to have a different Local SAP address.
11. Beginning with MAS V3.2, TN3270E Server has its own command-line submenu.
12. For implicit LUs, you just have to define the pools. The `@LU1A` is a template that will be used to create the actual LU names in the pool. In this example, with 10 LUs in the pool, the LU names generated are `@LU1A2, @LU1A3,...@LU1A11` which correspond to LOCADDRs 2-11 for the PU defined in VTAM. Similarly, `@LU2A` will generate `@LU2A2, @LU2A3, ... @LU2A11`.

Table 21. Network Dispatcher Config -Highly Scalable, Fault-Tolerant TN3270. See page 136 for a description and 157 for a diagram of this configuration.

This table provides the configuration for the Primary Network Dispatcher, ND A. *See Table 22 on page 167 for the configuration of the backup Network Dispatcher. See Table 19 on page 152 for the configuration of the TN3270E servers for this example.*

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes
Devices Adapters Slots	Slot 1: 2-Port TR	See ″add device″ on next row	1
Devices Adapters Ports	Slot 1/Port 1: Interface 0: TR	`Config>`**`add dev tok`**	2

Devices
Interfaces
	Interface 0
	MAC address: 400022AA0001
		`Config>`**`net 0`**
		`TKR config>`**`set phy 40:00:22:AA:00:01`**	

System
General
	System name: NU_A
	Location: XYZ
	Contact: Administrator
		`Config>`**`set host`**
		`Config>`**`set location`**
		`Config>`**`set contact`**	

System
SNMP Config
General
	SNMP (checked)	`Config>`**`p snmp`**
		`SNMP Config>`**`enable snmp`**	

System
SNMP Config
Communities
General
	Community name: admin
	Access type: Read-write trap
	Community view: All
		`SNMP Config>`**`add community`**
		`SNMP Config>`**`set comm access write`**
			3

Protocols
IP
General
	Internal address: 172.128.252.1
	Router ID: 172.128.1.1
		`Config>`**`p ip`**
		`IP config>`**`set internal 172.128.252.1`**
		`IP config>`**`set router-id 172.128.1.1`**
			4

Protocols IP Interfaces	Interface 0 (TR slot 1 port 1) IP address: 172.128.1.1 Subnet mask: 255.255.255.0	`IP config>`**`add address`**	

Protocols
IP
OSPF
General
	OSPF (checked)	`Config>`**`p ospf`**
		`OSPF Config>`**`enable ospf`**	

Protocols IP OSPF Area Configuration General	Area number: 0.0.0.0 Stub area (not checked)	`OSPF Config>`**`set area`**	

Table 21. Network Dispatcher Config -Highly Scalable, Fault-Tolerant TN3270 (continued). See page 136 for a description and 157 for a diagram of this configuration.

This table provides the configuration for the Primary Network Dispatcher, ND A. *See Table 22 on page 167 for the configuration of the backup Network Dispatcher. See Table 19 on page 152 for the configuration of the TN3270E servers for this example.*

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes
Protocols IP OSPF Interfaces	Interface 0 OSPF (checked)	OSPF Config>**set interface** Interface IP address **172.128.1.1** Attaches to area **0.0.0.0** (Accept other defaults)	
Features Network Dispatcher Router Executor	Executor (checked)	Config>**feat ndr** NDR Config>**enable executor**	
Features Network Dispatcher Router Clusters Detail	Cluster address: 172.128.1.100	NDR Config>**add cluster** Cluster Address: **172.128.1.100** (Accept other defaults)	
Features Network Dispatcher Router Clusters Ports	Port Number 23	NDR Config>**add port** Cluster Address **172.128.1.100** Port number **23** (Accept other defaults)	
Features Network Dispatcher Router Clusters Servers	Server address: 172.128.1.3 172.128.1.4	NDR Config>**add server** Cluster Address: **172.128.1.100** Port number: **23** Server Address: **172.128.1.3** (Accept other defaults) (Repeat for 172.128.1.4)	
Features Network Dispatcher Router Manager	Manager (checked) Proportion Active: 10 New: 10 Advisor: 80 System: 0	NDR Config>**enable manager** NDR Config>**set manager propor** Active: **10** New: **10** Advisor: **80** System: **0** (Accept other defaults)	5

Table 21. Network Dispatcher Config -Highly Scalable, Fault-Tolerant TN3270 (continued). See page 136 for a description and 157 for a diagram of this configuration.

This table provides the configuration for the Primary Network Dispatcher, ND A. *See Table 22 on page 167 for the configuration of the backup Network Dispatcher. See Table 19 on page 152 for the configuration of the TN3270E servers for this example.*

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes
Features Network Dispatcher Router Advisors	Advisor (checked) Advisor name: TN3270 Advisor port: 23 Timeout: 10	NDR Config>**add advisor** Advisor name: **3** (for TN3270) Timeout: **10** (Accept other defaults) NDR Config>**enable advisor** Advisor name: **3** (for TN3270) Port number: **23**	6
Features Network Dispatcher Router Backup	Backup (checked to enable) Backup role: PRIMARY Switch back Strategy: MANUAL	NDR Config>**add backup** Role: **0**=PRIMARY Switch back strategy: **1**=MANUAL	7
Features Network Dispatcher Router Reaches	Reach address: (Enter each address and click on **Add**) 172.128.1.3 172.128.1.4	NDR Config>**add reach** Address to reach: **172.128.1.3** (Repeat for 172.128.1.4)	8
Features Network Dispatcher Router Heart Beats	Source address: 172.128.1.1 Target address: 172.128.1.2 (Enter addresses and click on **Add**)	NDR Config>**add heartbeat** Source Heartbeat address: **172.128.1.1** Target Heartbeat Address: **172.128.1.2**	8

Notes:

1. **add dev** defines a single port, not an adapter.
2. The configuration program assigns an interface number to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the **add dev** command for each port you want to use, and the interface number (also known as ″net number″) is the output of the command.
3. You need a write-capable SNMP community only if you want to download configuration files from the configuration program directly into the router. SNMP is not required to TFTP a configuration file to the router.
4. The internal address must be set in order for the advisor and the manager functions to communicate with the executor component of network dispatcher.
5. The values for Active, New, Advisor, and System must add up to 100. The Advisor proportion defaults to 0. You need to change this so that the Advisor input can be used to load balance the TN3270 traffic. In this case, it has been set to `80` to give it the a much greater weight than those for active and new connections.
6. The communication port number (defaults to 10008) must match the server's ″Network Dispatcher advisor port.″
7. Switchback Strategy must be the same for both primary and backup network dispatchers. IBM recommends a manual setting so that you can schedule the switchback at a time when you have the least probability of disrupting your SNA sessions.
8. The reach addresses are the addresses that the Network Dispatcher must be able to reach in order for it to determine that it is functioning correctly. The primary sends this information at regular intervals to the backup. If the backup determines that it has better reachability than the primary, then it will perform a switchover and assume the primary role. Choose at least one host on each subnet that the Network Dispatcher uses. Also, add the addresses for each server in the cluster. In this example, the Network Dispatcher uses only one interface and both servers are on the same subnet as this interface.
9. Here, you are configuring the connection that the primary Network dispatcher will use to send the heart beats to the backup Network Dispatcher. You can define several paths if you have multiple connections between the primary and backup. The heartbeats will be sent over the first path that is available. The most robust solution is to configure a second path between the primary network dispatcher and the backup network dispatcher using the second slot that is available in each Network Utility.

Table 22. Network Dispatcher Config -Highly Scalable, Fault-Tolerant TN3270. See page 136 for a description and 157 for a diagram of this configuration.

This table provides the configuration differences for the backup Network Dispatcher ND B based on Table 21 on page 163, which gives the configuration for the primary Network Dispatcher. *The definition for the backup Network Dispatcher is the same as for the Primary except for the differences that are shown in this table. These differences correspond to the interface addresses and the Network Dispatcher backup functions. The parameters related to the Network Dispatcher that are not shown here must be identical to the values configured on the primary. It is also recommended that the hardware configuration be the same for both the primary and backup Network Dispatchers. See Table 20 on page 158 for the configuration of the TN3270E servers for this example.*

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes
Devices Interfaces	Interface 0 MAC Address 400022AA0002	Config>**net 0** TKR config>**set phy 40:00:22:AA:00:02**	
System General	System name: NU_ND2	Config>**set host**	
Protocols IP General	Internal address: 172.128.252.2 Router ID: 172.128.1.2	Config>**p ip** IP config> **set internal 172.128.252.2** **set router-id 172.128.1.2**	1
Protocols IP Interfaces	Interface 0 (TR slot 1 port 1) IP address: 172.128.1.2 Subnet mask: 255.255.255.0	Config>**p ip** IP config>**add address**	
Protocols IP OSPF Interfaces	Interface 0 OSPF (checked)	Config>**p ospf** OSPF Config>**set interface** Interface IP address: **172.128.1.2** Attaches to area: **0.0.0.0** (accept other defaults)	
Features Network Dispatcher Router Backup	Backup (checked to enable) Backup role: BACKUP Switch back Strategy: MANUAL	Config>**feat NDR** NDR Config>**add backup** Role: **1**=BACKUP Switch back strategy: **1**=MANUAL	
Features Network Dispatcher Router Heart Beats	Source address: 172.128.1.2 Target address: 172.128.1.1 (Enter addresses and click on **Add**)	NDR Config>**add heartbeat** Source Heartbeat address: **172.128.1.2** Target Heartbeat Address: **172.128.1.1**	2

Notes:

1. The internal address must be set in order for the advisor and the manager functions to communicate with the executor component of network dispatcher.
2. The backup must be configured with all the same information as the primary network dispatcher so that if the primary fails, the backup can assume the full role of primary including the sending of the heartbeats and the reachability information to the primary when it comes back online.

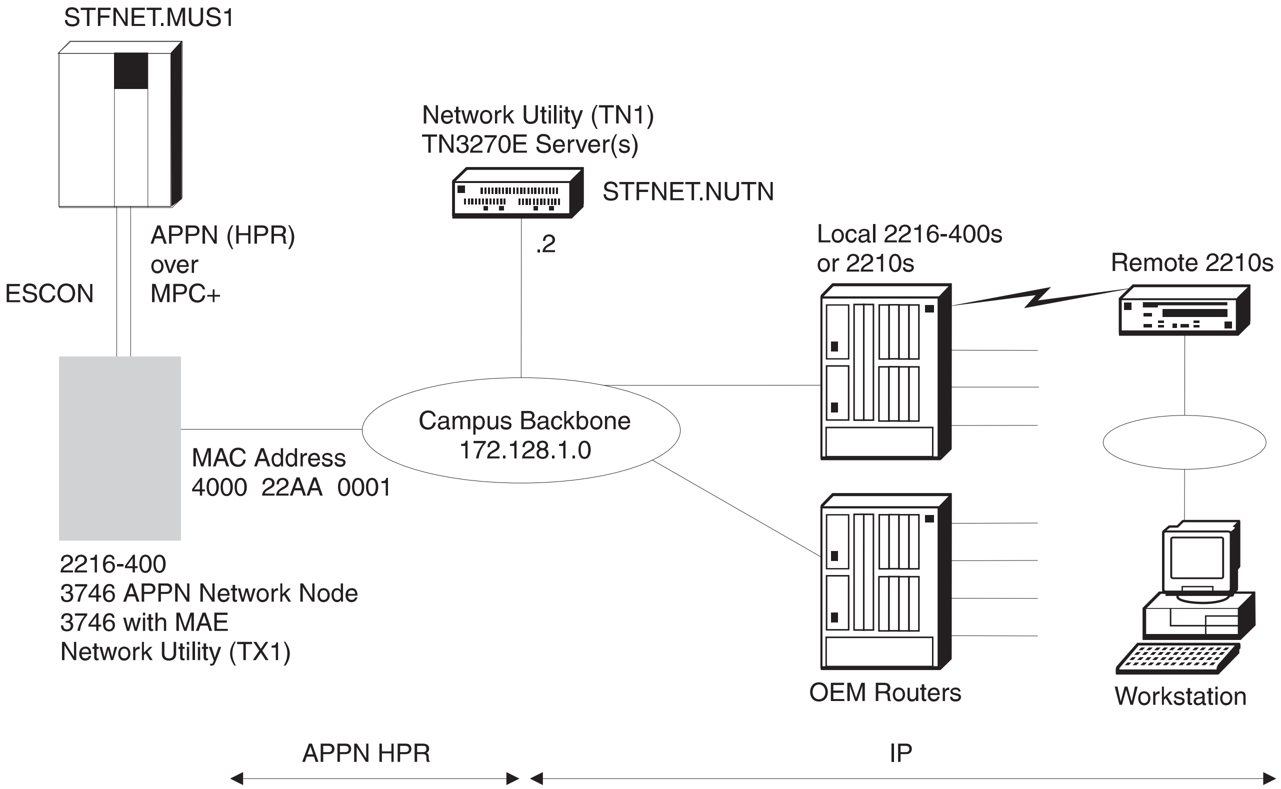


Figure 19. TN3270 via DLUR over APPN

Table 23. TN3270 via DLUR over APPN. See page 139 for a description and 168 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes
Devices Adapters Slots	Slot1: 2-Port TR	See "add dev" on next row	1
Devices Adapters Ports	Slot 1/Port 1: Interface 0: TR	Config>**add dev tok**	2
Devices Interfaces	Interface 0 Mac Address 400022AA0011	Config>**net 0** TKR config>**set phy 40:00:22:AA:00:11**	
System General	System name: NU_A Location: XYZ Contact: Administrator	Config>**set host** Config>**set location** Config>**set contact**	
System SNMP Config General	SNMP (checked)	Config>**p snmp** SNMP Config>**enable snmp**	
System SNMP Config Communities General	Community name: admin Access type: Read-write trap Community view: All	SNMP Config>**add community** SNMP Config>**set comm access write**	3
Protocols IP General	Internal address: 172.128.252.2 Router ID: 172.128.1.2	Config>**p ip** IP config>**set internal 172.128.252.2** IP config>**set router-id 172.128.1.2**	
Protocols IP Interfaces	Interface 0 (TR slot 1 port 1) IP address: 172.128.1.2 Subnet mask: 255.255.255.0	IP config>**add address**	
Protocols IP OSPF General	OSPF (checked)	Config>**p ospf** OSPF Config>**enable ospf**	
Protocols IP OSPF Area Configuration General	Area number: 0.0.0.0 Stub area (not checked)	OSPF Config>**set area**	

Table 23. TN3270 via DLUR over APPN (continued). See page 139 for a description and 168 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
IP
OSPF
Interfaces
	Interface 0
	OSPF (checked)
		`OSPF Config>`**`set interface`**
		`Interface IP address:` **`172.128.1.2`**
		`Attaches to area:` **`0.0.0.0`**
		(Accept other defaults)	

Protocols
APPN
General
	APPN network node (checked to enable)
	Network ID: STFNET
	Control point name: NUTN
		`Config>`**`p appn`**
		`APPN config>` **`set node`**
		`Enable APPN`
		`Network ID:` **`STFNET`**
		`Control point name:` **`NUTN`**
		(Accept other defaults)	

Protocols
APPN
Interfaces
	(highlight Interface 0 Token Ring)
	(click on the configure tab)
	Define APPN port (checked to enable)
	Port name: TR001
		`APPN config>`**`add port`**
		`APPN Port Link Type:` **`TOKEN RING`**
		`Port name:` **`TR001`**
		`Enable APPN`
		(Accept other defaults)
			4

Protocols
APPN
Interfaces
	(highlight Interface 0 Token Ring)
	(click on the Link stations tab)
	TRTG001 (new definition)
	General-1 Tab:
	Link station name: TRTG001
	General-2 Tab:
	MAC address of adjacent node:
	400022AA0001
	Adjacent Node Type:
	APPN Network Node
	(click on **Add** to create the Link station)
		`APPN config>`**`add link`**
		`Port name for the link station:` **`TR001`**
		`Station name:` **`TRTG001`**
		`MAC address of adjacent node:` **`400022AA0001`**
		(Accept other defaults)
			5

Protocols
APPN
DLUR
	DLUR (checked to enable)
	Fully-qualified CP name of
	primary DLUS: STFNET.MVS1
		`APPN config>`**`set dlur`**
		`Enable DLUR`
		`Fully-qualified CP name of`
		`primary DLUS:` **`STFNET.MVS1`**
		(Accept other defaults)
			6

Protocols
APPN
TN3270E Server
General
	TN3270E (checked to enable)
	IP address : 172.128.1.2
	Automatic logoff (checked to enable)
		`APPN config>`**`tn`**
		`TN3270E config>`**`set`**
		`Enable TN3270E Server`
		`TN3270E Server IP Address:` **`172.128.1.2`**
		`Automatic logoff:` **`Yes`**
		(Accept other defaults)
			7

Table 23. TN3270 via DLUR over APPN (continued). See page 139 for a description and 168 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
APPN
TN3270E Server
Local PUs
	Link Station Name: PUPS08T
	Node ID: 12244
	
	Link Station Name: PUPS18T
	Node ID: 12245
		`TN3270E config>`**`exit`**
		`APPN config>`**`add loc`**
		`Station Name:` **`PUPS08T`**
		`Local Node ID:` **`12244`**
		(Accept other defaults)
		`APPN config>`**`add loc`**
		`Station Name:` **`PUPS18T`**
		`Local Node ID:` **`12245`**
		(Accept other defaults)
			8

Protocols
APPN
TN3270E Server
LUs
	Local PU Name: PUPS08T
	(click on **Implicit Pool**)
	LU name mask: @LU1A
	Number of implicit workstation definitions: 5
	Local PU Name: PUPS18T
	(click on **Implicit Pool**)
	LU name mask: @LU2A
	Number of implicit workstation definitions: 5
		`APPN config>`**`tn`**
		`TN3270E config>`**`add imp`**
		`Station Name:` **`PUPS08T`**
		`LU name mask:` **`@LU1A`**
		`Number of Implicit LUs in Pool:` **`5`**
		
		`TN3270E config>`**`add imp`**
		`Station Name:` **`PUPS18T`**
		`LU name mask:` **`@LU2A`**
		`Number of Implicit LUs in Pool:` **`5`**
			9

Notes:

1. **add dev** defines a single port, not an adapter.
2. The configuration program assigns an interface number to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the **add dev** command for each port you want to use, and the interface number (also known as ″net number″) is the output of the command.
3. You need a write-capable SNMP community only if you want to download configuration files from the configuration program directly into the router. SNMP is not required to TFTP a configuration file to the router.
4. When using APPN, you can either use High Performance Routing (HPR) or Intermediate Session Routing (ISR). HPR is the default and is what is used in this scenario.
5. The MAC address specified is the MAC address of the APPN host gateway.
6. The CP name of the DLUS is the host VTAM.
7. The Local Node IDs entered for these PUs need to match the IDNUM fields in the PU definitions in the host VTAM.
8. Beginning with MAS V3.2, TN3270E Server has its own command-line submenu.
9. For implicit LUs, you just have to define the pools. The `@LU1A` is a template that will be used to create the actual LU names in the pool. In this example, with 5 LUs in the pool, the LU names generated are `@LU1A2`, `@LU1A3`, `@LU1A4`, `@LU1A5` and `@LU1A6` which correspond to LOCADDRs 2-6 for the PU defined in VTAM. Similarly, @LU2A will generate `@LU2A2` through `@LU2A6`.

Dynamic Definition of Dependent LUs

This scenario was done with an MVS running VTAM V4R4 and an IBM 2216 which was ESCON attached to the host. The Network Utility had a two-port token-ring adapter in slot 1 and an ESCON adapter in slot 2. LSA loopback protocol was used for communication over the ESCON channel. The Network Utility was defined as a Network Node, and it used APPN/ISR to communicate with VTAM. The TN3270E PUs and LUs were connected to VTAM using DLUR/DLUS.

The DDDLU function was tested using the following VTAM definitions:

- X5303—An XCA major node, needed for ESCON connection using LSA
- SW5303N—A VTAM switched major node for the NN PU in Network Utility
- DDDPU—A VTAM switched major node for the dynamically defined LUs that contains only a PU statement with reference to an LUGROUP and with LUSEED parameter
- DDGROUP—The VTAM LUGROUP major node, with model definition for 3270 LUs

The 2216 configuration file for this scenario was DDD.

Figure 20 on page 173 shows the relationships of different parameters in both VTAM and the 2216 for this scenario.

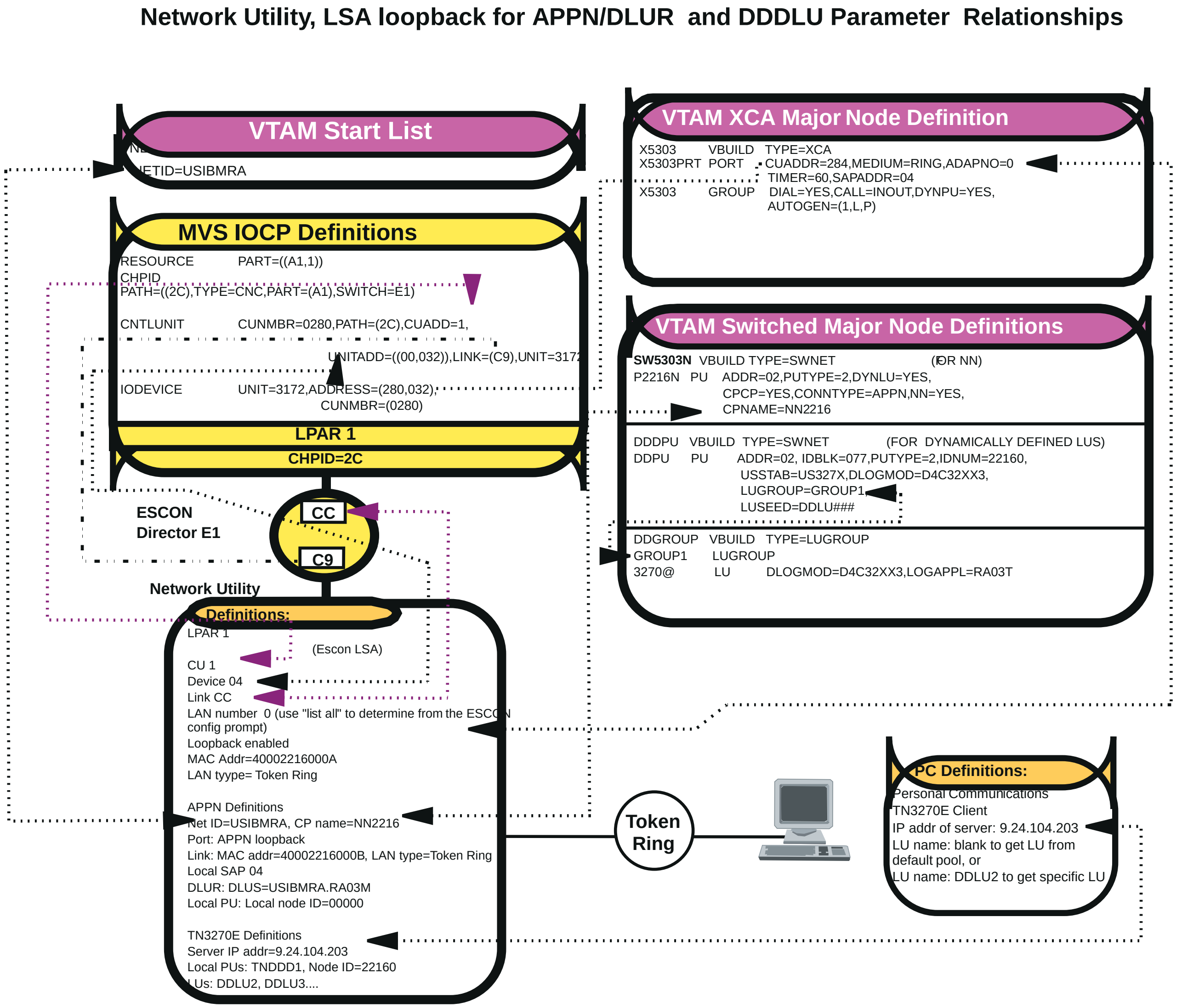

Network Utility, LSA loopback for APPN/DLUR and DDDLU Parameter Relationships
VTAM Start List
NETID=USIBMRA
MVS IOCP Definitions
RESOURCE PART=((A1,1))
CHPID PATH=((2C),TYPE=CNC,PART=(A1),SWITCH=E1)
CNTLUNIT CUNMBR=0280,PATH=(2C),CUADD=1,
UNITADD=((00,032)),LINK=(C9),UNIT=3172
IODEVICE UNIT=3172,ADDRESS=(280,032),
CUNMBR=(0280)
LPAR 1
CHPID=2C
ESCON
Director E1
CC
C9
Network Utility
Definitions:
LPAR 1
(Escon LSA)
CU 1
Device 04
Link CC
LAN number 0 (use "list all" to determine from the ESCON config prompt)
Loopback enabled
MAC Addr=40002216000A
LAN tyype= Token Ring
APPN Definitions
Net ID=USIBMRA, CP name=NN2216
Port: APPN loopback
Link: MAC addr=40002216000B, LAN type=Token Ring
Local SAP 04
DLUR: DLUS=USIBMRA.RA03M
Local PU: Local node ID=00000
TN3270E Definitions
Server IP addr=9.24.104.203
Local PUs: TNDDD1, Node ID=22160
LUs: DDLU2, DDLU3....
VTAM XCA Major Node Definition
X5303 VBUILD TYPE=XCA
X5303PRT PORT CUADDR=284,MEDIUM=RING,ADAPNO=0
TIMER=60,SAPADDR=04
X5303 GROUP DIAL=YES,CALL=INOUT,DYNPU=YES,
AUTOGEN=(1,L,P)
VTAM Switched Major Node Definitions
SW5303N VBUILD TYPE=SWNET (FOR NN)
P2216N PU ADDR=02,PUTYPE=2,DYNLU=YES,
CPCP=YES,CONNTYPE=APPN,NN=YES,
CPNAME=NN2216
DDDPU VBUILD TYPE=SWNET (FOR DYNAMICALLY DEFINED LUS)
DDPU PU ADDR=02, IDBLK=077,PUTYPE=2,IDNUM=22160,
USSTAB=US327X,DLOGMOD=D4C32XX3,
LUGROUP=GROUP1,
LUSEED=DDLU###
DDGROUP VBUILD TYPE=LUGROUP
GROUP1 LUGROUP
3270@ LU DLOGMOD=D4C32XX3,LOGAPPL=RA03T
Token Ring
PC Definitions:
Personal Communications
TN3270E Client
IP addr of server: 9.24.104.203
LU name: blank to get LU from default pool, or
LU name: DDLU2 to get specific LU

Figure 20. Parameter Relationships, Network Utility/2216 Running TN3270E With DDDLU And APPN/DLUR Using LSA Loopback Over ESCON Channel

Table 24 lists the actual VTAM XCA major node used.

Table 24. XCA Major Node X5303 for ESCON channel connection

```
**************************Top of Data ***********************
X5303    VBUILD TYPE=XCA
X5303    VBUILD TYPE=XCA
X5303PRT PORT  ADAPNO=0,
X5303PRT PORT  ADAPNO=0,                                     *
               CUADDR=284,                                   *
               SAPADDR=4,                                    *
               MEDIUM=RING
X5303GRP GROUP DIAL=YES,CALL=INOUT,DYNPU=YES,                *
               AUTOGEN=(1,L,P)
*********************** Bottom of Data *******************
```

Table 25 on page 174 contains the actual VTAM SW5303N switched major node for the Network Node in 2216.

Table 25. Switched Major Node SW5303N for 2216 Network Node

```
***************************** Top of Data ******************************
SW5303N  VBUILD TYPE=SWNET
P2216N   PU    ADDR=02,         X
               PUTYPE=2,        X
               CPCP=YES,        X
               CONNTYPE=APPN,   X
               USSTAB=US327X,   X
               NN=YES,          X
               DYNLU=YES,       X
               CPNAME=NN2216
**************************** Bottom of Data ****************************
```

For APPN Control Point PUs, such as the Network Node shown in Table 25, you do not need ID Block nor ID NUm numbers. Network Nodes can recognize each other by fully qualified network names.

Table 26. LUGROUP Major Node DDGROUP, Model for LU Definitions

```
***************************** Top of Data ******************************
DDGROUP  VBUILD TYPE=LUGROUP
GROUP1   LUGROUP
3270@    LU    DLOGMOD=D4C32XX3,LOGAPPL=RA03T      1
**************************** Bottom of Data ****************************
```

1. The name 3270@ specifies device type 3270. Using @ as the last character specifies that it matches any model number of the product, 3270. Of the LU statement parameters, only DLOGMOD and LOAGAPPL are shown. However, you may specify any additional LU parameters that you may require.

Table 27. Switched Major Node DDDPU for the Dynamically Defined LUs

```
***************************** Top of Data ******************************
DDDPU    VBUILD TYPE=SWNET
DDPU     PU    ADDR=02,                                               X
               IDBLK=077,        1                                    X
               IDNUM=22160,                                           X
               PUTYPE=2,                                              X
               USSTAB=US327X,                                         X
               LUGROUP=GROUP1,   2                                    X
               LUSEED=DDLU###,   3                                    X
               DLOGMOD=D4C32XX3
**************************** Bottom of Data ****************************
```

1. IBM Network Utility uses 077 as the IDBLK value.
2. This parameter points to LUGROUP statement with name GROUP1 in an LUGROUP major node. See 26.
3. With this LUSEED value, the dynamically created LUs will have names starting with DDLU, followed by the LU local address number in three decimal digit format.

Table 28. DDD Configuration Done With MAS 3.3 Configuration Program (Part 1 of 2)

Configuration Program Navigation	Configuration Program Values
Devices Slots	Slot 1: 2-Port TR Slot 2: ESCON

Table 28. DDD Configuration Done With MAS 3.3 Configuration Program (Part 1 of 2) (continued)

Configuration Program Navigation	**Configuration Program Values**
Devices Channel Adapters ESCON Interfaces ESCON Interfaces	Protocol Type: LSA LAN Type: Token Ring MAC Address (for LSA virtual LAN, VTAM side): e.g. 40002216000A Click on **Loopback** Click on **Add**
Devices Channel Adapters ESCON Interfaces ESCON Subchannels	Device Addresses: 4 Subchannel type: read/write LPAR: 1 Link address: CC CU: 1 Click on **Add**
Devices Channel Adapters APPN Loopback Net	LAN type: Token Ring MAC Address: 40002216000B Click on **Add**
System General	System name: DDD Location: Machine room Contact: your name
System Users	Name: userid Permission: Administrative Password: password Repeat password: password Click on **Add**
System SNMP Config Communities General	Name: public Access type: Read-write trap Click on **Add**
Protocols IP Details	Internal address: 9.24.104.203
Protocols IP Interfaces	Interface 1 (TR slot 1 port 2) IP address: 9.24.106.9 Subnet mask: 255.255.255.0 Click on **Add** IP address: 9.24.104.203 Subnet mask: 255.255.255.0 Click on **Add**
Protocols APPN General	Click on **APPN network node** Network ID: USIBMRA Control point name: NN2216
Protocols APPN DLUR	Click on **DLUR** Fully qualified name of primary DLUS: USIBMRA.RA03M

Table 29. DDD Configuration, Done With MAS 3.3 Configuration Program (Part 2 of 2)

Configuration Program Navigation	**Configuration Program Values**
Protocols APPN Interfaces	Click on the line item **APPN Net—Token Ring** Click on heading **Configure** (Tab General selected) Click on **Define APPN Port** Click on **Service Any Node** Click ***off*** **High Performance Routing (HPR)** Supported Click on **Support Multiple PUs** Select tab **Port Definition** Specify Local SAP address: 04
Protocols TN3270E Server General	Click on **TN3270E** IP address: 9.24.104.203
Protocols TN3270E Server Local PUs	Link station name: TNDDD1 Node ID: 22160 Primary DLUS: USIBMRA.RA03M Click on **Add**
Protocols TN3270E Server LUs	Select **TNDDD1** Click on heading **LUs** LU name: DDLU2 Class: Implicit NAU address: 2 Click on **Add** Repeat LU name, Class, NAU address Add sequence for each LU

Monitoring the configuration

In Network Utility, you can monitor the status of connections in use, pools, mappings and more, generally under **talk 5/appn/tn3270e**.

To get a list of all locally defined resources such as LUs, PUs and pools, you can use **talk 6/appn/tn3270e.** See Table 30 on page 177 for the TN3270E configuration.

Table 30. Listing of TN3270E Configuration Under talk 6/p app/tn3270e

```
DDD TN3270E config>LIST ALL
TN3270E Server Definitions
TN3270E enabled: YES
TN3270E IP Address: 9.24.104.203
TN3270E Port Number: 23
Default Pool Name : PUBLIC
NetDisp Advisor Port Number: 10008
Client IP Address Mapping : N
Keepalive type: NONE
Automatic Logoff: N        Timeout: 30
        Enable IP Precedence: N

DLUS Link Station: TNDDD1
        Fully-qualified CP name of primary DLUS: USIBMRA.RA03M
        Fully-qualified CP name of backup DLUS:
        Local Node ID: 22160
        Auto activate : YES
        Host Initiated Dynamic LU Definition : NO
        LU Name   NAU addr     Class            Assoc LU Name   Assoc NAU
addr
        ----------------------------------------------------------------------
        DDLU2        2     Implicit Workstation
        DDLU3        3     Implicit Workstation
        DDLU4        4     Implicit Workstation
--More--
        DDLU5        5     Implicit Workstation
        DDLU6        6     Explicit Workstation
        DDLU7        7     Explicit Workstation


Client IP Address mapping
--------------------------
Client IP Address   Address Mask       Resource Name
------------------------------------------------------


Multiple Port
------------------------------
Port Number    Enable TN3270E    Resource Name
-------------------------------------------------
DDD TN3270E config>
```

The commands in Table 31 on page 178 take you first to **talk 5/appn/tn3270e**.

The list status then displays the current status of TN3270E resources. There are no end user sessions active at this time, but a PU with 6 LUs is active in SSCP-LU session.

Table 31. Moving to talk 5/appn/tn3270 for TN3270E Status

```
DDD *TALK 5

DDD +PROTOCOL APPN
APPN GWCON
DDD APPN >TN3270E
TN3270E GWCON
DDD TN3270E >LIST STATUS
TN3270E Server Status Summary

TN3270E IP Address: 9.24.104.203
NetDisp Advisor Port Number: 10008
  Keepalive type: None
  Automatic Logoff: N
  Client IP Address mapping : N
  Number of connections               : 1
  Number of available LUA LU's        : 5
  Number of LUA LU's pending termination  : 0
  Number of defined LU's              : 6
  Number of connections in SSCP-LU state  : 0
  Number of connections in LU-LU state    : 1     1
 DDD TN3270E >
```

This number increases with the LU-LU sessions established between LUs defined under Network Utilities and VTAM applications.

After the first user has established a session and left the LU name field blank in the client, LIST CONNECTIONS under **talk 5/appn/tn3270e** looks like Table 32:

Table 32. First Session Established, LU Selected From Default Pool

```
DDD TN3270E >LIST CONNECTIONS
Connection information for all the LUs

Local LU  Class  Assoc LU   Client Addr     Status   Prim LU  Sec LU  Idle
Min
------------------------------------------------------------------------------
DDLU5      IW             9.24.106.217      LU-LU  RA03T07   DDLU005   0
DDD TN3270E >
```

After a second user gets his session, again with no specific LU or pool name, the list of connections looks like Table 33:

Table 33. Two Sessions Using the Default Pool

```
DDD TN3270E >LIST CONNECTIONS
Connection information for all the LUs

Local LU  Class  Assoc LU   Client Addr     Status   Prim LU  Sec LU  Idle
Min
------------------------------------------------------------------------------
DDLU2      IW             9.24.106.127      LU-LU  RA03T08   DDLU002   0
DDLU5      IW             9.24.106.217      LU-LU  RA03T07   DDLU005   4
DDD TN3270E >
```

Table 34 on page 179 is an example of how the list of connections looks when a third user, requesting explicit LU DDLU7, gets connected.

Table 34. List of Connections With Two Implicit and One Explicit LU User

```
DDD TN3270E >LIST CONNECTIONS
Connection information for all the LUs

Local LU  Class  Assoc LU   Client Addr     Status   Prim LU  Sec LU  Idle
Min
------------------------------------------------------------------------------
DDLU7      EW             9.24.106.146      LU-LU  RA03T09   DDLU007   0
DDLU2      IW             9.24.106.127      LU-LU  RA03T08   DDLU002   6
DDLU5      IW             9.24.106.217      LU-LU  RA03T07   DDLU005   6
DDD TN3270E >
```

Host Initiated Dynamic LU Definition

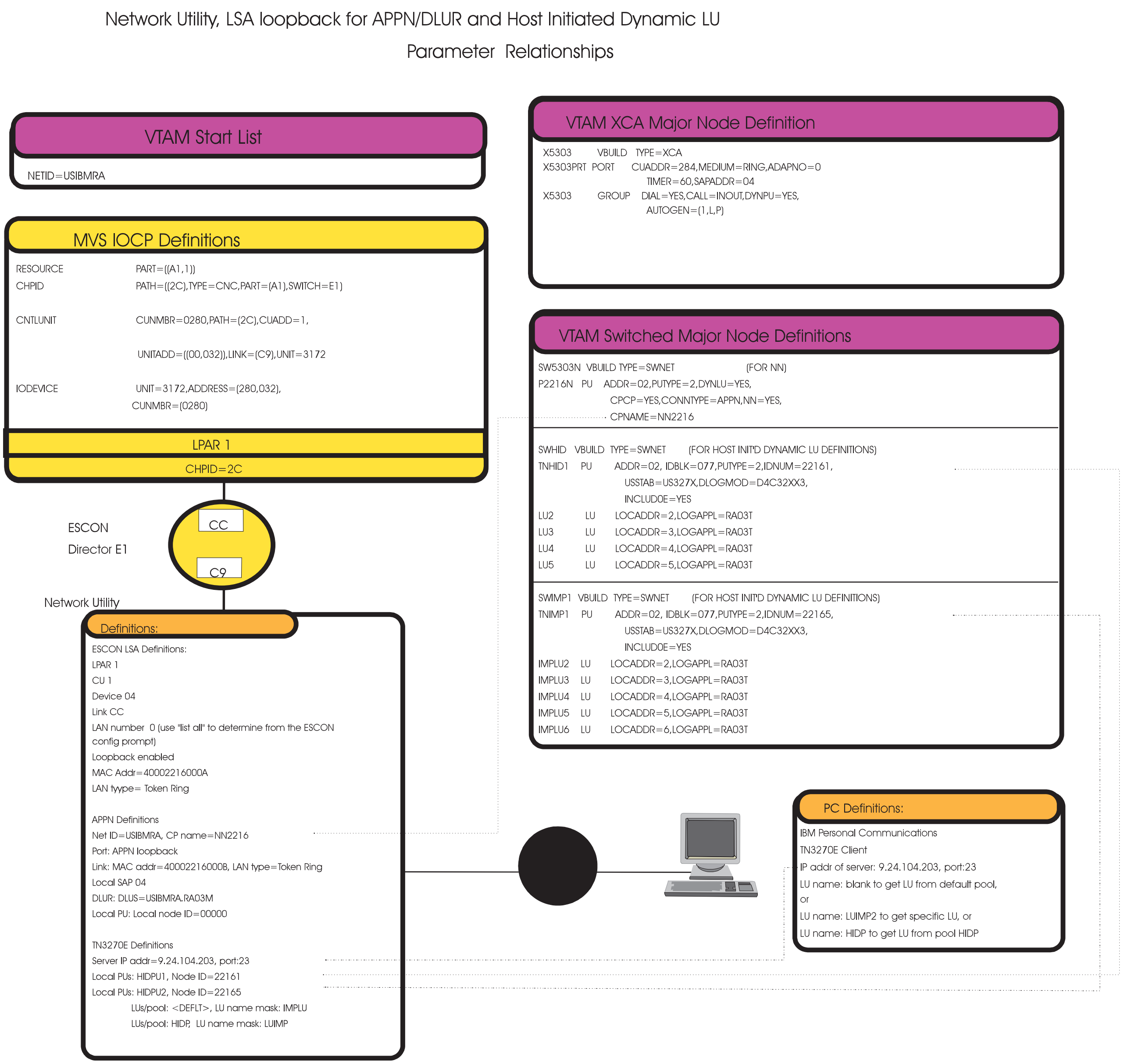
This scenario was established with an MVS running VTAM V4R4 and an IBM Network Utility, which was ESCON attached to the host. In the Network Utility, there was a two-port token-ring adapter in slot 1 and an ESCON adapter in slot 3. LSA loopback protocol was used for communication over the ESCON channel. The Network Utility was defined as a Network Node and it used APPN/ISR to communicate with VTAM. The TN3270E PUs and LUs were connected to VTAM using DLUR/DLUS.

The Host Initiated Dynamic LU (HIDLU) Definition function was tested using the following VTAM definitions:

- X5303—an XCA major node for LSA Loopback ESCON attached Network Utility. This is exactly the same as in previous scenario for DDDLU.
- SW5303N—a switched major node for Network Utility APPN Network Node. This is exactly the same as in previous scenario for DDDLU.
- SWHID—a switched major node containing LU definitions that will create explicit LU definitions in Network Utility.
- SWIMP—a switched major node containing LU definitions that will create implicit LUs in the Network Utility. Implicit LUs go to pool(s).

Note: To VTAM, all LUs—implicit or explicit—are defined the same. An LU becomes explicit or implicit based on pool definitions in the Network Utility.

The parameter relationships between VTAM and Network Utility configuration are shown in Figure 21 on page 180.


Network Utility, LSA loopback for APPN/DLUR and Host Initiated Dynamic LU
Parameter Relationships
VTAM Start List
MVS IOCP Definitions
LPAR 1
CHPID=2C
ESCON
Director E1
CC
C9
Network Utility
Definitions:
VTAM XCA Major Node Definition
VTAM Switched Major Node Definitions
PC Definitions:

```
NETID=USIBMRA
```

```
RESOURCE        PART=((A1,1))
CHPID           PATH=((2C),TYPE=CNC,PART=(A1),SWITCH=E1)

CNTLUNIT        CUNMBR=0280,PATH=(2C),CUADD=1,

                UNITADD=((00,032)),LINK=(C9),UNIT=3172

IODEVICE        UNIT=3172,ADDRESS=(280,032),
                CUNMBR=(0280)
```

```
ESCON LSA Definitions:
LPAR 1
CU 1
Device 04
Link CC
LAN number  0 (use "list all" to determine from the ESCON
config prompt)
Loopback enabled
MAC Addr=40002216000A
LAN tyype= Token Ring

APPN Definitions
Net ID=USIBMRA, CP name=NN2216
Port: APPN loopback
Link: MAC addr=40002216000B, LAN type=Token Ring
Local SAP 04
DLUR: DLUS=USIBMRA.RA03M
Local PU: Local node ID=00000

TN3270E Definitions
Server IP addr=9.24.104.203, port:23
Local PUs: HIDPU1, Node ID=22161
Local PUs: HIDPU2, Node ID=22165
        LUs/pool: <DEFLT>, LU name mask: IMPLU
        LUs/pool: HIDP,  LU name mask: LUIMP
```

```
X5303      VBUILD  TYPE=XCA
X5303PRT  PORT     CUADDR=284,MEDIUM=RING,ADAPNO=0
                     TIMER=60,SAPADDR=04
X5303      GROUP   DIAL=YES,CALL=INOUT,DYNPU=YES,
                     AUTOGEN=(1,L,P)
```

```
SW5303N  VBUILD TYPE=SWNET                  (FOR NN)
P2216N   PU    ADDR=02,PUTYPE=2,DYNLU=YES,
                 CPCP=YES,CONNTYPE=APPN,NN=YES,
                 CPNAME=NN2216

SWHID   VBUILD  TYPE=SWNET       (FOR HOST INIT'D DYNAMIC LU DEFINITIONS)
TNHID1    PU       ADDR=02, IDBLK=077,PUTYPE=2,IDNUM=22161,
                     USSTAB=US327X,DLOGMOD=D4C32XX3,
                     INCLUD0E=YES
LU2         LU      LOCADDR=2,LOGAPPL=RA03T
LU3         LU      LOCADDR=3,LOGAPPL=RA03T
LU4         LU      LOCADDR=4,LOGAPPL=RA03T
LU5         LU      LOCADDR=5,LOGAPPL=RA03T

SWIMP1  VBUILD  TYPE=SWNET       (FOR HOST INIT'D DYNAMIC LU DEFINITIONS)
TNIMP1    PU       ADDR=02, IDBLK=077,PUTYPE=2,IDNUM=22165,
                     USSTAB=US327X,DLOGMOD=D4C32XX3,
                     INCLUD0E=YES
IMPLU2    LU     LOCADDR=2,LOGAPPL=RA03T
IMPLU3    LU     LOCADDR=3,LOGAPPL=RA03T
IMPLU4    LU     LOCADDR=4,LOGAPPL=RA03T
IMPLU5    LU     LOCADDR=5,LOGAPPL=RA03T
IMPLU6    LU     LOCADDR=6,LOGAPPL=RA03T
```

```
IBM Personal Communications
TN3270E Client
IP addr of server: 9.24.104.203, port:23
LU name: blank to get LU from default pool,
or
LU name: LUIMP2 to get specific LU, or
LU name: HIDP to get LU from pool HIDP
```

Figure 21. Relationship of Parameters for HIDLU Definition Configuration

The actual SWHID switched major node is shown in Table 35.

Table 35. Switched Major Node SWHID1 Definition in VTAM

```
***************************** Top of Data ******************************
SWHID1   VBUILD TYPE=SWNET
TNHID1   PU     ADDR=02,                                               X
                IDBLK=077,      1                                      X
                IDNUM=22161,                                           X
                PUTYPE=2,                                              X
                USSTAB=US327X,                                         X
                INCLUD0E=YES,   2                                      X
                DLOGMOD=D4C32XX3
LU2      LU     LOCADDR=02,LOGAPPL=RA03T
LU3      LU     LOCADDR=03,LOGAPPL=RA03T
LU4      LU     LOCADDR=04,LOGAPPL=RA03T
LU5      LU     LOCADDR=05,LOGAPPL=RA03T
**************************** Bottom of Data ****************************
```

1. Network Utility uses IDBLK value of 077.
2. This parameter is new and needed for HIDLU definition.

Table 35 on page 180 is a Switched Major node definition for Network Utility LUs defined in pools.

These LUs will become defined in Network Utility with the names shown here (LU2 through LU5). Since the LUs representing these VTAM network NAUs are not already defined to the TN3270E server, the LUs will become explicit LUs.

For a TN3270 client, to get one of these LUs, the LU name must be defined in the client's LU name field.

You can also define Host Initiated Dynamic LUs to go to a pool in Network Utility. This is done in the MAS configuration program (or talk 6) by defining one or more pools in the Network Utility and specifying number of LUs or LU ranges, as well as an LU name mask for the LUs going into a pool. The following LUs under PU TNIMP1 were defined this way.

Table 36. Switched Major Node SWIMP1 Definition in VTAM

```
***************************** Top of Data ******************************
SWIMP1   VBUILD TYPE=SWNET
TNIMP1   PU    ADDR=02,                                                X
               IDBLK=077,      1                                       X
               IDNUM=22165,                                            X
               PUTYPE=2,                                               X
               USSTAB=US327X,                                          X
               INCLUD0E=YES,   2                                       X
               DLOGMOD=D4C32XX3
IMPLU2   LU    LOCADDR=02,LOGAPPL=RA03T
IMPLU3   LU    LOCADDR=03,LOGAPPL=RA03T
IMPLU4   LU    LOCADDR=04,LOGAPPL=RA03T
IMPLU5   LU    LOCADDR=05,LOGAPPL=RA03T
IMPLU6   LU    LOCADDR=06,LOGAPPL=RA03T
**************************** Bottom of Data ****************************
```

1. Network Utility uses IDBLK value of 077.
2. This parameter is new and needed for HIDLU definition.

LUs defined in switched major node SWIMP will become both implicit and explicit LUs in the Network Utility due to definitions in MAS configuration, as shown in Table 38 on page 183.

The first three LUs—IMPLU2, IMPLU3 and IMPLU4—go to pool HIDP. A TN3270E user can get one of these LUs by leaving the LU name field empty in the TN3270E client.

The next LU, IMPLU5, has been defined in the MAS configuration program to go into the default pool, <DEFLT>. A user can get this LU by leaving the LU name field empty in the TN3270E client.

For the last LU, IMPLU6, no pool has been defined in the Network Utility. It becomes thus an explicit LU, named IMPLU6.

Note: When you define a pool in TN3270E server, you must also give an LU name mask. This mask actually overrides the LU name of VTAM, and the implicit LUs that belong to a pool will have names based on the LU name mask. Only when you define explicit LUs (which do not go to a pool), do you get the LU names directly from VTAM.

Table 37. HIDLU Configuration, Done With MAS 3.3 Configuration Program (Part 1 of 2)

Configuration Program Navigation	**Configuration Program Values**
Devices Slots	Slot 1: 2-Port TR Slot 2: ESCON
Devices Channel Adapters ESCON Interfaces ESCON Interfaces	Protocol Type: LSA LAN Type: token ring MAC Address: (for LSA virtual LAN, VTAM side): e.g. 40002216000A Click on **Loopback** Click on **Add**
Devices Channel Adapters ESCON Interfaces ESCON Subchannels	Device address: 4 Subchannel type: read/write LPAR: 1 Link address: CC CU: 1 Click on **Add**
Devices Channel Adapters APPN Loopback Net	LAN type: token ring MAC address: 40002216000B Click on **Add**
System General	System name: HID Location: Machine room Contact: your name
System Users	Name: userid Permission: Administrative Password: password Repeat password: password Click on **Add**
System SNMP Config Communities General	Name: public Acess type: Read-write trap Click on **Add**
Protocols IP Details	Internal address: 9.24.104.203
Protocols IP Interfaces	Interface 1 (TR slot 1 port 2) IP address: 9.24.106.9 Subnet mask: 255.255.255.0 Click on **Add** IP address: 9.24.104.203 Subnet mask: 255.255.255.0 Click on **Add**
Protocols APPN General	Click on **APPN network node** Network ID: USIBMRA Control point name: NN2216
Protocols APPN DLUR	Click on **DLUR** Fully qualified name of primary DLUS: USIBMRA.RA03M

Table 38. HIDLU Configuration, Done With MAS 3.3 Configuration Program (Part 2 of 2)

Configuration Program Navigation	Configuration Program Values
Protocols APPN Interfaces	Click on line item **APPN Net—token ring** Click on heading **Configure** (Tab General selected) Click on **Define APPN Port** Click on **Service Any Node** Click ***off*** **High Performance Routing (HPR) Supported** Click on **Support Multiple PUs** Select tab **Port Definition** Specify Local SAP address: 04
Protocols TN3270E Server General	Click on **TN3270E** IP address: 9.24.104.203
Protocols TN3270E Server Pools	Pool name: HIDP Click on **Add**
Protocols TN3270E Server Local PUs	Link station name: HIDPU1 Node ID: 22161 Click on **Host-Initiated Dynamic LUs allowed for PU** Primary DLUS: USIMBRA.RA03M Click on **Add** Link station name: HIDPU2 Click on **Host-Initiated Dynamic LUs allowed for PU** Node ID: 22165 Primary DLUS: USIBMRA.RA03M Click on **Add**
Protocols TN3270E Server LUs	Select line HIDPU2 Click on heading Implicit pools Select pool name: <DEFLT> LU name mask: IMPLU Click on **Specify Address Ranges** Address Ranges: 5 Click on **Add** Select Pool name: HIDP LU name mask: LUIMP Click on **Specify Address Ranges** Address Ranges: 2-4 Click on **Add**

Monitoring the configuration

The HIDLU configuration can be monitored in talk 5 under **protocol appn** and **tn3270e**.

Under talk 5, you can monitor all the interfaces as shown in Table 39:

Table 39. Listing Interfaces Command Under Talk 5

```
HID +INTERFACE
                                             Self-Test  Self-Test Maintenance
Net   Net'   Interface   Slot-Port              Passed     Failed      Failed
0     0      TKR/0       Slot: 1   Port: 1           1          0        0
1     1      TKR/1       Slot: 1   Port: 2           1          0        0
2     2      ESCON/0     Slot: 3   Port: 1           1          0        0
3     2      LSA/0       Slot: 0   Port: 0           1          3        0
4     4      TKR/2       Slot: 0   Port: 0           1          0        0
```

Table 40. Statistics Under Talk 5

```
HID +STATISTICS
Net   Interface      Unicast  Multicast      Bytes    Packets      Bytes
                    Pkts Rcv   Pkts Rcv   Received     Trans      Trans
0     TKR/0            22521      25742    1399673      22522     472997
1     TKR/1            24301    1476136   97150812      23588     582533
2     ESCON/0          11453          0    2976076       9930    1481020
3     LSA/0            11452          0    2976060       9929    1480999
4     TKR/2                0          0          0          0          0
```

By issuing command **p app** (protocol appn) at talk 5 level, you can monitor APPN-related functions, such as verifying CP-CP connections as in Table 41.

Table 41. Verifying NN-NN Connection To VTAM

```
HID +PROTOCOL APPN
HID APPN >LIST CP-CP_SESSIONS
          CP Name Type     Status    ConWinner ConLoser ConWinner ConLoser
                                        ID        ID    Sense     Sense
===========================================================================
    USIBMRA.RA03M  NN     Active     3710C590  3710C592 00000000  00000000
```

TN3270E functions are under APPN and can be reached by issuing the command, **TN3270E**, as in Table 43 on page 185. At TN3270E level, you can only issue **LIST** command with the completions in Table 42:

Table 42. LIST Options Under TN3270E

```
HID TN3270E >LIST ?

Possible completions:
        CONNECTIONS
        MAPPING
        POOLS
        PORTS
        STATUS
(you may cycle through these commands by pressing the TAB key)
HID TN3270E >LIST
```

Table 43. Monitoring TN3270E Sessions

```
HID APPN >TN3270E
TN3270E GWCON
HID TN3270E >LIST CONNECTIONS
Connection information for all the LUs

Local LU  Class  Assoc LU   Client Addr     Status   Prim LU  Sec LU  Idle
Min
------------------------------------------------------------------------------
IMPLU5     IW             9.24.106.127      LU-LU  RA03T01   IMPLU5    38
```

Table 43 shows the TN3270E connections after the first session was established. The TN3270E client had defined no LU name, so an LU from the default pool was selected.

Another user then makes a TN3270E connection, specifying HIDP pool name as the LU name in the client. Table 44 is an example of what the list looks like:

Table 44. LIST CONNECTIONS After Second User In Session

```
HID TN3270E >LIST CONNECTIONS
Connection information for all the LUs

Local LU  Class  Assoc LU   Client Addr     Status   Prim LU  Sec LU  Idle
Min
------------------------------------------------------------------------------
LUIMP4     IW             9.24.106.217      LU-LU  RA03T07   IMPLU4    1
IMPLU5     IW             9.24.106.127      LU-LU  RA03T01   IMPLU5    45
```

If a third user establishes a session, defining IMPLU6 in the client, the list of connections looks like the one in Table 45:

Table 45. LIST CONNECTIONS After Third User Has Connected

```
HID TN3270E >LIST CONNECTIONS
Connection information for all the LUs

Local LU  Class  Assoc LU   Client Addr     Status   Prim LU  Sec LU  Idle
Min
------------------------------------------------------------------------------
IMPLU6     EW             9.24.106.146      LU-LU  RA03T09   IMPLU6    0
LUIMP4     IW             9.24.106.217      LU-LU  RA03T07   IMPLU4    4
IMPLU5     IW             9.24.106.127      LU-LU  RA03T01   IMPLU5    48
```

TN3270E Host On-Demand (HOD) Client Cache

For the HOD Client Cache function, you should configure both the TN3270E server and HOD function under Network Dispatcher. For this configuration, you can keep exactly the same configuration as for the TN3270E server, which is done in Table 28 on page 174 for DDDLU and Table 37 on page 182 for HIDLU. In this scenario, HIDLU definitions were used for TN3270E server.

For the HOD Client Cache part of the environment, Network Dispatcher ’Executor’ and HOD Client Cache should both be defined, in addition to the other definitions already done in HIDLU or DDDLU (in previous sections in this chapter).

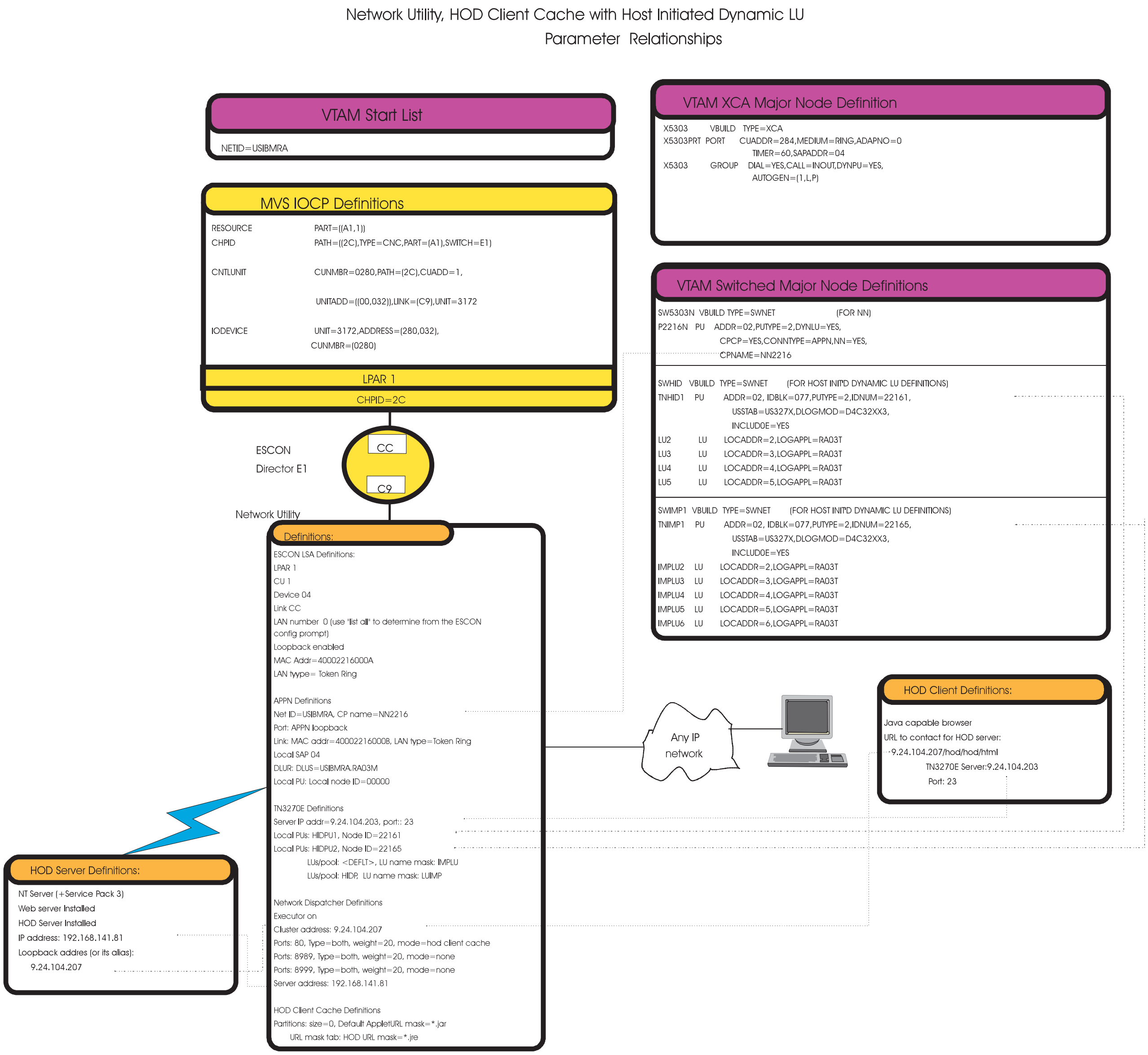
For the HOD Server installation in this scenario, the following products were installed: NT Server, NT Service Pack 3, Web Server, and HOD Server.

The Loopback address on the HOD Server box should be defined to point to the Network Dispatcher Cluster IP address. The Network Dispatcher Cluster IP address can not be pinged.

In the following scenario, the Loopback adapter was defined on MS NT Server in the following way:

1. Click **Start**, and then **Settings**.
2. Click **Control Panel**, double click **Network**.
3. Add MS Loopback Adapter Driver.
4. In the Network Window, click **Adapters**.
5. Select MS Loopback Adapter, click **OK**.
6. When prompted, insert installation CD or diskettes.
7. In the Network Window, click **Protocols**.
8. Select TCP/IP protocols and then click **Properties**.
9. Select Loopback Adapter and click **OK**.
10. Set the loopback adapter address to your Network Dispatcher Cluster address and accept the subnet mask (255.0.0.0). Do not enter a gateway address.

When defining the HOD Server, do not forget to take the following action as well. In the scenario for the Windows NT Server, the cluster IP address (9.24.104.207) is found in the gateway column after issuing the command **netstat -r**. In this column, you should delete the extraneous routes if you see more than one entry with the Cluster IP address. Delete the one starting with the same network address as Cluster IP address and with the rest being zeros. In this case, the entry to be deleted is the line with IP address 9.0.0.0 .



VTAM Start List

```
NETID=USIBMRA
```

MVS IOCP Definitions

```
RESOURCE        PART=((A1,1))
CHPID           PATH=((2C),TYPE=CNC,PART=(A1),SWITCH=E1)

CNTLUNIT        CUNMBR=0280,PATH=(2C),CUADD=1,

                 UNITADD=((00,032)),LINK=(C9),UNIT=3172

IODEVICE        UNIT=3172,ADDRESS=(280,032),
               CUNMBR=(0280)
```

VTAM XCA Major Node Definition

```
X5303      VBUILD  TYPE=XCA
X5303PRT  PORT    CUADDR=284,MEDIUM=RING,ADAPNO=0
                    TIMER=60,SAPADDR=04
X5303      GROUP   DIAL=YES,CALL=INOUT,DYNPU=YES,
                    AUTOGEN=(1,L,P)
```

VTAM Switched Major Node Definitions

```
SW5303N  VBUILD TYPE=SWNET              (FOR NN)
P2216N   PU   ADDR=02,PUTYPE=2,DYNLU=YES,
               CPCP=YES,CONNTYPE=APPN,NN=YES,
               CPNAME=NN2216

SWHID   VBUILD  TYPE=SWNET       (FOR HOST INIT'D DYNAMIC LU DEFINITIONS)
TNHID1   PU      ADDR=02, IDBLK=077,PUTYPE=2,IDNUM=22161,
                   USSTAB=US327X,DLOGMOD=D4C32XX3,
                   INCLUD0E=YES
LU2        LU      LOCADDR=2,LOGAPPL=RA03T
LU3        LU      LOCADDR=3,LOGAPPL=RA03T
LU4        LU      LOCADDR=4,LOGAPPL=RA03T
LU5        LU      LOCADDR=5,LOGAPPL=RA03T

SWIMP1  VBUILD  TYPE=SWNET       (FOR HOST INIT'D DYNAMIC LU DEFINITIONS)
TNIMP1   PU      ADDR=02, IDBLK=077,PUTYPE=2,IDNUM=22165,
                   USSTAB=US327X,DLOGMOD=D4C32XX3,
                   INCLUD0E=YES
IMPLU2   LU     LOCADDR=2,LOGAPPL=RA03T
IMPLU3   LU     LOCADDR=3,LOGAPPL=RA03T
IMPLU4   LU     LOCADDR=4,LOGAPPL=RA03T
IMPLU5   LU     LOCADDR=5,LOGAPPL=RA03T
IMPLU6   LU     LOCADDR=6,LOGAPPL=RA03T
```

Definitions:

```
ESCON LSA Definitions:
LPAR 1
CU 1
Device 04
Link CC
LAN number  0 (use "list all" to determine from the ESCON
config prompt)
Loopback enabled
MAC Addr=40002216000A
LAN tyype= Token Ring

APPN Definitions
Net ID=USIBMRA, CP name=NN2216
Port: APPN loopback
Link: MAC addr=40002216000B, LAN type=Token Ring
Local SAP 04
DLUR: DLUS=USIBMRA.RA03M
Local PU: Local node ID=00000

TN3270E Definitions
Server IP addr=9.24.104.203, port:: 23
Local PUs: HIDPU1, Node ID=22161
Local PUs: HIDPU2, Node ID=22165
          LUs/pool: <DEFLT>, LU name mask: IMPLU
          LUs/pool: HIDP,  LU name mask: LUIMP

Network Dispatcher Definitions
Executor on
Cluster address: 9.24.104.207
Ports: 80, Type=both, weight=20, mode=hod client cache
Ports: 8989, Type=both, weight=20, mode=none
Ports: 8999, Type=both, weight=20, mode=none
Server address: 192.168.141.81

HOD Client Cache Definitions
Partitions: size=0, Default AppletURL mask=*.jar
     URL mask tab: HOD URL mask=*.jre
```

HOD Client Definitions:

```
Java capable browser
URL to contact for HOD server:
 9.24.104.207/hod/hod/html
           TN3270E Server:9.24.104.203
           Port: 23
```

HOD Server Definitions:

```
NT Server (+Service Pack 3)
Web server Installed
HOD Server Installed
IP address: 192.168.141.81
Loopback addres (or its alias):
    9.24.104.207
```

Figure 22. HOD Client Cache Parameter Relationships

For the Host On-Demand Client Cache configuration, the HOD is only added on top of the DDDLU or HIDLU configuration as done previously in Table 28 on page 174 or Table 37 on page 182, since these already contain TN3270E configurations.

Table 46. HOD Client Cache Configuration

Configuration Program Navigation	**Configuration Program Values**
Devices Interfaces	Interface 0: Slot/Port=1/1 Click **Interface**
Protocols IP Interfaces Addresses	Click on **Interface 0** IP address: 192.168.141.82 Mask 255.255.255.240 in our scenario Click on **ADD**

Table 46. HOD Client Cache Configuration (continued)

<table>
<tr><th>Configuration Program Navigation</th><th colspan="4">Configuration Program Values</th></tr>
<tr><td>Protocols
IP
OSPF
Interfaces</td><td colspan="4">Click on <b>address 192.168.141.82</b>
Check OSPF box</td></tr>
<tr><td>Features
Network Dispatcher
Executor</td><td colspan="4">Click <b>Executor</b> (executor ’on’)</td></tr>
<tr><td>Features
Network Dispatcher
Clusters
Details</td><td colspan="4">Cluster address: 9.24.104.207
The rest is default values.
Click on <b>ADD</b></td></tr>
<tr><td>Features
Network Dispatcher
Clusters
Ports</td><td><i>Number</i>
80
8989
8999</td><td><i>Type</i>
Both
Both
Both</td><td><i>Mode</i>
HOD Client
None
None</td><td><i>Weight</i>
20
20
20</td></tr>
<tr><td>Features
Network Dispatcher
Clusters
Servers</td><td colspan="4">For all three ports:
Address: 192.168.141.81 (NT Server)
Weight: 20
Server state: up
Click <b>ADD</b></td></tr>
<tr><td>Features
Network Dispatcher
Clusters
HOD Client Cache
Proxies</td><td colspan="4">Default values excepted</td></tr>
<tr><td>Features
Network Dispatcher
Clusters
HOD Client Cache
Partitions</td><td colspan="4">Partitions: Default applet: *.jar
The other parameters left to default
URL mask: URL mask: *jre
URL mask type: Include
Click on <b>ADD</b></td></tr>
</table>

In this example, Netscape browser was pointed to **http://9.24.104.207/hod/hod.html** and the screen seen in Figure 23 on page 189 came up. Right mouse-click on the 3270 icon to get the screen seen in Figure 24 on page 189. In Figure 24 on page 189, the TN3270E server parameters are defined the same as in the Network Utility.

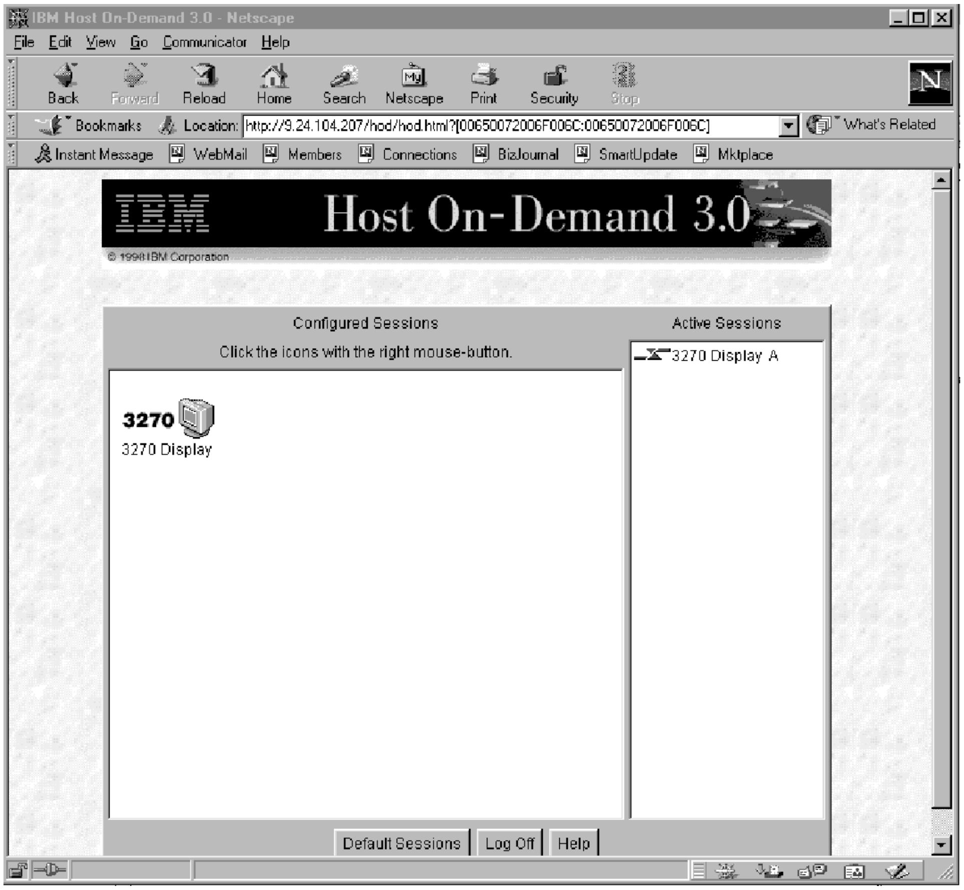
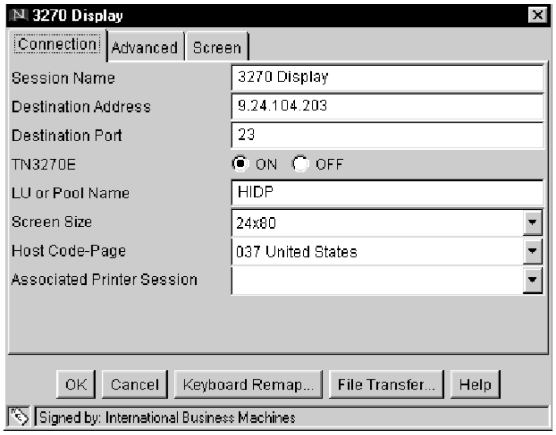

IBM Host On-Demand 3.0 - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Stop
Bookmarks Location: http://9.24.104.207/hod/hod.html?[00650072006F006C:00650072006F006C]
What's Related
Instant Message WebMail Members Connections BizJournal SmartUpdate Mktplace
IBM Host On-Demand 3.0
© 1998 IBM Corporation
Configured Sessions
Click the icons with the right mouse-button.
Active Sessions
3270 Display A
3270
3270 Display
Default Sessions
Log Off
Help

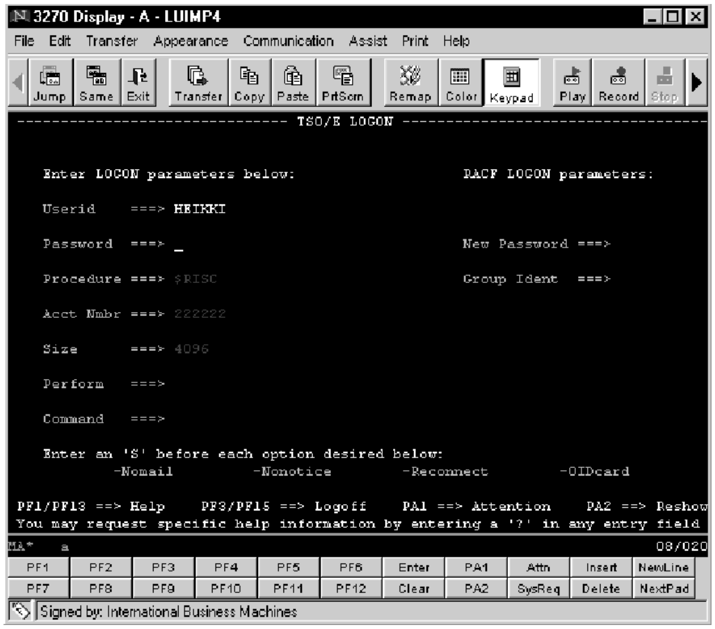
Figure 23. HOD Client Screen On Internet Browser (Netscape)


3270 Display
Connection Advanced Screen
OK Cancel Keyboard Remap... File Transfer... Help
Signed by: International Business Machines

Session Name	3270 Display
Destination Address	9.24.104.203
Destination Port	23
TN3270E	● ON ○ OFF
LU or Pool Name	HIDP
Screen Size	24x80
Host Code-Page	037 United States
Associated Printer Session	

Figure 24. TN3270E Server Definition on HOD Client

After this definition, if you double-click (with left mouse button) on the 3270 icon, the following screen (Figure 25 on page 190) comes up.


3270 Display - A - LUIMP4
File Edit Transfer Appearance Communication Assist Print Help
Jump Same Exit Transfer Copy Paste PrtScrn Remap Color Keypad Play Record Stop
PF1 PF2 PF3 PF4 PF5 PF6 Enter PA1 Attn Insert NewLine
PF7 PF8 PF9 PF10 PF11 PF12 Clear PA2 SysReq Delete NextPad
Signed by: International Business Machines

```
------------------------------ TSO/E LOGON ---------------------------------


   Enter LOGON parameters below:                    RACF LOGON parameters:

   Userid    ===> HEIKKI

   Password  ===> _                                 New Password ===>

   Procedure ===> $RISC                             Group Ident  ===>

   Acct Nmbr ===> 222222

   Size      ===> 4096

   Perform   ===>

   Command   ===>

   Enter an 'S' before each option desired below:
           -Nomail         -Nonotice        -Reconnect        -OIDcard

 PF1/PF13 ==> Help    PF3/PF15 ==> Logoff    PA1 ==> Attention    PA2 ==> Reshow
 You may request specific help information by entering a '?' in any entry field
MA*   a                                                                  08/020
```

Figure 25. HOD Client Screen After Connection Established

Monitoring the Configuration

The following commands are issued in order to monitor the HOD configuration.

Table 47. Start of Monitoring Cache in T5/ELS

```
HODCAC0 *TALK 5
HODCAC0 +EVENT
HODCAC0 ELS>NODISPLAY SUBSYSTEM ALL ALL
Complete
HODCAC0 ELS>DISPLAY SUBSYSTEM WEBH ALL
HODCAC0 ELS>    ..(Ctrl-P)
HODCAC0 *TALK 2
:
00:00:01  DOLOG: Server 192.168.141.81 has been set up.
:
00:20:01 WEBH.017: Client connection 31AE11C accepted as Socket 31BF564
00:20:01 WEBH.015: Conn (31AE11C) HTTP Proxy(cluster 9.24.104.207 port 80)
partition (0) opened
00:20:01 WEBH.009: HTTP Proxy(cluster 9.24.104.207 port 80) conn (31AE11C)
new req being parsed
00:20:01 WEBH.012: HTTP Proxy(cluster 9.24.104.207 port 80) partition (0)
conn (31AE11C) not using cache because object not found in cache     1
 :
 :
 :
```

1. For the first time, the client issued a request of Java applets from the Cluster address. Therefore, this message explains that it is not found in the Network Utility’s cache.

Table 48. Monitoring HOD Client Cache Definition

```
HODAC0 +FEATURE
Feature name or number [WAN Restoral System] ? hod
Host On-Demand Client Cache Console
HODCAC0 HOD Client Cache>LIst All
HOD Client Cache Partition 0    Status: Enabled
     Cluster address: 9.24.104.207  Port 80
1 partition(s) active.
External Cache Manager: Disabled
```

Table 49. Listing HOD Client Cache

```
HODCAC0 HOD Client Cache>LIst PArtition
HOD Client Cache Partition 0      Status: Enabled
        Cluster address: 9.24.104.207   Port 80
Partition size: Current - 1030296 bytes  Highest - 1030296 bytes  Maximum - Unlimited
Number of objects: Current - 37  Highest - 37  Maximum - Unlimited
Maximum object size: Unlimited
HOD Client Cache purge interval : 600 minute(s)
Hit ratio: 61%
Total number of hits:  59      1
Total number of misses:  37
Object Excluded (Object too large):         0
                (Object expired)            0
                (DONT CACHE header):        0
                (URL Mask excluded):        0
                (Image excluded):           0
                (Static object excluded):   0
                (Dynamic object excluded):  0
                (Cache disabled):           0
Objects explicitly Included: 0
Total number of purged objects: 0
Purged objects (Cache full):     0
               (Object stale):   0
               (Purged by user): 0
               (Invalidation):   0

```

1. Number of cache hits in Network Utility. When the HOD Clients hit the cache, the Java applets are delivered/downloaded from Network Utility and no load or traffic on the HOD Server will be installed on the NT Server.

Table 50. Displaying the Screen When Another HOD Client Requests Java Applets

```
HODCAC0 HOD Client Cache>LIst PArtition
HOD Client Cache Partition 0      Status: Enabled
         Cluster address: 9.24.104.207   Port 80
Partition size: Current - 1030296 bytes  Highest - 1030296 bytes  Maximum - Unlimited
Number of objects: Current - 37  Highest - 37  Maximum - Unlimited
Maximum object size: Unlimited
HOD Client Cache purge interval : 600 minute(s)
Hit ratio: 61%              1
Total number of hits:  59     2
Total number of misses:  37
Object Excluded (Object too large):         0
                (Object expired)            0  .....
:
:
```

1. Cache hit ratio increases with the increasing HOD client requests.
2. Total number of hits to the cache also increases with other HOD client requests. These prove that Java applets are delivered from Network Utility HOD Client Cache.

Now, the TN3270E connections can be monitored in the same way as in DDDLU and HIDLU sections earlier in this chapter.

TN3270E Subarea SNA over DLSw

For the TN3270E Subarea SNA over DLSw function scenario, see Figure 26 on page 195. Network Utility A is ESCON channel attached and also connected to Network Utility B over a PPP link. As seen in the configuration screens below, there is no APPN function on Network Utility A since the connection to VTAM-Host is Subarea. On the host, a VTAM Switched Major Node (with IDNUM parameter) pointing to the Local Node-ID in Network Utility (which is ’221B1’ in our sample scenario) needs to be defined. Although this is a pure Subarea SNA connection, the definition is done under APPN on Network Utility B.

Table 51. DLSw Configuration on Network Utility A

```
dlsa-ok DLSw config>ENABLE DLSW
Data Link Switching is now enabled
dlsa-ok DLSw config>ADD TCP
Enter the DLSw neighbor IP Address [0.0.0.0]? 172.16.220.253
Connectivity Setup Type (a/p) [a]? p
Transmit Buffer Size (Decimal) [5120]?
Receive Buffer Size (Decimal) [5120]?
Maximum Segment Size (Decimal) [1024]?
TCP Keepalive (E/D) [D]?
NetBIOS SessionAlive Spoofing (E/D) [D]?
Neighbor Priority (H/M/L) [M]?
TCP Neighbor has been changed
dlsa-ok DLSw config>OPEN-SAP
Enter Interface number [0]? 1
Enter the SAP in hex (range 0-FE), 'SNA', 'NB', or 'LNM'  [4]? sna
```

This is the definition needed for Network Utility A as seen in Table 51 above andFigure 26 on page 195.

When Network Utility B is configured, DLSw and TN3270E Server under the APPN prompt need to be configured.

Table 52. DLSw Configuration on Network Utility B

```
dlsb-ok DLSw config>ENABLE DLSW
Data Link Switching is now enabled
dlsb-ok DLSw config>ADD TCP
Enter the DLSw neighbor IP Address [0.0.0.0]? 9.24.104.203
Record already exists, can be changed
Connectivity Setup Type (a/p) [a]? a
Transmit Buffer Size (Decimal) [5120]?
Receive Buffer Size (Decimal) [5120]?
Maximum Segment Size (Decimal) [1024]?
TCP Keepalive (E/D) [D]?
NetBIOS SessionAlive Spoofing (E/D) [D]?
Neighbor Priority (H/M/L) [M]?
TCP Neighbor has been changed
dlsb-ok DLSW config>OPEN-SAP
Enter Interface number [0]? 1
Enter SAP in hex (range 0-FE), 'SNA', 'NB', or 'LNM'  [4]? sna
dlsb-ok DLSw config>EXIT
```

Table 53. Loading APPN and TN3270E Packages If Not Already Loaded

```
2216 Config>load add package appn
appn package configured successfully
This change requires a reload.

2216 Config>load add package tn3270e
tn3270e package configured successfully
This change requires a reload.
```

If the APPN and/or TN3270E packages are not loaded, they must be loaded from the operational code to the memory in order for you to work with them. To load APPN and TN3270E packages, see Table 53 above.

Table 54. Adding Link Station under APPN

```
dlsb-ok Config>PROTOCOL APPN
dlsb-ok APPN config>ADD/UPDATE LINK-STATION
APPN Station
Port name for the link station [ ]? dls65
Station name (Max 8 characters) [ ]? tnpub1
WARNING!! You are changing an existing record.
        Activate link automatically (Y)es (N)o  [Y]
        MAC address of adjacent node [40002216000A]?
        SAP address of adjacent node (04-EC) [4]?
        Solicit SSCP Session: (Y)es (N)o  [Y]?
                Local Node ID (5 hex digits)  [221B1]?
                Enable Host Initiated Dynamic LU Definition : (Y)es (N)o  [N]?
        Local SAP address (04-EC)  [4]?
        Does link support APPN function: (Y)es (N)o  [N]?  N
Edit TG Characteristics: (Y)es (N)o  [N}?
Write this record? [Y]?  y
The record has been written.
```

Table 55. Adding Port Under APPN

```
dlsb-ok APPN config>ADD/UPDATE PORT
APPN Port
Link Type: (P)PP, (FR)AME RELAY, (E)THERNET, (T)OKEN RING,
(M)PC, (S)DLC, (X)25, (FD)DI, (D)LSw, (A)TM, (I)P, [ ]? d
Port Name (Max 8 characters) [D65534]? dls65

WARNING!! You are changing an existing record.
Enable APPN on this port (Y)es (N)o? y
Port Definition
     Support multiple PU (Y)es (N)o [Y] y
All active port names will be of the form <port name sap>
     Service any node: (Y)es (N)o [N] n
     Maximum BTU size (768-4096) [2048]?
     Maximum number of link stations (1-65535) [65535]?
     Percent of link stations reserved for incoming calls (0-100) [0]?
     Percent of link stations reserved for outgoing calls (0-100) [0]?
     Local SAP address (04-EC) [4]?
     Locally administered MAC address (hex) [40002216B00B]?
Edit TG characteristics (Y)es (N)o [N]?
Write this record? [Y]? y
The record has been written.
```

Table 56. Definition of TN3270E Server Under APPN

```
dlsb-ok APPN config>TN3270E
lsadirect TN3270E config>SET
TN3270E Server Parameters
        Enable TN3270E Server (Y/N) [Y]?
        TN3270E Server IP Address [9.24.104.203]? 172.16.220.2
        Port Number [23]?
        Enable Client Address Mapping (Y/N) [N]?
        Default Pool name (Max 8 characters) [PUBLIC]?
        NetDisp Advisor Port Number [10008]?
        Keepalive type:
         0 = none,
         1 = Timing Mark,
         2 = NOP [0]?
        Automatic Logoff (Y/N) [N]?
        Enable IP Precedence (Y/N) [N]?
Write this record? [Y]?
dlsb-ok TN3270E config>ADD/UPDATE IMPLICIT-POOL
TN3270E Server Implicit Definitions
        Pool Name (Max 8 characters)  [<DEFLT>]?
        Station Name (Max 8 characters)  []? tnpub1
WARNING!! You are changing an existing record.
        LU Name Mask (Max 5 characters)  [@01LU]?
        LU Type  ( 1 - 3270 mod 2 display
                   2 - 3270 mod 3 display
                   3 - 3270 mod 4 display
                   4 - 3270 mod 5 display) [1]? 1
        Specify LU Address Ranges (s) (y/n)  [N]?
        Number of Implicit LUs in Pool(1-253)  [5]?
Write this record? [Y]? y
The record has been written.
```

These screens complete the definition of TN3270E server on Network Utility B.

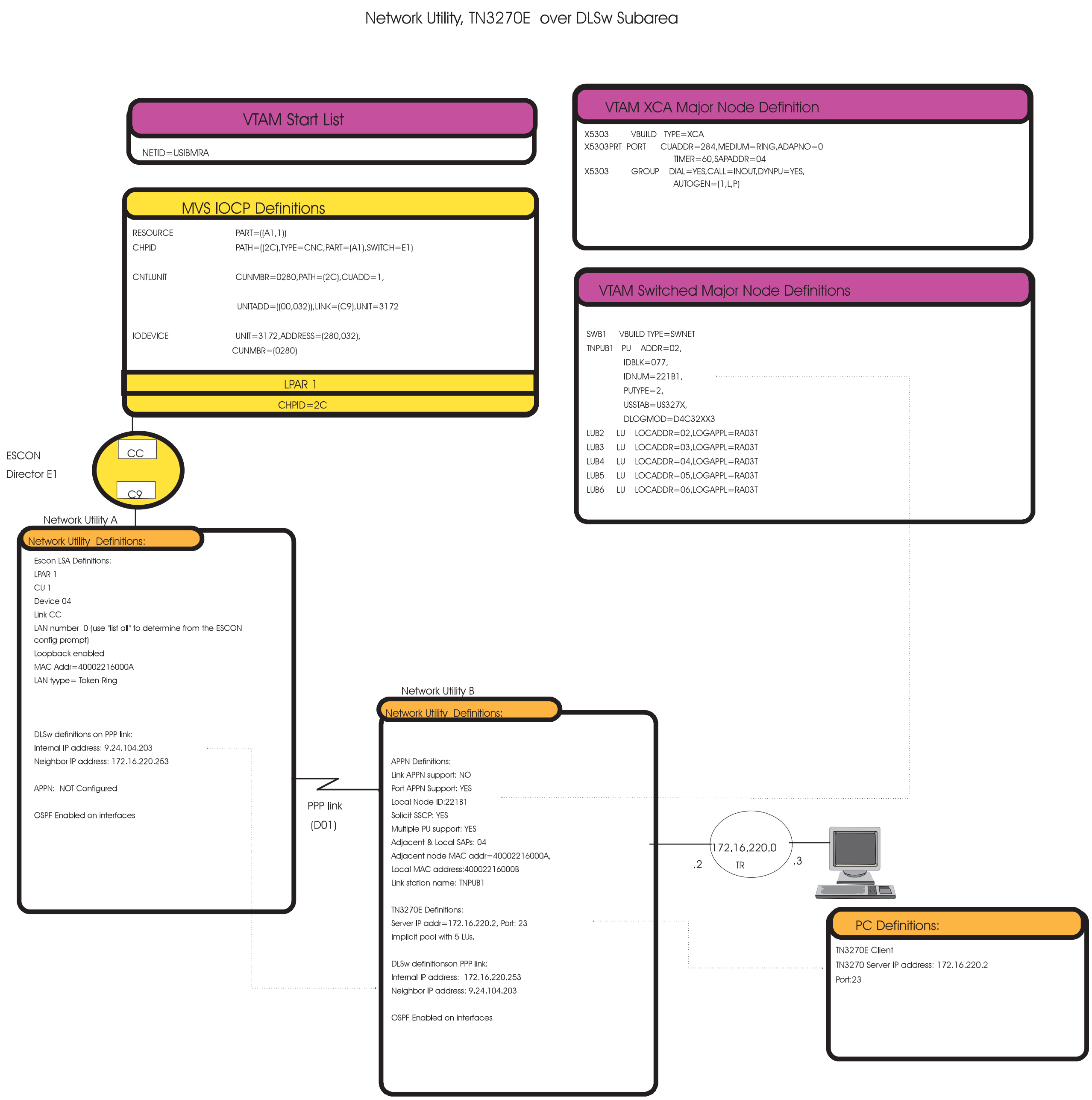
The corresponding definition on VTAM is done like Table 57 below:

Table 57. VTAM Switched Major Node Definition

```
SWB1                VBUILD     TYPE=SWNET
TNPUB1              PU    ADDR=02,                           X
                          IDBLK=077,                       X
                          IDNUM=221B1,                        X
                          PUTYPE=2,                           X
                          USSTAB=US327X,                      X
                          DLOGMOD=D4C32XX3
LUB2                LU    LOCADDR=02,LOGAPPL=RA03T
LUB3                LU    LOCADDR=03,LOGAPPL=RA03T
LUB4                LU    LOCADDR=04,LOGAPPL=RA03T
LUB5                LU    LOCADDR=05,LOGAPPL=RA03T
LUB6                LU    LOCADDR=06,LOGAPPL=RA03T
```

After these definitions are in place, the function is ready to work on Network Utility boxes.

See Figure 26 on page 195 for the general picture of the configuration.



VTAM Start List

```
NETID=USIBMRA
```

MVS IOCP Definitions

```
RESOURCE        PART=((A1,1))
CHPID           PATH=((2C),TYPE=CNC,PART=(A1),SWITCH=E1)

CNTLUNIT        CUNMBR=0280,PATH=(2C),CUADD=1,

                 UNITADD=((00,032)),LINK=(C9),UNIT=3172

IODEVICE        UNIT=3172,ADDRESS=(280,032),
               CUNMBR=(0280)
```

VTAM XCA Major Node Definition

```
X5303       VBUILD   TYPE=XCA
X5303PRT  PORT      CUADDR=284,MEDIUM=RING,ADAPNO=0
                         TIMER=60,SAPADDR=04
X5303       GROUP    DIAL=YES,CALL=INOUT,DYNPU=YES,
                         AUTOGEN=(1,L,P)
```

VTAM Switched Major Node Definitions

```
SWB1     VBUILD TYPE=SWNET
TNPUB1   PU    ADDR=02,
          IDBLK=077,
          IDNUM=221B1,
          PUTYPE=2,
          USSTAB=US327X,
          DLOGMOD=D4C32XX3
LUB2     LU    LOCADDR=02,LOGAPPL=RA03T
LUB3     LU    LOCADDR=03,LOGAPPL=RA03T
LUB4     LU    LOCADDR=04,LOGAPPL=RA03T
LUB5     LU    LOCADDR=05,LOGAPPL=RA03T
LUB6     LU    LOCADDR=06,LOGAPPL=RA03T
```

Network Utility Definitions:

```
Escon LSA Definitions:
LPAR 1
CU 1
Device 04
Link CC
LAN number  0 (use "list all" to determine from the ESCON
config prompt)
Loopback enabled
MAC Addr=40002216000A
LAN tyype= Token Ring


DLSw definitions on PPP link:
Internal IP address: 9.24.104.203
Neighbor IP address: 172.16.220.253

APPN:  NOT Configured

OSPF Enabled on interfaces
```

Network Utility Definitions:

```
APPN Definitions:
Link APPN support: NO
Port APPN Support: YES
Local Node ID:221B1
Solicit SSCP: YES
Multiple PU support: YES
Adjacent & Local SAPs: 04
Adjacent node MAC addr=40002216000A,
Local MAC address:40002216000B
Link station name: TNPUB1

TN3270E Definitions:
Server IP addr=172.16.220.2, Port: 23
Implicit pool with 5 LUs,

DLSw definitionson PPP link:
Internal IP address:  172.16.220.253
Neighbor IP address: 9.24.104.203

OSPF Enabled on interfaces
```

PC Definitions:

```
TN3270E Client
TN3270 Server IP address: 172.16.220.2
Port:23
```

Figure 26. TN3270E Over DLSw Subarea Connection Parameter Relationships

Monitoring the TN3270E Over DLSw SNA Subarea Configuration

After defining the definitions above, the configuration and its status must be monitored both on the Network Utility and VTAM sides.

Table 58. Display of VTAM Screen for the LUs That We Defined in Network Utility B

```
D NET,ID=SWB1,E
IST097I DISPLAY ACCEPTED
IST075I NAME=SWB1, TYPE=SW SNA MAJ NODE 774
IST486I STATUS=ACTIV, DESIRED STATE-ACTIV
IST1656I VTAMTOPO=REPORT, NODE REPORTED - YES
IST084I NETWORK RESOURCES:
IST089I TNPUB1     TYPE=PU_T2.1        , ACTIV
IST089I LUB2       TYPE=LOGICAL UNIT   , ACTIV
IST089I LUB3       TYPE=LOGICAL UNIT   , ACTIV
IST089I LUB4       TYPE=LOGICAL UNIT   , ACTIV
IST089I LUB5       TYPE=LOGICAL UNIT   , ACTIV
IST089I LUB6       TYPE=LOGICAL UNIT   , ACTIV
IST314I END
```

The configuration of Network Utility B can be monitored as seen below.

Table 59. Display of TN3270E Server Under 'T 6'

```
dlsb-ok *TALK 6
Gateway user configuration
dlsb-ok Config>PROTOCOL APPN
dlsb-ok APPN config>TN3270E
dlsb-ok TN3270E config>LIST ALL
TN3270E Server Definitions
TN3270E enabled: YES
TN3270E IP Address: 172.16.220.2
TN3270E Port Number: 23
Default Pool Name: PUBLIC
NetDisp Advisor Port Number: 10008
Client IP Address Mapping: N
Keepalive type: NONE
Automatic Logoff: N    Timeout: 30
        Enable IP Precedence: N

Link Station: TNPUB1
        Local Node ID: 221B1
        Auto Activate: YES
        Host Initiated Dynamic LU Definition: NO
        Implicit Pool Information
        Pool Name: <DEFLT>
               Number of LUs: 5
               LU Mask: @01LU
        LU Name    NAU Addr     Class     Assoc LU Name  Assoc NAU addr
-------------------------------------------------------------------------
        LUB2          2      Explicit Workstation

Client IP Address Mapping
-------------------------
Client IP Address              Address Mask      Resource Name
--------------------------------------------------------------

Multiple Port
--------------------------------------------------------------
PORT NUMBER   ENABLE TN3270E     RESOURCE NAME  DISABLE FILTERING
----------   --------------     -------------  -----------------
dlsb-ok TNE3270E config>
```

Table 60. Monitoring DLSw Connections and Sessions

```
(ctrl-p)
dlsb-ok *TALK 5

CGW Operator Console

dlsb-ok +PROTOCOL DLSW
Data Link Switching Console

dlsb-ok DLSw>LIST TCP SESSIONS
Group/Mcast@     IP Address  Conn State  CST Version ACTSes SesCreates
--------------------------------------------------------------------------
     1         9.24.104.203  ESTABLISHED  A  AIW V2RO    1       1

dlsb-ok DLSw>LIST DLSW SESSIONS ALL
Source     Destination   State   Flags   Dest IP Addr    Id
-----------------------------------------------------------
 1 APPN  04 40002216000a04 Connected     9.24.104.203     0
```

Table 61. Monitoring APPN Link

```
dlsb-ok APPN >LIST LINK_INFORMATION
   Name    Port Name  Intf   Adj CP Name   Type    HPR  State
================================================================
TNPUB1     DLS65     65534 USIBMRA.RA03M   NN   INACTIVE ACT_LS 1

```

1. This means APPN is active and in session.

Table 62. Display of LU-LU connection under TN3270E

```
dlsb-ok APPN >TN3270E
TN3270E GWCON
dlsb-ok TN3270E >LIST CONNECTIONS
Connection information for all the LUs
Local LU  Class  Assoc LU  Client Addr  Status  Prim LU  SEC LU  Idle Min
-------------------------------------------------------------------------
@01LU6     IW             9.24.106.44  LU-LU   RA03T03  LUB6       1
```

Table 63. Status of TN3270E Server

```
dlsb-ok TN3270E >LIST STATUS
TN3270E Server Status Summary

TN3270E IP Address: 172.16.220.2
NetDisp Advisor Port Number: 10008
   Keepalive type: None
   Automatic Logoff: N
   Client IP Address mapping: N
   Number of Connections                :1
   Number of Available LUA LU's         :5
   Number of LUA LU's pending termination  :0
   Number of defined LU's               :6
   Number of connections in SSCP-LU state  :0
   Number of connections in LU-LU state    :1
```

TN3270E LSA SNA Subarea Connection

When transporting SNA, you may configure the TN3270E Server using SNA subarea links in the same 2216. With this configuration, you will need the following definitions in the host:

- An XCA Major Node Definition

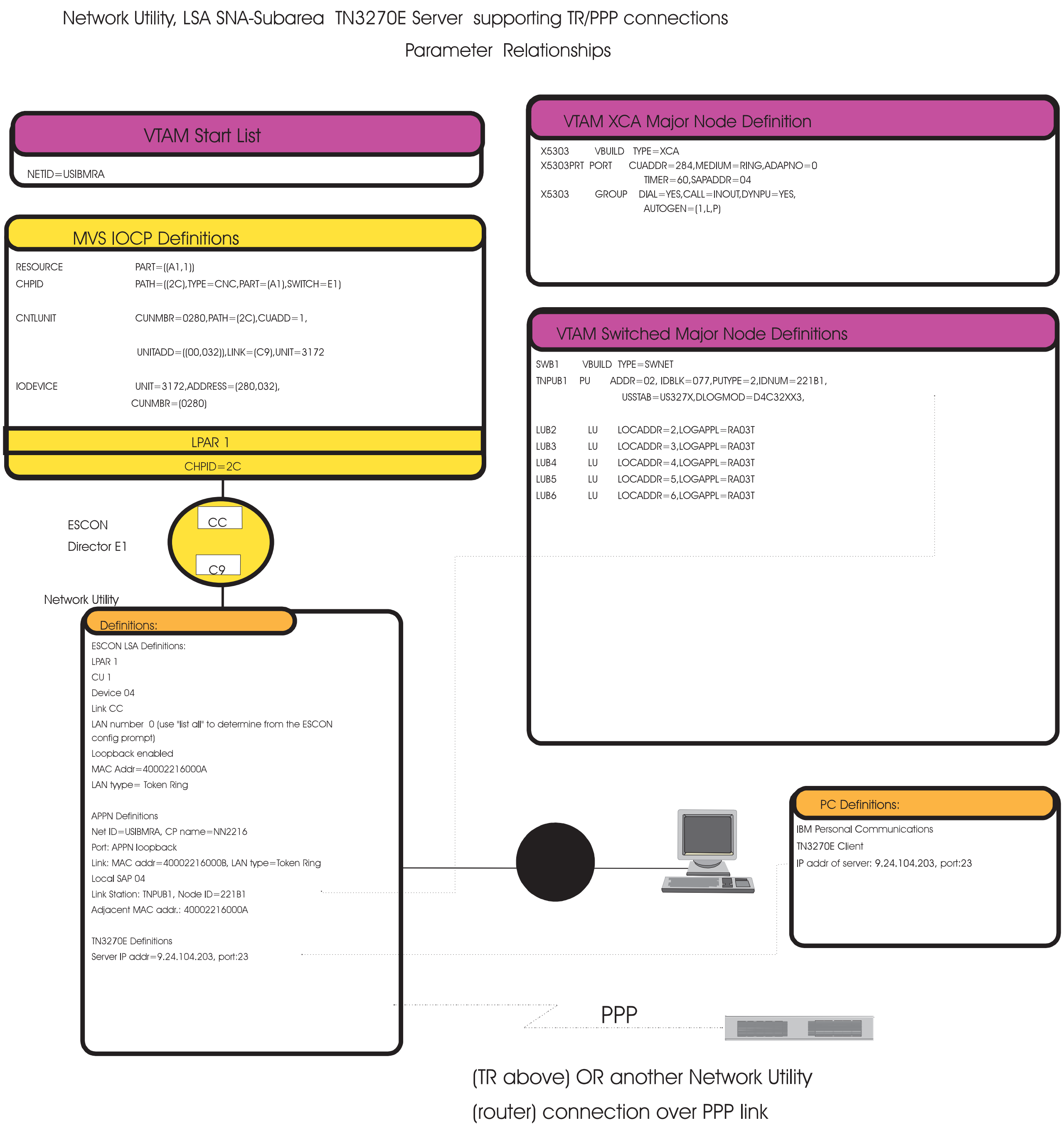
Table 64. VTAM XCA Switched Major Node Definition

```
X5303    VBUILD TYPE=XCA
X5303PRT PORT  ADAPNO=0,                                               *
               CUADDR=284,                                             *
               SAPADDR=4,                                              *
               MEDIUM=RING
X5303GRP GROUP DIAL=YES,CALL=INOUT,DYNPU=YES,                          *
               AUTOGEN=(1,L,P)
```

- A Switched Major Node Definition

Table 65. VTAM Switched Major Node: SWB1

```
SWB1     VBUILD TYPE=SWNET
TNPUB1   PU    ADDR=02,              X
               IDBLK=077,            X
               IDNUM=221B1,          X
               PUTYPE=2,             X
               USSTAB=US327X,        X
               DLOGMOD=D4C32XX3
LUB2     LU    LOCADDR=02,LOGAPPL=RA03T
LUB3     LU    LOCADDR=03,LOGAPPL=RA03T
LUB4     LU    LOCADDR=04,LOGAPPL=RA03T
LUB5     LU    LOCADDR=05,LOGAPPL=RA03T
LUB6     LU    LOCADDR=06,LOGAPPL=RA03T
```


Network Utility, LSA SNA-Subarea TN3270E Server supporting TR/PPP connections
Parameter Relationships
VTAM Start List
MVS IOCP Definitions
LPAR 1
CHPID=2C
ESCON
Director E1
CC
C9
Network Utility
Definitions:
VTAM XCA Major Node Definition
VTAM Switched Major Node Definitions
PC Definitions:
PPP
(TR above) OR another Network Utility
(router) connection over PPP link

```
NETID=USIBMRA
```

```
RESOURCE     PART=((A1,1))
CHPID        PATH=((2C),TYPE=CNC,PART=(A1),SWITCH=E1)

CNTLUNIT     CUNMBR=0280,PATH=(2C),CUADD=1,

              UNITADD=((00,032)),LINK=(C9),UNIT=3172

IODEVICE     UNIT=3172,ADDRESS=(280,032),
            CUNMBR=(0280)
```

```
ESCON LSA Definitions:
LPAR 1
CU 1
Device 04
Link CC
LAN number  0 (use "list all" to determine from the ESCON
config prompt)
Loopback enabled
MAC Addr=40002216000A
LAN tyype= Token Ring

APPN Definitions
Net ID=USIBMRA, CP name=NN2216
Port: APPN loopback
Link: MAC addr=40002216000B, LAN type=Token Ring
Local SAP 04
Link Station: TNPUB1, Node ID=221B1
Adjacent MAC addr.: 40002216000A

TN3270E Definitions
Server IP addr=9.24.104.203, port:23
```

```
X5303       VBUILD  TYPE=XCA
X5303PRT  PORT      CUADDR=284,MEDIUM=RING,ADAPNO=0
                      TIMER=60,SAPADDR=04
X5303       GROUP    DIAL=YES,CALL=INOUT,DYNPU=YES,
                      AUTOGEN=(1,L,P)
```

```
SWB1       VBUILD  TYPE=SWNET
TNPUB1    PU      ADDR=02, IDBLK=077,PUTYPE=2,IDNUM=221B1,
                    USSTAB=US327X,DLOGMOD=D4C32XX3,

LUB2        LU     LOCADDR=2,LOGAPPL=RA03T
LUB3        LU     LOCADDR=3,LOGAPPL=RA03T
LUB4        LU     LOCADDR=4,LOGAPPL=RA03T
LUB5        LU     LOCADDR=5,LOGAPPL=RA03T
LUB6        LU     LOCADDR=6,LOGAPPL=RA03T
```

```
IBM Personal Communications
TN3270E Client
IP addr of server: 9.24.104.203, port:23
```

Figure 27. LSA SNA-Subarea for TN3270E Supporting TR/PPP Connections

The Switched Major Node Definitions for the TN3270E Server PUs were defined as shown below.

Table 66. List of Escon Configuration

```
lsadirect ESCON Config>LIst
Net:  4   Protocol: APPN Loopback   LAN type: Token-Ring/802.5   1
          APPN loopback MAC address: 40002216000B
Net:  3   Protocol: LSA    LAN type: Token Ring      LAN number:  0
          Maxdata: 2052
          Loopback is enabled.
          MAC address: 40002216000A
          Block Timer:   10 ms   ACK length:   10 bytes
```

1. This is APPN Loopback network number (4). This number will be used later in definition: **APPN config>add port** as seen in Table 68 on page 201.

Now we can add APPN port, APPN link and TN3270E server definitions as seen in the following screens. If APPN and/or TN3270E packages are not loaded, refer to Table 53 on page 193.

Table 67. Basic APPN CP-Name Definition

```
2216 Config>protocol appn
2216 APPN config>set node
Enable APPN (Y)es (N)o [Y]?
Network ID (Max 8 characters) [  ]? usibmra
Control point name (Max 8 characters) [  ]? NN2216
Enable branch extender or border node
        (0=Neither, 1=Branch Extender, 2=Border Node) [0]?
Route addition resistance(0-255) [128]?
XID ID number for subarea connection (5 hex digits) [00000]?
Use enhanced #BATCH COS (Y)es (N)o [Y]?
Use enhanced #BATCHSC COS (Y)es (N)o [Y]?
Use enhanced #INTER COS (Y)es (N)o [Y]?
Use enhanced #INTERSC COS (Y)es (N)o [Y]?
Write this record? [Y]?
The record has been written.
2216 APPN config>ex
2216 Config>write
Config Save: Using bank A and config number 1
2216 Config>
2216 *reload y
```

Table 68. Adding APPN Port and Link Station

```
lsadirect APPN config>ADD/UPDATE PORT
APPN Port
Link Type: (P)PP, (FR)AME RELAY, (E)THERNET, (T)OKEN RING,
(M)PC, (S)DLC, (X)25, (FD)DI, (D)LSw, (A)TM, (I)P  [ ]? t
Interface number(Default 0): [0]? 4          1
Port name (Max 8 characters) [T00004]?
Enable APPN on this port (Y)es (N)o [Y]?
Port Definition
        Support multiple PU (Y)es (N)o [N]?
        Service any node: (Y)es (N)o [Y]?
        High performance routing: (Y)es (N)o [N]?
        Maximum BTU size (768-17745) [2048]?
        Maximum number of link stations (1-65535) [65535]?
        Percent of link stations reserved for incoming calls (0-100) [0]?
        Percent of link stations reserved for outgoing calls (0-100) [0]?
        Local SAP address (04-EC) [4]?
        Local HPR SAP address (04-EC) [C8]?
Edit TG Characteristics: (Y)es (N)o [N]?
Edit LLC Characteristics: (Y)es (N)o [N]?
Edit HPR defaults: (Y)es (N)o [N]?
Write this record? [Y]?
The record has been written.
lsadirect APPN config>
lsadirect APPN config>
lsadirect APPN config>
lsadirect APPN config>
lsadirect APPN config>ADD/UPDATE LINK-STATION
APPN Station
Port name for the link station [ ]? T00004
Station name (Max 8 characters) [ ]? tnpub1
        Activate link automatically (Y)es (N)o [Y]?
        MAC address of adjacent node [000000000000]? 40002216000A
        Solicit SSCP Session: (Y)es (N)o [N]? y
                Local Node ID (5 hex digits) [00000]? 221B1
                Enable Host Initiated Dynamic LU Definition : (Y)es (N)o [N]?
        Does link support APPN function: (Y)es (N)o [Y]? n
Edit TG Characteristics: (Y)es (N)o [N]?
Edit LLC Characteristics: (Y)es (N)o [N]?
Edit HPR defaults: (Y)es (N)o [N]?
Write this record? [Y]?
The record has been written.
```

1. This is the LSA Loopback interface number as seen in Table 66 on page 200.

Table 69. Definition of TN3270E

```
lsadirect APPN config>TN3270E
lsadirect TN3270E config>SET
TN3270E Server Parameters
        Enable TN3270E Server (Y/N) [Y]?
        TN3270E Server IP Address [9.24.104.203]?
        Port Number [23]?
        Enable Client Address Mapping (Y/N) [N]?
        Default Pool name (Max 8 characters) [PUBLIC]?
        NetDisp Advisor Port Number [10008]?
        Keepalive type:
         0 = none,
         1 = Timing Mark,
         2 = NOP [0]?
        Automatic Logoff (Y/N) [N]?
        Enable IP Precedence (Y/N) [N]?
Write this record? [Y]?
The record has been written.
lsadirect TN3270E config>ADD/UPDATE IMPLICIT-POOL
TN3270E Server Implicit Definitions
        Pool name (Max 8 characters) []?
        Station name (Max 8 characters) []?
Invalid name please re-enter
        Station name (Max 8 characters) []? tnpub1
        LU Name Mask (Max 5 characters) [@01LU]?
        LU Type  ( 1 - 3270 mod 2 display
                   2 - 3270 mod 3 display
                   3 - 3270 mod 4 display
                   4 - 3270 mod 5 display) [1]?
        Specify LU Address Ranges(s) (y/n) [N]?
        Number of Implicit LUs in Pool(1-253) [1]? 5
Write this record? [Y]?
```

Monitoring the Configuration

This configuration can be monitored with similar commands as seen in “Monitoring the TN3270E Over DLSw SNA Subarea Configuration” on page 195. The displayed results will also be similar.

Chapter 14. Channel Gateway

Overview

The Network Utility provides host connectivity through an ESCON channel or parallel channel. It enables the Network Utility to function as a gateway from the host to other networks.

Configurations Supported

There are three interfaces from host software to a Network Utility.

The first interface is the 8232-compatible support, called LAN Channel Station (LCS). This interface defines a number of commands for direct LAN connection and a blocking and deblocking structure. LAN-ready frames are transmitted from the host to the virtual LAN adapters and conversely. This interface is used by TCP/IP for VM and MVS and AIX/370.

The second interface is the Link Services Architecture (LSA) support, which is accessed in the host through VTAM.

The LSA support is a control interface to allow VTAM to use the Logical Link Control (LLC) portion of the Data Link Control (DLC) layer of the SNA stack. Included is access to LLC Type 1 (connectionless) and LLC Type 2 (connection oriented) data transport. This interface is used by VTAM for both SNA subarea and APPN ISR and HPR data transport.

The third interface is the Multi-Path Channel (MPC+) support, which is accessed in the host through VTAM. The MPC+ support is a protocol layer that allows multiple read and write subchannels to be treated as a single transmission group between the host and channel attached devices. This interface is used by OS/390 for APPN HPR, TCP/IP, and HPDT UDP data transport. Note that the Channel does not support MPC+ subchannel groups which are shared over more than one physical channel interface.

The Network Utility can support 64 ESCON subchannels, in any combination of LCS subchannel pairs, LSA subchannels, and MPC+ groups. This allows a maximum of 32 LCS virtual LAN adapters, or 32 LSA virtual LAN adapters, or 32 MPC+ groups (an MPC+ group must include at least one read subchannel and one write subchannel).

LSA and LCS virtual LAN adapters emulate a Token-Ring, FDDI, or Ethernet interface for communications with the host. This does not restrict the format of the remote network interface. It is intended only to maintain the existing host interfaces of the 3172 Interconnect Controller, to eliminate host support changes.

Each virtual LAN adapter or MPC+ group can support only one host connection type (LCS/LSA/MPC+). LSA and LCS subchannels can support multiple virtual LAN adapters, for example, one Token-Ring interface and one Ethernet interface. There is no perceived value to supporting multiple virtual LAN adapters of the same type on a single subchannel or pair, but configuration will not preclude it.

Host LAN Gateway Function

The host LAN gateway function allows host applications to communicate with LAN-based workstations. The two main host applications supported by the host LAN gateway function are TCP/IP and VTAM. These applications encapsulate LAN frames into channel control words (CCWs) for transport across the channel. This is also referred to as ″blocking″, because a CCW consists of a block of LAN frames sent as a single logical unit. The CCW is then ″deblocked″ by the receiver into individual frames.

Much of the Network Utility LAN gateway function is based on the 3172 Interconnect Controller Program (ICP). Even though there are differences in the 3172 ICP gateway function and the Network Utility Channel function, the hardware and software interfaces between the host and the Network Utility Channel are the same as the interfaces between the host and the 3172 ICP (except for the IP routing support provided within the Network Utility). To preserve the software interface, it is necessary for the Network Utility to create the appearance of a LAN adapter so that the host application still believes it is communicating with a real LAN.

ESCON Channel Concepts

Subchannels

The ESCON channel interface is divided into 256 logical addresses (inaccurately but consistently referred to as ″subchannels″ for historical reasons). Each host application interface uses one or more subchannels to connect the host application to the Network Utility. Through configuration, each subchannel is assigned a unique relative index, which may or may not match its actual logical address. The ESCON channel may be shared by multiple applications on multiple hosts, but each host application will have dedicated use of its subchannels. (This is not strictly true for MPC+, as explained later, but the statement applies at the MPC+ level; MPC+ subchannels cannot be shared with non-MPC+ applications.) The Network Utility supports up to 64 subchannels at a time.

Channel Protocols

Network Utility supports three channel protocols, corresponding to the three host software interfaces discussed above. Each protocol uses its subchannels differently, and a subchannel can support only one protocol at a time. The channel protocols supported are LAN Channel Station (LCS), Link Services Architecture (LSA) and Multi-Path Channel (MPC+).

LAN Channel Station (LCS): LCS is a channel protocol supported by TCP/IP applications in the host. Each application defines a consecutive pair of subchannels, one for TCP/IP to read from the channel, and one for TCP/IP to write to the channel. The LCS interface allows LAN MAC frames to be transported over the channel, and provides a command interface to activate, deactivate, and query the LAN interfaces. Each MAC frame has a header that identifies the virtual LAN adapter destination of the frame.

Link Services Architecture (LSA): LSA is an interface to support SNA traffic over the channel. Each LSA path is a single bidirectional subchannel between the host application and the Network Utility. The host software (VTAM) issues a read command immediately following each write command to retrieve data from the channel. The Network Utility also issues an Attention command when it has something for the host application to read. LSA has a command interface which allows VTAM to open Service Access Points (SAPs) to communicate with downstream workstations using the IEEE 802.2 logical link control (LLC) interface. The channel blocking/deblocking mechanism for LSA subchannels is the same as for LCS subchannel pairs.

Multi-Path Channel (MPC+): MPC+ is a data link control (DLC) interface for the channel. Each MPC+ path consists of one or more read subchannels and one or more write subchannels, bound together to form a transmission group. MPC+ transmission groups which span more than one physical ESCON channel are not supported in this release. VTAM and the Network Utility exchange XIDs to identify the number and direction of subchannels at initialization, and then each frame has a header to indicate the sending and receiving applications.

Blocks: The host channel interface packages control and data frames in blocks of up to 32 KB (36 KB for MPC+). The format of data blocks is different for MPC+ and non-MPC+ host applications. LSA and LCS blocks consist of one or more contiguous frames, each with a header that identifies the destination device by its LAN type and LAN number. MPC+ blocks contain one or more ″discontiguous″ frames, with the first 4 KB of the block containing MPC+ PDU headers and offsets of application data, which is stored in the last 32 KB of the block. MPC+ groups are identified by a ″LAN type″ and ″LAN number″ as well for implementation consistency.

A block of data is transmitted either when it is filled, or when the block delay timer (which determines how long the adapter waits for the block to fill before transmitting) expires. The process of receiving a block of data and forwarding the individual frames to the device driver is called ″deblocking.″

Virtual LAN Adapters: First, a little history: the 3172 Interconnect Control Program (on which the Network Utility is partially based) transferred frames from a host channel to one or more LANs. In its configuration, each subchannel was connected to one or more LAN device drivers. Data from the host was received by a deblocker, which would distribute the frames to one of the LAN adapters based on the LAN Type and LAN Number contained in the frame header. If a host application needed access to multiple LAN adapters, the configuration file would contain one entry for each LAN adapter.

In the Network Utility, instead of each subchannel being connected to one or more real LAN adapters, all of the subchannels are connected to the Base Net Handler, which is in turn connected to one or more virtual net handlers. Each virtual net handler supports one of the three channel protocols (LSA/LCS/MPC+) and sends and receives frames with one of the protocol applications (LLC/IP/APPN), which sends the data to another net handler representing a network connection. There may or may not be any real LAN adapters connected to the Network Utility.

To preserve the existing host interfaces, the Network Utility takes on the appearance of multiple LAN adapters for LSA and LCS connections. Based on configuration parameters, the Virtual net handlers register with the appropriate protocols as either Token-Ring, Ethernet, or FDDI adapters. The Base Net Handler allows the host to activate and deactivate this ″virtual LAN adapter″ in the same way it controls the 3172's real LAN adapters. Each virtual LAN adapter has its own MAC address, which allows the Network Utility to appear to the host as one or more LAN adapters on an actual local area network.

A single subchannel (or pair) can be connected to one or more virtual LAN adapters. This is necessary to allow a single host application to communicate with different types of LANs (Token-Ring, Ethernet, FDDI) over the same subchannel. LAN-bound frames are directed to the correct destination by the LAN Type and LAN Number in the frame header.

However, the inverse is true only for LSA connections. A single LCS virtual LAN adapter can be connected to only one subchannel. This restriction improves data throughput by allowing host-bound frames to be directed to the correct subchannel by the virtual net handler, without forcing the net handler to examine the MAC address or IP address of each host-bound frame. Multiple VTAMs can share a single LSA net handler if each opens a SAP with a unique number. This cannot be done for the LCS net handler because all TCP/IP traffic uses the multiprotocol SAP number 'AA'x. See Figure 28.

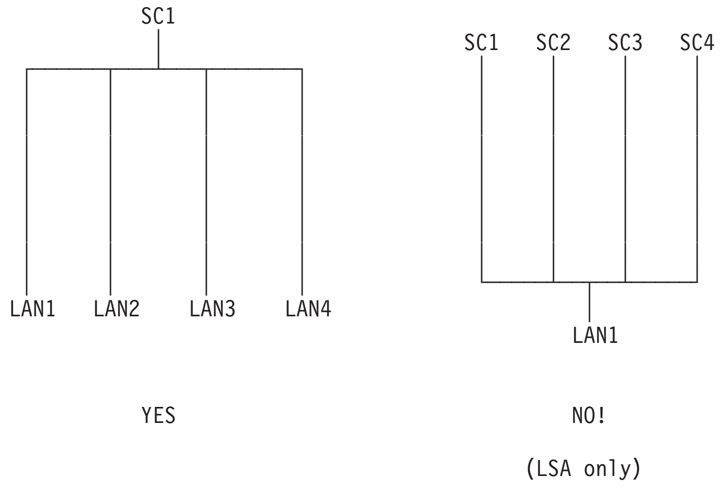


Figure 28. LAN-to-Subchannel Configuration

MPC+ Groups: MPC+ does not use the virtual LAN adapter concepts common to both LSA and LCS interfaces, because MPC+ does not support a LAN gateway appearance for the Network Utility. The equivalent interface for MPC+ is the MPC+ group. An MPC+ group is a set of ESCON subchannels configured to act as a single data pipe between the host and Network Utility. An MPC+ group consists of at least one ″read″ subchannel and at least one ″write″ subchannel. Any number of subchannels may be designated as read or write, and multiple MPC+ groups may be defined, subject to a maximum of 64 total subchannels per Network Utility.

Data may be sent over any or all of the active subchannels in an MPC+ group. The MPC+ endpoint is responsible for maintaining data order over a group. The number of subchannels is fixed when the MPC+ group is defined.

MPC+ groups are identified in the microcode using the same ″LAN type″ and ″LAN number″ designation as virtual LAN adapters. As frames are deblocked by the microcode, each frame is given a ″LAN type″ of MPC+ and a ″LAN number″ that corresponds to the MPC+ group associated with the subchannel it was received on. This allows the microcode and net handler to process MPC+ frames in a manner consistent with LSA and LCS frames.

LLC Loopback: LLC Loopback is an extension of the virtual LAN adapter concept to allow VTAM connections with the APPN and DLSw functions in the Network

Utility. To establish an SNA connection, the LSA interface creates an LLC connection between itself and the remote device across the LAN using IEEE 802.2 frames. See Figure 29.

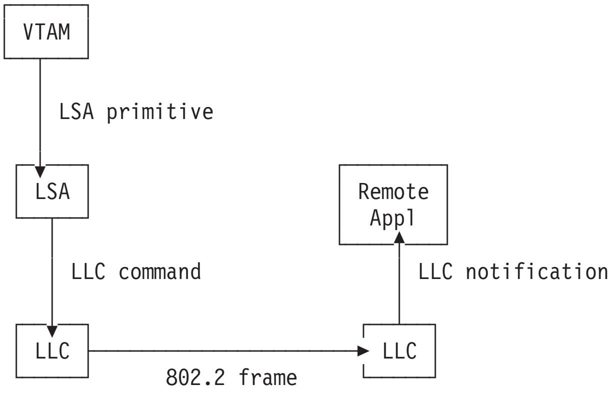


Figure 29. Normal LLC Connection

LLC Loopback allows the Network Utility to communicate directly with other LLC users (APPN and DLSw) in the Network Utility. Instead of turning LLC commands from LSA into 802.2 frames, they are converted into LLC notifications and sent to the appropriate LLC user. See Figure 30.

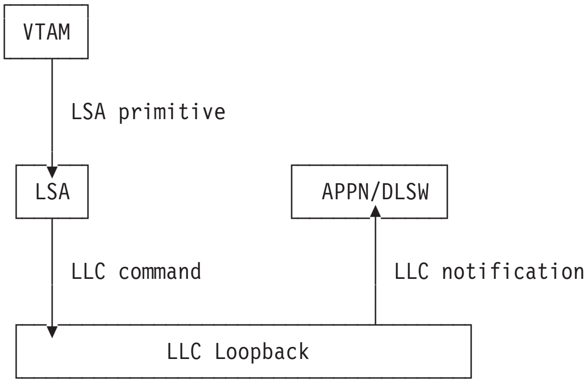


Figure 30. LLC Loopback Connection

LLC Loopback allows the APPN Network Node in Network Utility to act as the adjacent node to VTAM. It also permits VTAM to connect to remote devices and applications using Data Link Switching without requiring changes to VTAM’s LSA support, because the loopback connection appears the same as a normal LLC connection to VTAM.

Example Configurations

This section describes four sample configurations that use the Network Utility as a channel gateway to a mainframe system. Three of the samples show ESCON channel configurations and one shows a parallel channel. These configurations are:

- ESCON Channel Gateway (SNA and IP)
- Parallel Channel Gateway (SNA and IP)
- ESCON Channel Gateway (APPN and IP)
- ESCON Channel Gateway - High Availability

All of these configurations can be built using either the Network Utility model TN1 or TX1. You do not need the extra function provided by the model TN1 unless you are planning to configure the TN3270E server function in the same machine.

ESCON Channel Gateway

This scenario is shown in Figure 31. The Network Utility is configured to support both SNA and IP traffic into the host from both remote sites and LAN segments at the main site. The ESCON channel adapter is configured with an LSA direct interface to transport the SNA traffic and an LCS interface to perform IP forwarding.

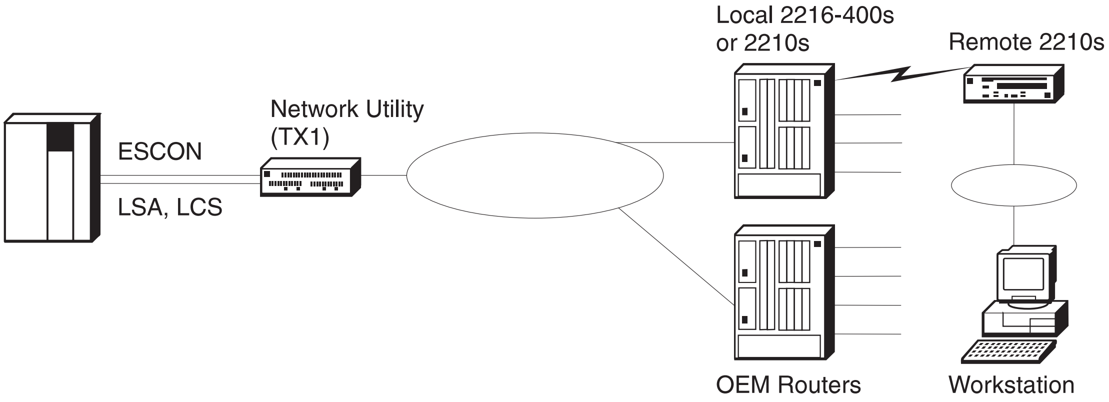


Figure 31. ESCON Channel Gateway

Keys to Configuration

The subchannel definitions for both the LCS and the LSA interfaces must match parameters used in the host to define the Network Utility to the host channel subsystem. The key subchannel parameters to configure at the Network Utility are shown in Table 70 on page 209.

Table 70. Network Utility Subchannel Configuration Parameters

Command	Description
device	The unit address transmitted on the channel path to select the Network Utility. It is also referred to as subchannel number in S/370 I/O architecture. It is a two-digit hexadecimal value in the range 00 to FF. This value is defined in the host Input/Output Configuration Program (IOCP) by the UNITADD statement on the CNTLUNIT macro instruction for the real device. **Valid Values:** X'00' to X'FF' **Default:** None
cu	The Control Unit address defined in the host for the Network Utility. This value is defined in the host IOCP by the CUADD statement on the CNTLUNIT macro instruction. **Valid Values:** X'0' to X'F' **Default:** X'0'
link	This parameter is significant when an IBM 9032 ESCON Director (ESCD) is used between the Network Utility and the host. When an ESCD is used, the link address is the port number of the ESCON Director (ESCD) to which the *host* is attached. If two ESCDs are in the path, it is the host-side port number of the ESCD defined with the dynamic connection. When no ESCD is in the communication path, this value must be set to X'01'. **Valid Values:** X'01' to X'FE' **Default:** X'01'
lpar	Logical partition number. This allows multiple logical host partitions to share one ESCON fiber. This value is defined in the host IOCP by the RESOURCE macro instruction. If the host is not using ESCON Multiple Image Facility (EMIF), use the default of 0 for the LPAR number. **Valid Values:** X'0' to X'F' **Default:** X'0'

LPAR and CU Parameters: When defining an LSA, LCS, or MPC+ interface on the Network Utility, you need to specify correct values for the CU and the LPAR parameters.

Notes on the CU Parameter:

The value for the CU needs to be set if you have multiple LPARs or multiple MVS or OS/390 images that need to access the Network Utility. If so, then you will need to create an interface definition (LSA, LCS, or MPC+) for each LPAR and each will use a different value for the CU parameter.

Further, the value of the CU parameter should match the CUADD parameter in the CNTLUNIT macro of the IOCP definition.

Previously, whenever a new LPAR (partition) was configured, a unique CU number had to be configured with it. With PTF01, the CU number and LPAR are independent for ESCON. You do not need a unique CU number for each LPAR number. this greatly increases user configuration flexibility and simplifies operation in large host systems.

Notes on the LPAR Parameter:

The first issue is whether the host is partitioned into multiple logical partitions (LPARs). If it is not, then the LPAR parameter will be zero.

If it is, then you will need a RESOURCE macro in the host Input/Output Configuration Program (IOCP) definitions that specify each partition by name and assign a numeric value to each. This numeric value is used when configuring the Network Utility for the LPAR parameter.

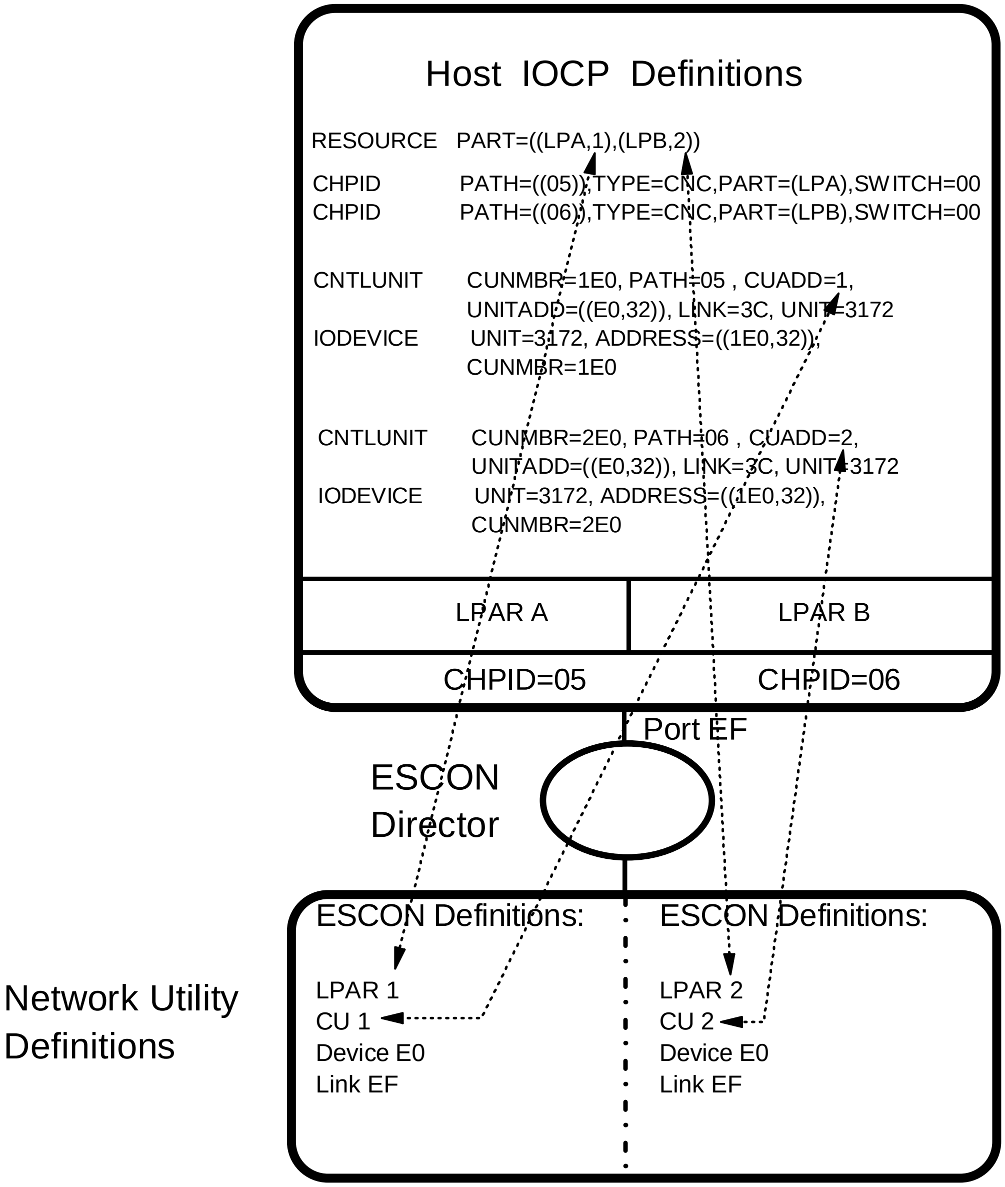
The second issue is whether the channel path identifiers (CHPIDs) are shared between one or more LPARs[19].

If you are not using shared channels (or you do not have EMIF), then the value for the LPAR parameter will be 0.

The maximum number of LPARs per ESCON adapter has increased from 32 to 64. In order to support this, we have increased the maximum number of subchannels per adapter from 32 to 64 and increased the maximum number of virtual nets per adapter from 16 to 32. This is a benefit for LSA protocol users who need to configure more than 32 LPARs.

Figure 32 on page 211 shows an example where the host is partitioned but the channel paths are not shared between the LPARs.

[19] You need EMIF to share channels between LPARs.



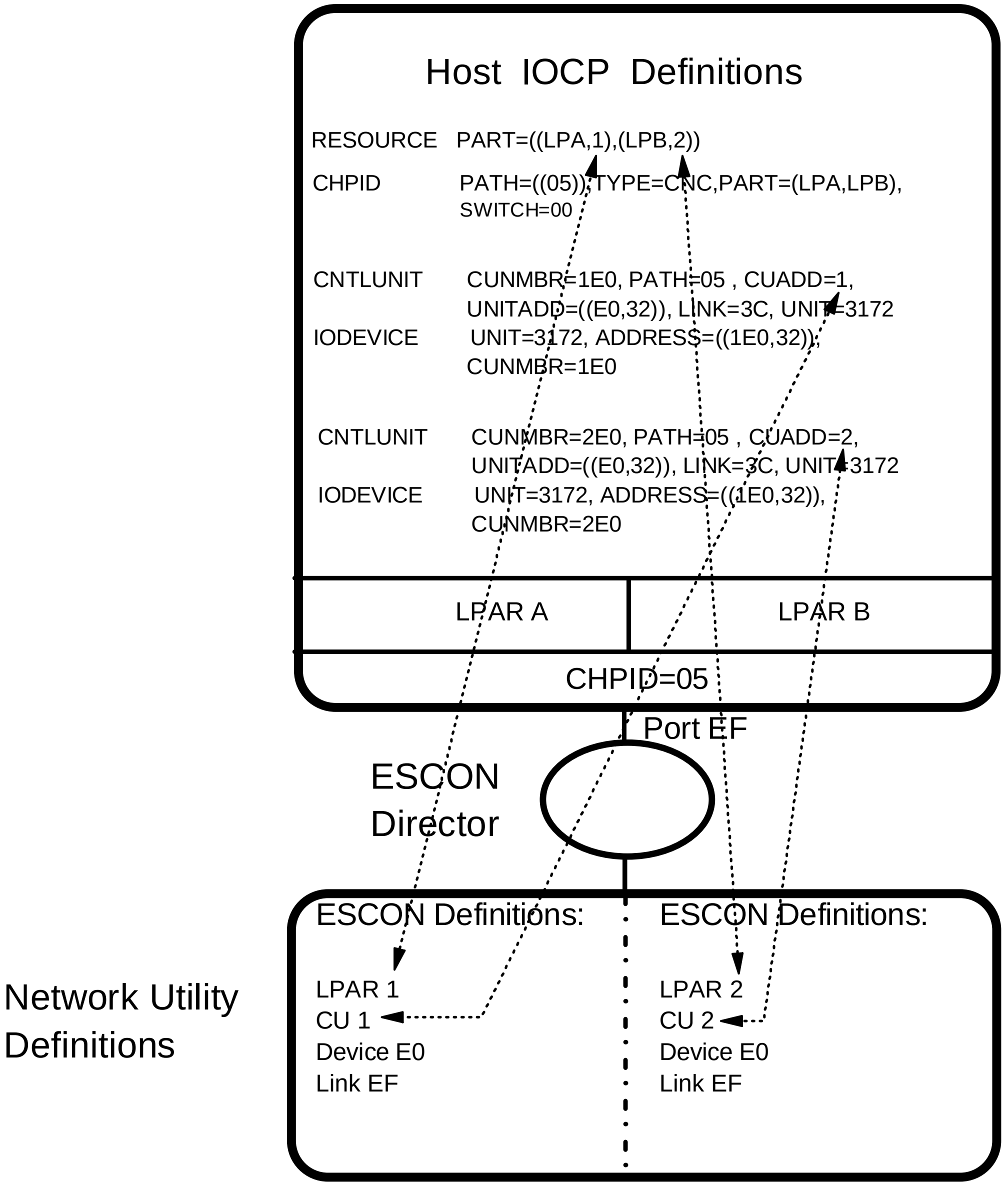
```
RESOURCE   PART=((LPA,1),(LPB,2))
CHPID          PATH=((05)),TYPE=CNC,PART=(LPA),SWITCH=00
CHPID          PATH=((06)),TYPE=CNC,PART=(LPB),SWITCH=00

CNTLUNIT       CUNMBR=1E0, PATH=05 , CUADD=1,
               UNITADD=((E0,32)), LINK=3C, UNIT=3172
IODEVICE       UNIT=3172, ADDRESS=((1E0,32)),
               CUNMBR=1E0

CNTLUNIT       CUNMBR=2E0, PATH=06 , CUADD=2,
               UNITADD=((E0,32)), LINK=3C, UNIT=3172
IODEVICE       UNIT=3172, ADDRESS=((1E0,32)),
               CUNMBR=2E0
```

Figure 32. Host/Network Utility Parameter Relationships (Nonshared CHPIDs)

If you are using EMIF on the host, then multiple LPARs can share the same CHPID to the Network Utility. In this case, you will still need two interfaces defined on the Network Utility and each will have a different value specified for the CU parameter. The other parameters can use the same values. Figure 33 on page 212 shows an example where the host is partitioned and EMIF is used to allow both partitions to use the same CHPID.



```
RESOURCE   PART=((LPA,1),(LPB,2))
CHPID          PATH=((05)),TYPE=CNC,PART=(LPA,LPB),
               SWITCH=00

CNTLUNIT       CUNMBR=1E0, PATH=05 , CUADD=1,
               UNITADD=((E0,32)), LINK=3C, UNIT=3172
IODEVICE        UNIT=3172, ADDRESS=((1E0,32)),
               CUNMBR=1E0

CNTLUNIT       CUNMBR=2E0, PATH=05 , CUADD=2,
               UNITADD=((E0,32)), LINK=3C, UNIT=3172
IODEVICE        UNIT=3172, ADDRESS=((1E0,32)),
               CUNMBR=2E0
```

Figure 33. Host/Network Utility Parameter Relationships (Shared Channels)

The LSA Direct Interface: Figure 34 on page 213 shows how the configuration parameters for the Network Utility correlate to the host parameters for an LSA interface definition.

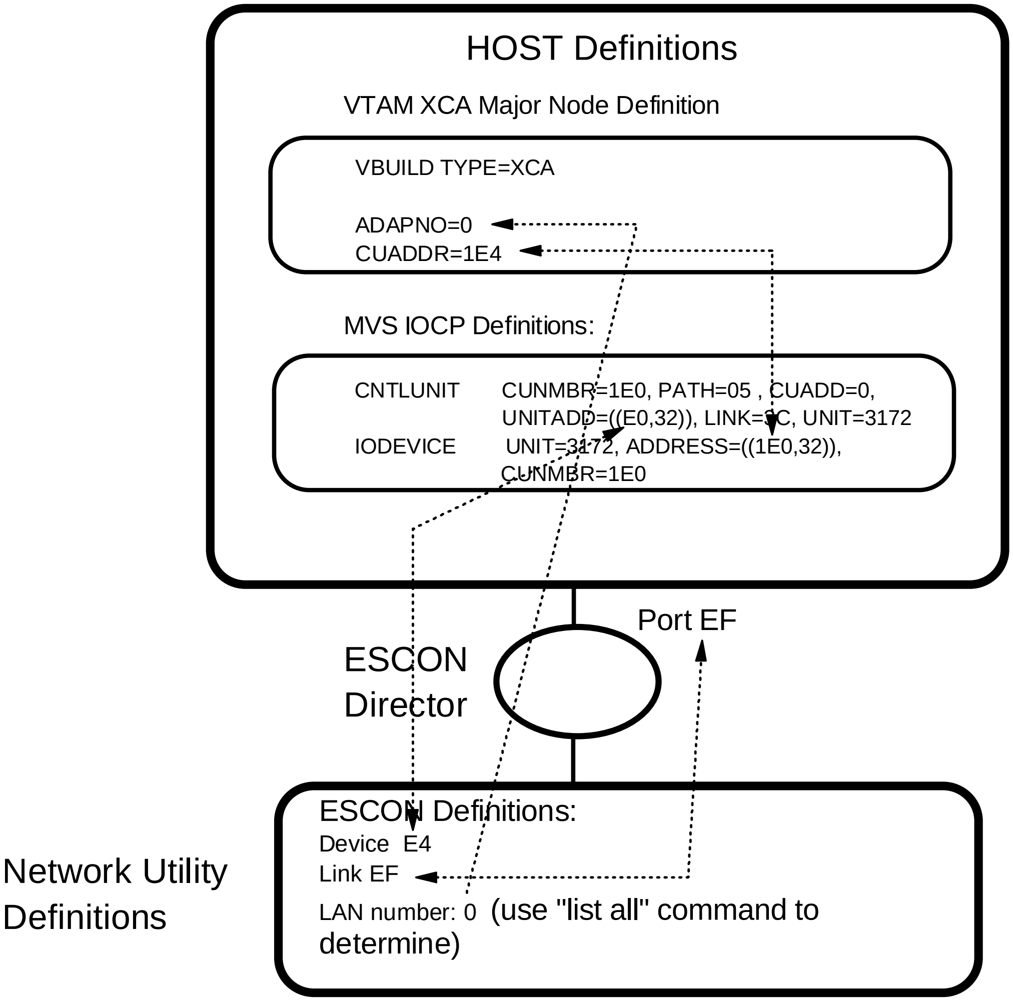

HOST Definitions
VTAM XCA Major Node Definition
VBUILD TYPE=XCA
ADAPNO=0
CUADDR=1E4
MVS IOCP Definitions:
CNTLUNIT CUNMBR=1E0, PATH=05 , CUADD=0,
UNITADD=((E0,32)), LINK=SC, UNIT=3172
IODEVICE UNIT=3172, ADDRESS=((1E0,32)),
CUNMBR=1E0
Port EF
ESCON
Director
ESCON Definitions:
Device E4
Link EF
LAN number: 0 (use "list all" command to
determine)
Network Utility
Definitions

Figure 34. Host/Network Utility Parameter Relationships - LSA

Notes:

1. LSA uses a single bidirectional subchannel between the host and the Network Utility. VTAM issues a read command immediately following each write command to retrieve data from the channel.
2. The device address specified in the Network Utility LSA interface definition must be within the range specified in the UNITADD parameter in the CNTLUNIT macro from the IOCP. For example, the UNITADD parameter in Figure 34 shows that 32 (decimal) device addresses starting at E0 (hex) are being reserved for the Network Utility definition. A device address E4 has been specified for the Network Utility LSA interface. Because E4 is in the range between E0 and FF hex, this is OK as long as no other device (or interface on this Network Utility) tries to use that subchannel.
3. The value specified in the CUADDR parameter in the VTAM XCA major node definition must be within the range specified in the ADDRESS parameter in the IODEVICE macro from the IOCP. For example, the CUADDR parameter in the XCA major node definition in Figure 34 is 1E4 hex, which is in the range 1E0 to 1FF that the ADDRESS parameter in the IODEVICE statement specifies.
4. The values specified for the ADDRESS parameter in the IODEVICE macro and the UNITADD parameter in the CNTLUNIT macro are related ***by convention only***. In this example, the value for the ADDRESS parameter has been

determined from the value for the UNITADD parameter by prepending a ***logical channel identifier*** (1 in this case) to the UNITADD value. This will often be the case. However, when defining the device address on the Network Utility LSA definition, use the UNITADD parameter and not the ADDRESS parameter to determine the valid range of values.

5. When you define an LSA direct interface on the Network Utility, you associate the interface with one of the LAN interfaces on the Network Utility. In effect, this puts the LSA direct interface on this same LAN segment. Every frame with a destination address of the MAC address of the Network Utility adapter on this LAN segment automatically gets forwarded over the channel to the host.

See “Chapter 18. Sample Host Definitions” on page 259 for more explanation and samples of host definitions for this interface type.

For a complete look at the configuration parameters needed for this scenario, see Table 19 on page 152.

The LCS interface: Defining an LCS interface creates a virtual LAN inside the Network Utility. There are two IP stations on this LAN: the Network Utility and the host. This LAN must be a unique IP subnet in the network. A MAC address is also needed for the LCS interface. After you create the LCS interface, do not forget to assign the IP address to this interface.

Network Utility provides three ways of operating the LCS interface:

1. LCS routing

   The LCS support described above and documented in the example configurations is the initial 2216 LCS support released in MAS V1R1.1. This type of LCS support passes host IP traffic to the IP routing function within the Network Utility. If you are replacing a 3172 with a Network Utility configured with this type of LCS support, you need to configure an additional IP subnet for the virtual LAN segment inside the Network Utility.

2. LCS bridging

   MAS V3.2 introduces ″LCS Bridging″ (officially called ″TCP/IP Passthru″), to enable 3172 replacement with no changes to the IP topology of the network. In this mode, the Network Utility simply bridges IP traffic between an LCS bridge port and other configured bridge ports. No IP routing is performed as frames are transferred from one port to another. To enable this mode, you do not specify an IP address for the LCS interface, but you do define a MAC address and enable bridging on it. See the MAS V3.2 *Software User’s Guide* for more information on configuring this function.

3. LCS 3172 emulation

   MAS V3.2 PTF01 makes available a third type of LCS support, which can be called ″3172 Emulation″. This LCS mode mirrors 3172 behavior exactly by mapping an LCS virtual interface to a single LAN interface. Unlike LCS Bridging, where multiple paths exist between the various bridge-enabled interfaces, LCS Passthrough sets up independent fixed paths between specific subchannels and specific LAN adapters. Traffic on one path cannot be seen anywhere else. To enable this mode, you enable the 3172 emulation for this mode, you do not specify an IP address, and you reference a specific LAN adapter instead of defining an LCS MAC address by using the ″NET″ parameter when defining the LCS. By doing this, you pick up the LANtype and the MAC address of the LAN you are attaching to.

This channel gateway function allows the Network Utility to function as a drop-in 3172 replacement in TCP/IP networks. Frames received from a TCP/IP host are passed directly to a downstream LAN adapter, bypassing the IP router and bridging functions of the Network Utility. IP and ARP frames received by a LAN adapter associated with the LCS Passthrough function are passed directly to LCS for delivery to the TCP/IP host. The Network Utility replaces the 3172 LCS function without requiring changes in IP network topology or adding additional bridge hops, as previous LCS methods did.

Beginning with V3.2 PTF01, you can dynamically add a new ESCON virtual interface (LSA, MPC +, or LCS) using an LPAR that is not currently configured. Previously, a dynamically added net could only be configured with subchannels on an already configured LPAR. Adding an interface with a new LPAR logical path required disabling the entire physical channel interface. To use the new support, you configure spare interfaces, add the new virtual interface using “talk 6”, and activate the new net using “talk 5”.

Parallel Channel Gateway

This scenario is shown in Figure 35. It is identical to the ESCON channel gateway except that the connection to the host is via a S/370 Bus and Tag (Parallel Channel) Adapter instead of an ESCON channel. Like the ESCON gateway, this configuration uses an LSA direct connection for the SNA traffic and an LCS interface for the IP traffic.

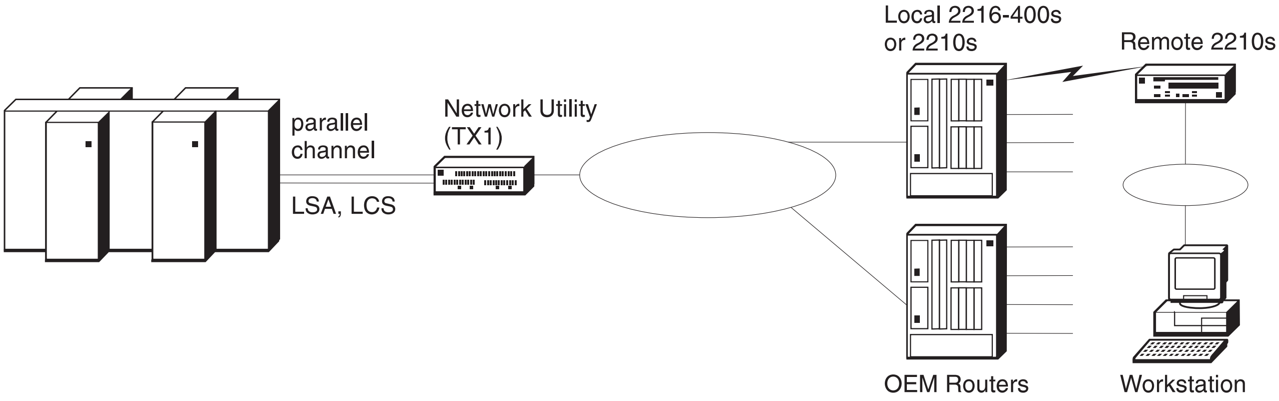


Figure 35. Parallel Channel Gateway

Keys to Configuration

The configuration for this scenario is very similar to that for the ESCON gateway (see “ESCON Channel Gateway” on page 208). The configuration of the LSA and LCS interfaces require fewer parameters because no LPAR, Link Address, or Control Unit values are required for a bus and tag connection. The device address is still required to identify the Network Utility on the channel.

For a complete look at the configuration parameters needed for this scenario, see Figure 8 on page 135. Also, “Chapter 18. Sample Host Definitions” on page 259 contains a sample of the host IOCP definition for a Network Utility with a Parallel Channel Adapter.

Channel Gateway (APPN and IP over MPC+)

This scenario is shown in Figure 36. Here, a Multi-Path Channel (MPC+) Group is used to transport both IP and APPN traffic between the Network Utility and the host. MPC+ uses a group of ESCON subchannels to maximize data transfer performance.

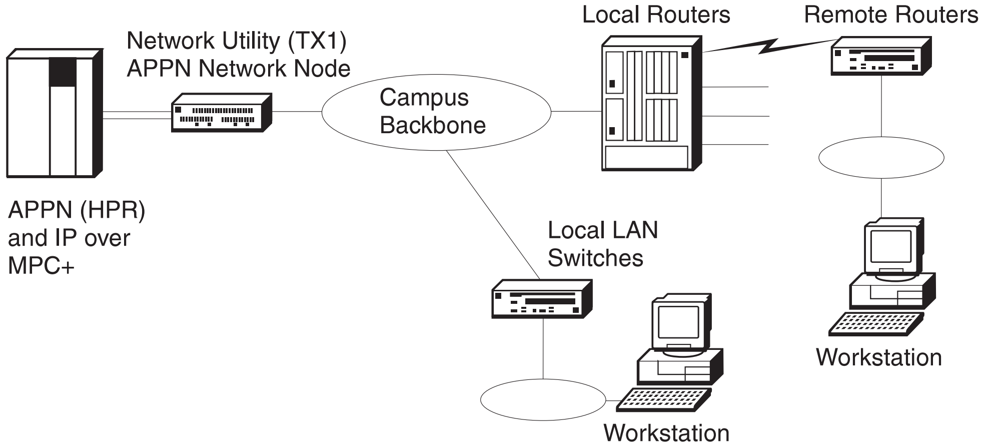


Figure 36. Channel Gateway (APPN and IP)

The APPN traffic coming through the Network Utility is comprised of several different types from the routers in the remote branches:

- TN3270 traffic from TN3270E servers in the branches that are configured with an APPN connection to the host. (See “Distributed TN3270E Server” on page 140 for an example of this type of configuration.)
- DLUR traffic from the routers in the branches that are providing support for PU 2.0 (dependent) devices.
- APPN host-to-host traffic from distributed processors (such as AS/400 processors) communicating with the mainframe at the central site.

In each of the above cases, the Network Utility is providing ANR forwarding only of the APPN traffic. [20] However, in addition to providing the ANR function, the Network Utility in this scenario could also be configured for TN3270E server support and DLUR support. The DLUR support could provide PU 2.0 devices on the local campus with access to the host and the TN3270E server could provide TN3270 support for workstations and printers on the local campus or for branches that do not have a distributed TN3270E server.

Keys to Configuration

Note the following when configuring the Network Utility for this scenario:

- You can either define a separate MPC+ group for your APPN and TCP/IP traffic or you can define a single group that is shared between APPN and TCP/IP.
- An MPC+ group can have as many as 64 subchannels in it. It must have at least one read and one write subchannel defined. From the talk 6 command line (from the `ESCON Add Virtual` prompt), the **sub addr** command is used to add a read subchannel while the **sub addw** command is used to add a write subchannel.

[20] The RTP sessions are between the APPN nodes at each end of the conversations.

- TCP/IP is configured on an MPC+ interface the same way it is for other interfaces. Specifically, configuring an IP address for the MPC+ virtual net handler enables TCP/IP over the MPC+ interface.
- APPN is configured over the MPC+ connection the same way that it is configured for other interfaces. When you use the **add port** command, specify a port type of **M** for MPC+.
- To run APPN / HPR traffic over a MPC+ Channel, two VTAM definitions need to be created:
  - A Transport Resource List (TRL) element that defines the line control, the subchannels, the number of buffers, and the channel programs to be used
  - A local SNA major node with a local PU definition
- Like the LSA and LCS definitions, the subchannel parameters must match parameters used in the host definitions when defining the Network Utility to the host channel subsystem. See Table 70 on page 209 for a description of the subchannel parameters and Figure 37 for a diagram of how these parameters correlate to the host parameters for an MPC+ definition.

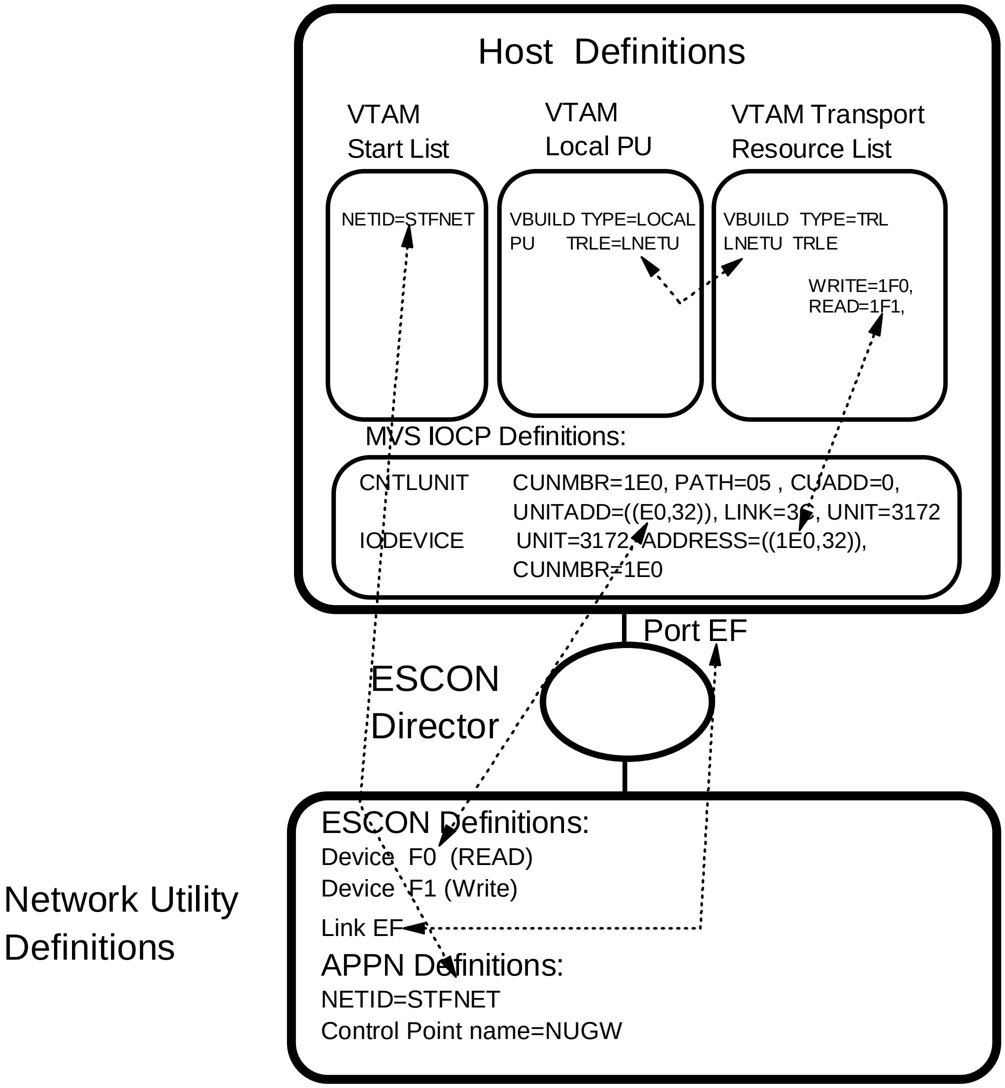


Figure 37. Host/Network Utility Parameter Relationships - MPC+

Notes:

1. The device addresses specified in the Network Utility MPC+ interface definition must be within the range specified in the UNITADD parameter in the CNTLUNIT macro from the IOCP. For example, the UNITADD parameter in Figure 37 on page 217 shows that 32 (decimal) device addresses starting at E0 (hex) are being reserved for the Network Utility definition. Device addresses F0 and F1 have been specified for the Network Utility MPC+ interface. Because F0 and F1 are in the range E0 to FF hex, this is OK as long as no other device (or interface on this Network Utility) tries to use these same subchannels.
2. The values specified in the VTAM TRL major node definition must be within the range specified in the ADDRESS parameter in the IODEVICE macro from the IOCP. For example, the TRL major node definition in Figure 37 on page 217 specifies 1F0 and 1F1, which are in the range 1E0 to 1FF that the ADDRESS parameter in the IODEVICE statement specifies.
3. The values specified for the ADDRESS parameter in the IODEVICE macro and the UNITADD parameter in the CNTLUNIT macro are related ***by convention only***. In this example, the value for the ADDRESS parameter has been determined from the value for the UNITADD parameter by prepending a ***logical channel identifier*** (1 in this case) to the UNITADD value. This will often be the case. However, when defining device addresses on the Network Utility MPC+ definition, use the UNITADD parameter and not the ADDRESS parameter to determine the valid range of values.

See “Chapter 18. Sample Host Definitions” on page 259 for examples of these host definitions.

Dynamic Routing Protocols on the ESCON Interface

In a single host environment it is not necessary to run a routing protocol (RIP, for example) on the ESCON subnet. In this case, it is sufficient to add the Network Utility as the default gateway in the host TCP/IP profile.

However, if there are multiple hosts or multiple Network Utility gateways, you should consider running RIP on the ESCON interface. Running a dynamic routing protocol in this environment allows you to route around network failures if an alternate path exists.

Network Utility supports both RIP V1 and V2. RIP V2 offers variable length subnets and other advanced features that RIP V1 does not, and is the recommended choice.

Importing the ESCON Subnet into OSPF

If you are running OSPF on your network, then you should import the ESCON subnet into OSPF (unless your host TCP/IP supports OSPF). If this is not done, only workstations connected directly to an interface on the Network Utility will be able to access the TCP/IP host on the ESCON interface.

For a complete look at the configuration parameters needed for this scenario, see Figure 13 on page 139.

ESCON Channel Gateway - High Availability

This scenario is shown in Figure 38. It utilizes redundant Network Utilities, each with an ESCON channel connection to the host. Also, the campus backbones have been duplexed and each Network Utility attaches to a different backbone.

With this configuration, you can still access the host even if you have a failure in one of the campus backbones or a Network Utility. The traffic coming in from the 2216s will still have a valid path to the host through one campus backbone and Network Utility. This is true for both IP and SNA traffic.

The ESCON Director (ESCD) is important in this configuration, especially in Parallel Sysplex environments, because it allows you to fully mesh the connections between the gateways and the LPARs in the sysplex. This provides the highest level of fault tolerance for host access.

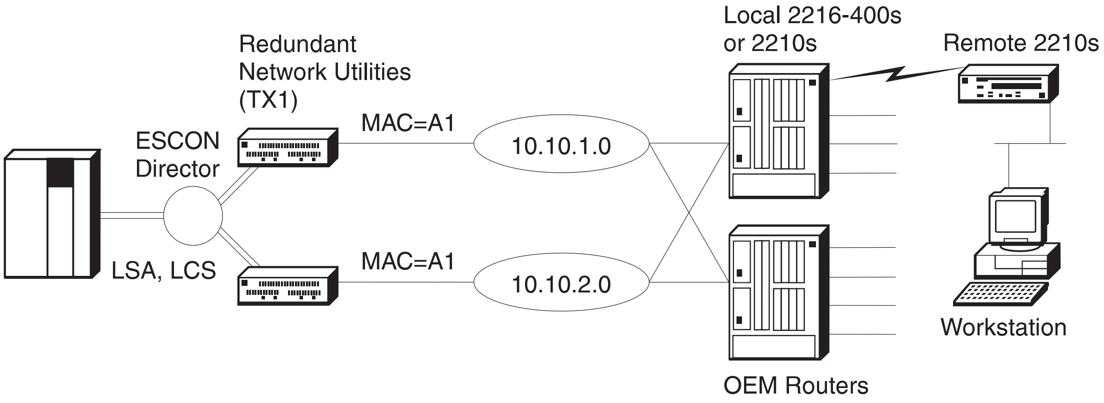


Figure 38. ESCON Channel Gateway - High Availability

Keys to Configuration

The configuration for this scenario is very much like the one in “ESCON Channel Gateway” on page 208. Each Network Utility is configured as a LAN Channel Gateway with a separate LSA and LCS interface defined on each. See Table 19 on page 152 for the parameters needed for configuring a Network Utility as a LAN Channel gateway.

Because each Network Utility is on a different Token Ring, the same MAC address can be used for the Token-Ring interface in each. The IP address used for each interface, however, must be different because each interface is on a different subnet.

Note: While this example shows the use of LSA and LCS connections on the ESCON channel, the use of MPC+ is equally effective in the high-availability environment.

Managing the Gateway Function

The configuration examples in this chapter and in “DLSw LAN Channel Gateway” on page 242 show several different uses of channel DLCs:

- A direct LSA interface maps to a LAN interface with no involvement from DLSw or APPN in forwarding frames.
- An LCS Routing or MPC+ virtual interface appears to the IP routing code as another interface, and IP performs its normal routing function to forward frames to other interfaces.

  An LCS Bridging interface appears to the bridge code as another LAN bridge port, and bridging performs its normal function to forward frames to other ports.
- The loopback LSA virtual interface appears as a link to either DLSw or APPN.
- An MPC+ virtual interface can appear as a link to APPN.

To manage the complete range of Network Utility gateway function, you need to manage IP, bridging, DLSw, and APPN as appropriate. This section does not cover these upper-layer functions, but focuses instead on the ways you can monitor and manage channel physical and virtual interfaces.

Command-Line Monitoring

You access the talk 5 commands that show the status of channel resources hierarchically as follows:

1. From the * prompt, type **talk 5** and press **Enter** to reach the + prompt.
2. From the + prompt, type **int** and press **Enter** and note the logical interface number for the physical ESCON or PCA interface you are interested in.

   The physical interface is commonly called the *base net*, and may have a number of LSA, LCS, or MPC+ virtual interfaces defined on top of it. The base net and all virtual interfaces each have a different logical interface number.
3. From the + prompt, type **net** *base n number* and press **Enter** to reach the ESCON or PCA Console subprocess. The command prompt changes to `ESCON>` or `PCA>` as appropriate.

   At these prompts, you can use the **li nets** command to see the current state of every (LSA, LCS, MPC+) virtual interface using this base net. You can also type **li sub** to view the currently running subchannel configuration for this base net.
4. From the base net `ESCON>` or `PCA>` prompt, type **net** *virtual net number* and press **Enter** to see more detail on a particular virtual interface that uses this base net. The command prompt changes to `LSA>`, `LCS>`, or `MPC+>`, depending on the type of the virtual interface you select.

   Each of these prompts supports a **list** command, to show configuration and current status information relevant to the virtual interface type.
5. To back out from any of these nested levels, type **exit** and press **Ctrl-p** to go back to the * prompt.

For examples and a detailed explanation of the output of these commands, see the chapter ″Configuring and Monitoring the ESCON and Parallel Channel Adapters″ in the *MAS Software User’s Guide*.

Event Logging Support

Events occurring within the channel functions are covered by the following ELS subsystems:

ESC Low-layer ESCON events

PCA Low-layer parallel channel events

LSA Events related to LSA virtual interfaces

LCS Events related to LCS virtual interfaces

MPC+ Events related to MPC+ virtual interfaces

To enable event logging, type **event** from talk 5 or talk 6 to reach the ELS Console or Config subprocess. If you want the logging output to go to talk 2, type **disp sub** *subsystem name* and press **Enter** to enable normal error reporting, or **disp sub** *subsystem name* **all** to enable all messages. To get the greatest visibility to a problem, you might enable both one of the ESCON or PCA subsystems, and one of the virtual interface subsystems. If you use these commands from talk 5, you can immediately move to talk 2 and monitor events as they occur.

You can get a feel for the events reported by each of these subsystems using the command **li sub** *subsystem name* from either the talk 5 or talk 6 ELS subprocess.

SNA Management Support

From a VTAM or NetView/390 operator console, you can control SNA resources associated with LSA direct gateway function, DLSw, or APPN as described in “NetView/390” on page 99.

The channel function itself does not send SNA alerts. It does not send traps that can be converted to alerts, but you can enable traps for channel ELS messages and use the products mentioned in “IBM Nways Manager for AIX” on page 96 to convert those traps to alerts.

SNMP MIB and Trap Support

Network Utility supports an IBM enterprise-specific MIB for ESCON. This MIB provides access to the following information:

- A list of physical interfaces and the fiber signal status of each
- A list of channel links and the host connection status of each
- A list of channel stations with both configuration and normal/error traffic statistics for each.

The ESCON MIB does not define any traps. Parallel channel functions have no MIB support.

Both ESCON and parallel channel interfaces are represented in the Interfaces MIB (RFC 1573), so a management station can access their status and basic per-interface traffic statistics. Network Utility allows a management station to control interface state, and can send traps to report when the interfaces go up or down.

Network Management Application Support

The Network Utility Java-based application discussed in “IBM Nways Manager Products” on page 96 provides integrated support for both the ESCON MIB and the Interfaces MIB. You can see color-coded interface status as well as specific panels that present key information from these MIBs. You can also use integrated browser support to view the information in either of these MIBs.

You can disable or enable the emission of interface up/down traps from the Nways Manager products.

Chapter 15. Channel Gateway Example Configuration Details

This chapter contains diagrams and configuration parameter tables for several of the example channel gateway network configurations in “Chapter 14. Channel Gateway” on page 203. The parameter values shown are from real working test configurations.

For an explanation of the columns and conventions in the configuration parameter tables, see “Example Configuration Table Conventions” on page 125.

The Network Utility World Wide Web pages contain binary configuration files that match these configuration parameter tables. To access these files, follow the Download link from:

```
http://www.networking.ibm.com/networkutility
```

The configurations documented in this chapter are:

Table 71. Cross-Reference of Example Configuration Information

Configuration Description	Parameter Table
“ESCON Channel Gateway” on page 208	Table 72 on page 224
“Parallel Channel Gateway” on page 215	Table 73 on page 228
“Channel Gateway (APPN and IP over MPC+)” on page 216	Table 74 on page 234

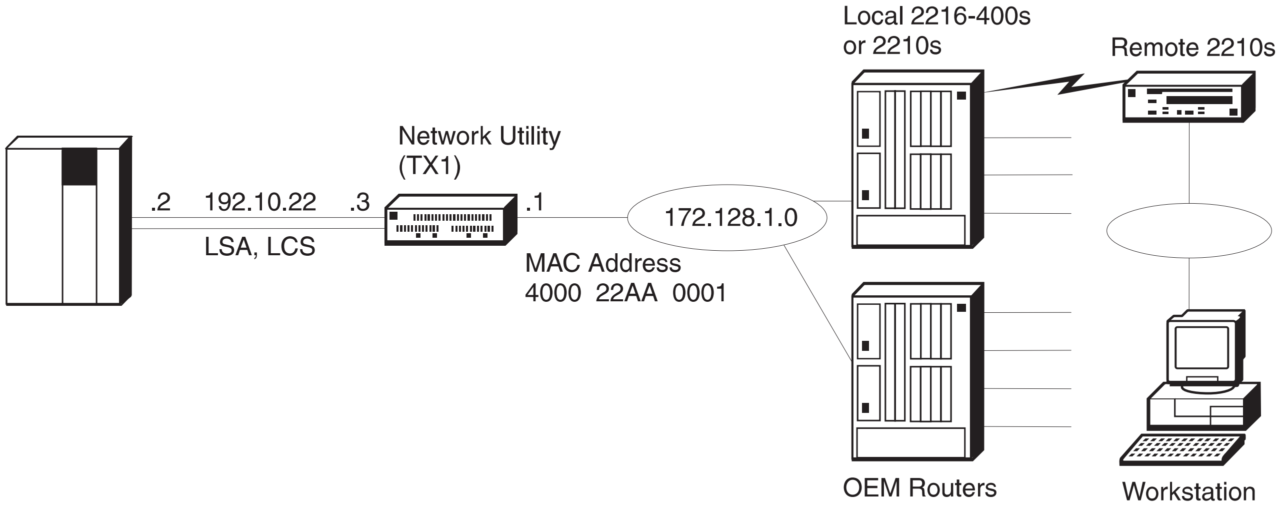


Figure 39. ESCON Channel Gateway

Table 72. ESCON Channel Gateway. See page 208 for a description and 223 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes
Devices Adapters Slots	Slot 1: 2-Port TR Slot 2: ESCON	`See "add device" on next row`	1

Devices
Adapters
Ports
	Slot 1
	Port 1: Interface 0: TR
	Slot 2
	Port 1: Interface 1: ESCON
		`Config>`**`add dev tok`**
		`Config>`**`add dev esc`**
			2

Devices
Interfaces
	Interface 0
	MAC address: 400022AA0001
		`Config>`**`net 0`**
		`TKR config>`**`set phy 40:00:22:AA:00:01`**	

Devices
Channel Adapters
ESCON Interfaces
ESCON Interfaces
	Interface 2 (new definition)
	Base Network Number: 1
	Protocol Type: LSA
	Maximum Data Frame: 2052
	LAN Net Number: 0
	(click on **Add** to create interface 2)
		`Config>`**`net 1`**
		`ESCON Config>`**`add lsa`**
		(added as interface 2)
		`ESCON Add Virtual>`**`maxdata 2052`**
		`ESCON Add Virtual>`**`net 0`**
		(continue in same session with next row)
			3,4,5

Devices
Channel Adapters
ESCON Interfaces
ESCON Subchannels
	Interface 2 (highlight LSA interface)
	Device Address: E4
	Link Address: EF
	(click on **Add**)
		`ESCON Add Virtual>`**`subchannel add`**
		`ESCON Add LSA Subchannel>`**`device E4`**
		`ESCON Add LSA Subchannel>`**`link EF`**
		(type **exit** twice and then **list all**)
			6

Devices
Channel Adapters
ESCON Interfaces
ESCON Interfaces
	Interface 3 (new definition)
	Base Network Number: 1
	Protocol Type: LCS
	LAN Type: Token Ring
	Maximum Data Frame: 2052
	MAC Address: 400022AA0009
	(click on **Add** to create interface 3)
		`Config>`**`net 1`**
		`ESCON Config>`**`add lcs`**
		(added as interface 3)
		`ESCON Add Virtual>`**`lantype token`**
		`ESCON Add Virtual>`**`Maxdata 2052`**
		`ESCON Add Virtual>`**`mac 40:00:22:AA:00:09`**
		(continue in same session with next row)	

Devices
Channel Adapters
ESCON Interfaces
ESCON Subchannels
	Interface 3 (highlight LCS interface)
	Device Address: E0
	Link Address: EF
	(click on **Add**)
		`ESCON Add Virtual>`**`subchannel add`**
		`ESCON Config LCS Subchannel>`**`device E0`**
		`ESCON Config LCS Subchannel>`**`link EF`**
		(type **exit** twice and then **list all**)
			7

System
General
	System name: NU_A
	Location: XYZ
	Contact: Administrator
		`Config>`**`set host`**
		`Config>`**`set location`**
		`Config>`**`set contact`**	

Table 72. ESCON Channel Gateway (continued). See page 208 for a description and 223 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes
System SNMP Config General	SNMP (checked)	Config>**p snmp** SNMP Config>**enable snmp**	
System SNMP Config Communities General	Community name: admin Access type: Read-write trap Community view: All	SNMP Config>**add community** SNMP Config>**set comm access write**	
Protocols IP General	Internal address: 172.128.252.1 Router ID: 172.128.1.1	Config>**p ip** IP config>**set internal 172.128.252.1** IP config>**set router-id 172.128.1.1**	
Protocols IP Interfaces	Interface 0 (TR slot 1 port 1) IP address: 172.128.1.1 Subnet mask: 255.255.255.0 Interface 3 (LCS interface) IP address: 192.10.22.3 Subnet mask: 255.255.255.0	IP config>**add address** (once per i/f)	8
Protocols IP OSPF General	OSPF (checked)	Config>**p ospf** OSPF Config>**enable ospf**	8
Protocols IP OSPF Area Configuration General	Area number: 0.0.0.0 Stub area (not checked)	OSPF Config>**set area**	
Protocols IP OSPF AS Boundary Routing	AS Boundary Routing (checked to enable) Import direct routes (checked to enable)	OSPF config>**enable as** Import direct routes (Accept other defaults)	9
Protocols IP OSPF Interfaces	Interface 0 OSPF (checked)	OSPF Config>**set interface** Interface IP address: **172.128.1.1** Attaches to area**: 0.0.0.0** (Accept other defaults)	

Notes:

1. **add dev** defines a single port, not an adapter.
2. The configuration program assigns an interface number to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the **add dev** command for each port you want to use, and the interface number (also known as ″net number″) is the output of the command.
3. When you select an interface of type LSA, the “LAN type” field is disabled (gets grayed out) and the “LAN net number” and “loopback” checkbox appears.
4. The “LAN number” field is disabled because a value is assigned by the router automatically. This value must be configured in the host definition for “ADAPTNO”.
5. When you “Add” the interface, a new interface will be generated and it will be assigned the next available interface number.
6. The values that you enter when configuring the subchannels must match values configured at the host. See “Chapter 18. Sample Host Definitions” on page 259 for examples of how to match these values.
7. When you add subchannels for an LCS virtual interface, it is only necessary to define one subchannel although LCS requires two. LCS automatically uses the next subchannel in addition to the one defined here. LCS uses the even device address (E0 in this case) as the write subchannel and the odd address (E1) as the read subchannel.
8. You can also use RIP in place of OSPF.
9. You need to import direct routes into OSPF from the ESCON interface because OSPF is not enabled on the ESCON interface. Instead, the subnet on the ESCON interface is imported into OSPF in the Network Utility and then propagated to the network. This is necessary to prevent error messages from occurring at the host if the Network Utility sends the OSPF updates over the LCS connection. TCP/IP at the host does not (yet) support Link State Advertisements from an OSPF router.

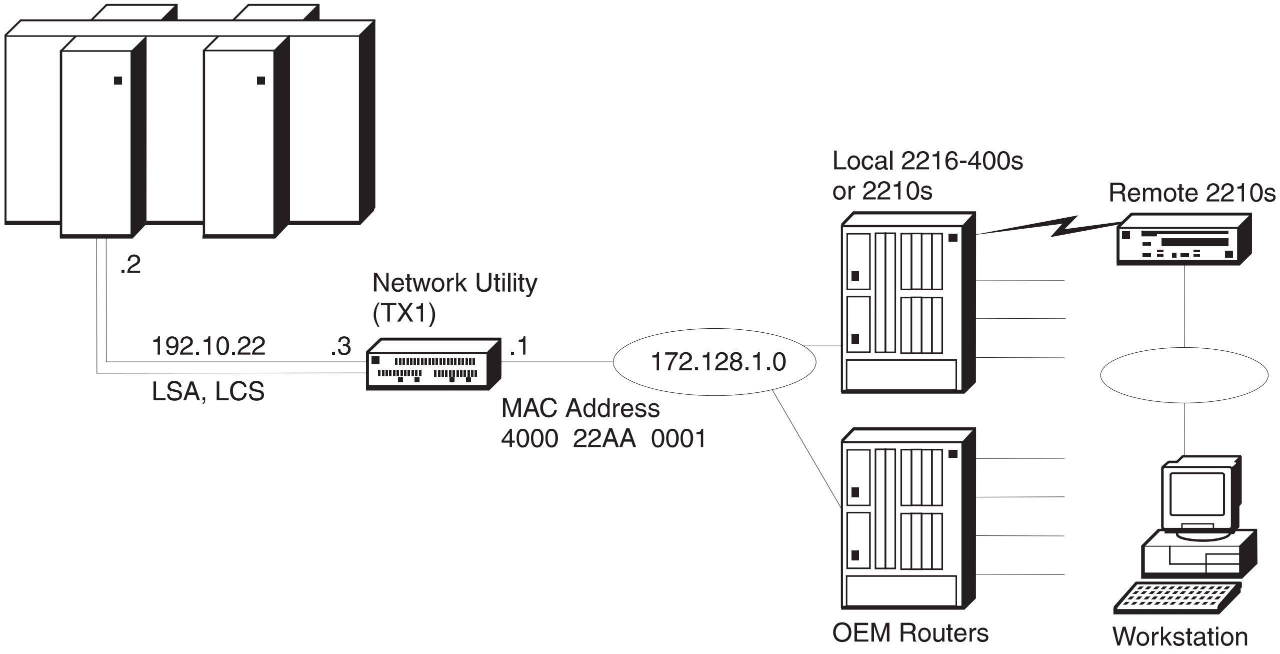


Figure 40. Parallel Channel Gateway

Table 73. Parallel Channel Gateway. See page 215 for a description and 227 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes
Devices Adapters Slots	Slot 1: 2-Port TR Slot 2: Parallel Channel Adapter (PCA)	`See "add device" on next row`	1

Devices
Adapters
Ports
	Slot 1/Port 1: Interface 0: TR
	Slot 2/Port 1: Interface 1: PCA
		`Config>`**`add dev tok`**
		`Config>`**`add dev PCA`**
			2

Devices
Interfaces
	Interface 0
	MAC address: 400022AA0001
		`Config>`**`net 0`**
		`TKR config>`**`set phy 40:00:22:AA:00:01`**	

Devices
Channel Adapters
PCA Interfaces
PCA Interfaces
	Interface 2 (new definition)
	Base Network Number: 1
	Protocol Type: LSA
	LAN Net Number: 0
	(click on **Add** to create interface 2)
		`Config>`**`net 1`**
		`PCA Config>`**`add lsa`**
		(added as interface 2)
		`PCA Add Virtual>`**`net 0`**
		(continue in same session with next row)
			3,4,5

Devices
Channel Adapters
PCA Interfaces
PCA Subchannels
	Interface 2 (highlight LSA interface)
	Device Address: 00
	Subchannel type: read/write
	(click on **Add**)
		`PCA Add Virtual>`**`subchannel add`**
		`PCA Add LSA Subchannel>`**`device 00`**
		(Type **exit** twice and then **list all**)
			6

Devices
Channel Adapters
PCA Interfaces
PCA Interfaces
	Interface 3 (new definition)
	Base Network Number: 1
	Protocol Type: LCS
	MAC Address: 400022AA0009
	(click on **Add** to create interface 3)
		`Config>`**`net 1`**
		`PCA Config>`**`add lcs`**
		(added as interface 3):
		`PCA Add Virtual>`**`mac 40:00:22:AA:00:09`**
		(continue in same session with next row)	

Devices
Channel Adapters
PCA Interfaces
PCA Subchannels
	Interface 3 (highlight LCS interface)
	Device Address: 02
	Subchannel type: write
	(click on **Add**)
		`PCA Add Virtual>`**`subchannel add`**
		`PCA Add LCS Subchannel>`**`device 02`**
		(Type **exit** twice and then **list all**)
			7

System
General
	System name: NU_A
	Location: XYZ
	Contact: Administrator
		`Config>`**`set host`**
		`Config>`**`set location`**
		`Config>`**`set contact`**	

System
SNMP Config
General
	SNMP (checked)	`Config>`**`p snmp`**
		`SNMP Config>`**`enable snmp`**	

Table 73. Parallel Channel Gateway (continued). See page 215 for a description and 227 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes
System SNMP Config Communities General	Community name: admin Access type: Read-write trap Community view: All	SNMP Config>**add community** SNMP Config>**set comm access write**	
Protocols IP General	Internal address: 172.128.252.1 Router ID: 172.128.1.1	Config>**p ip** IP config>**set internal 172.128.252.1** IP config>**set router-id 172.128.1.1**	
Protocols IP Interfaces	Interface 0 (TR slot 1 port 1) IP address: 172.128.1.1 Subnet mask: 255.255.255.0 Interface 3 (LCS interface) IP address: 192.10.22.3 Subnet mask: 255.255.255.0	IP config>**add address** (once per i/f)	8
Protocols IP OSPF General	OSPF (checked)	Config>**p ospf** OSPF Config>**enable ospf**	8
Protocols IP OSPF Area Configuration General	Area number: 0.0.0.0 Stub area (not checked)	OSPF Config>**set area**	
Protocols IP OSPF AS Boundary Routing	AS Boundary Routing (checked to enable) Import direct routes (checked to enable)	OSPF Config>**enable as** Import direct routes (Accept other defaults)	9
Protocols IP OSPF Interfaces	Interface 0 OSPF (checked)	OSPF Config>**set interface** Interface IP address: **172.128.1.1** Attaches to area: **0.0.0.0** (Accept other defaults)	

Notes:

1. **add dev** defines a single port, not an adapter.
2. The configuration program assigns an interface number to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the **add dev** command for each port you want to use, and the interface number (also known as ″net number″) is the output of the command.
3. When you select an interface of type LSA, the “LAN type” field is disabled (gets grayed out) and the “LAN net number” and “loopback” checkbox appears.
4. The “LAN number” field is disabled because a value is assigned by the router automatically. This value must be configured in the host definition for “ADAPTNO”.
5. When you “Add” the interface, a new interface will be generated and it will be assigned the next available interface number.
6. The values that you enter when configuring the subchannels must match values configured at the host. See “Chapter 18. Sample Host Definitions” on page 259 for examples of how to match these values.
7. When you add subchannels for an LCS virtual interface, it is only necessary to define one subchannel although LCS requires two. LCS automatically uses the next subchannel in addition to the one defined here. LCS uses the even device address (02 in this case) as the write subchannel and the odd address (03) as the read subchannel.
8. You can also use RIP in place of OSPF.
9. You need to import direct routes into OSPF from the PCA interface because OSPF is not enabled on the PCA interface. Instead, the subnet on the PCA interface is imported into OSPF in the Network Utility and then propagated to the network. This is necessary to prevent error messages from occurring at the host if the Network Utility sends the OSPF updates over the LCS connection. TCP/IP at the host does not (yet) support Link State Advertisements from an OSPF router.

Network Utility provides three ways of operating the LCS interface:

The following example illustrates an LCS Passthrough configuration:

```
*t 6
Gateway user configuration
config>add dev esc
Device Slot #(1-8) [1] ?3
Adding ESCON Channel device in slot 3 port 1 as interface #4
Use "net 4" to configure ESCON Channel parameters
Config>net 4
ESCON Config>add lcs
ESCON Add Virtual>?
LANtype
MAC address
MAXdata
BLKtimer
ACKlen
SUBchannels
ENable 3172 Emulation
Exit
ESCON Add Virtual>enable
Enabling LCS 3172 Emulation for network 5.
Please set the Network link using the "Net" command.
ESCON Add Virtual>?
```

```
BLKtimer
ACKlen
SUBchannels
DISable 3172 Emulation
NET link
Exit
ESCON Add Virtual>net 0
ESCON Add Virtual>sub add

Please add or configure one subchannel for an LCS virtual interface.
Although LCS requires two subchannels, it is only necessary to specify
one subchannel.  An adjacent subchannel will be chosen such that the
two subchannels will form a sequential pair with the write subchannel
(device address is even) before the read subchannel (device address
is odd).
ESCON Config LCS subchannel>?
LINk address (ESCD Port)
LPAR number
CU logical address
Device address
Exit
ESCON
ESCON Config LCS Subchannel>link f7
ESCON Config LCS Subchannel>lpar 0
ESCON Config LCS Subchannel>cu 0
ESCON Config LCS Subchannel>dev 20
ESCON Config LCS Subchannel>ex
ESCON Add Virtual>ex
>

ESCON Config>list
Net   5   Protocol: LCS    LAN type: Token Ring         LAN number: 0
          3172 Emulation is enabled
          MAC address: Obtained from net 0
          Block timer:    5 ms   ACK length:    10 bytes
ESCON config>list all
Net   5   Protocol: LCS    LAN type: Token Ring         LAN number: 0
          3172 Emulation is enabled
          MAC address: Obtained from net 0
          Block timer:    5 ms   ACK length:    10 bytes
          Read Subchannels:
          Sub  0   Dev addr: 21  LPAR: 0  Link addr: F7    CU addr: 0
          Write Subchannels:
          Sub  1   Dev addr: 20  LPAR: 0  Link addr: F7    CU addr:  0

ESCON Config
```

The following example illustrates the t 5 prompt with 3172 Emulation enabled:

```
LCS> list all

LCS Virtual Adapter
LCS Information for Net 5
--- ----------- --- --- --
LAN Type: Token-Ring           LAN Number: 0
Local Read Subchannel number:  1
Local Write Subchannel number: 0
MAC Address: 08005AFE0144
LCS 3172 Emulation to net 0
Status: Down
```

Figure 41 on page 232 shows how the parameters correlate between the host and the Network Utility for an LCS interface definition.

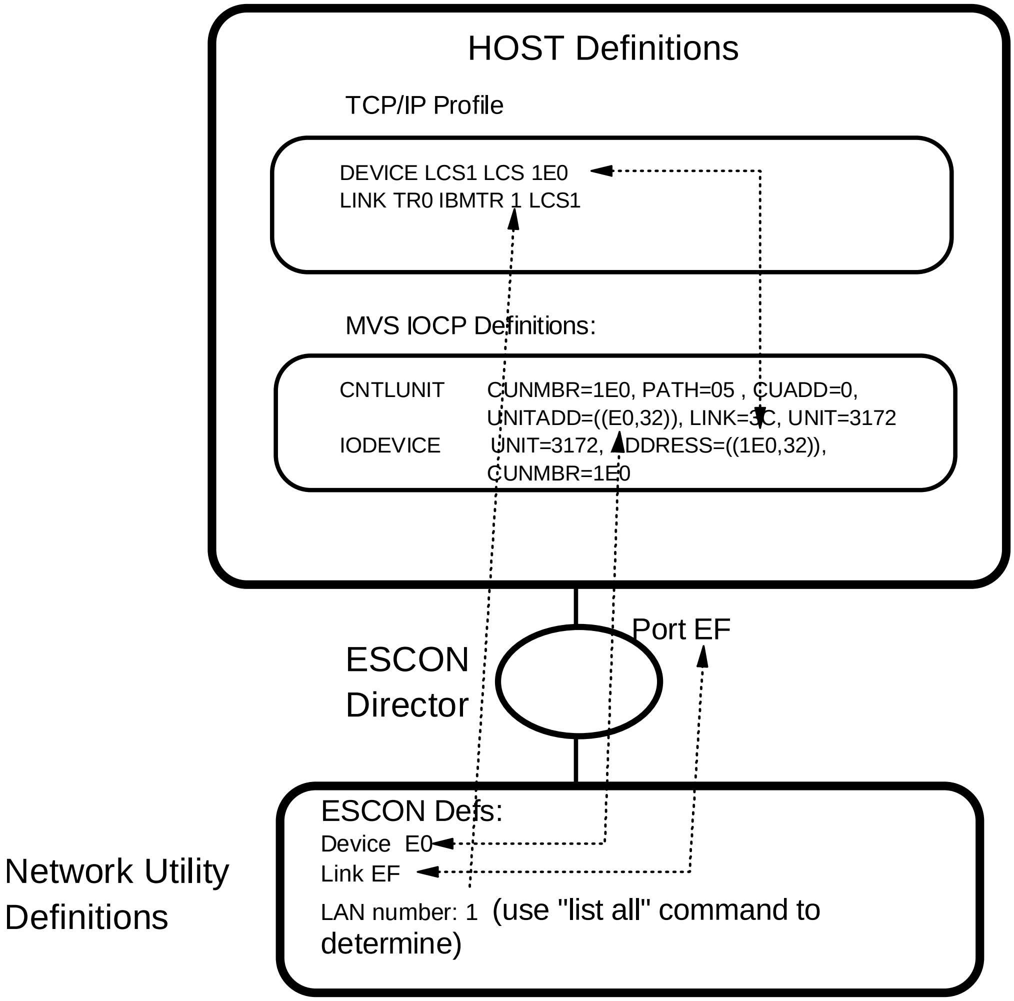

HOST Definitions
TCP/IP Profile
DEVICE LCS1 LCS 1E0
LINK TR0 IBMTR 1 LCS1
MVS IOCP Definitions:
CNTLUNIT CUNMBR=1E0, PATH=05 , CUADD=0, UNITADD=((E0,32)), LINK=3C, UNIT=3172
IODEVICE UNIT=3172, ADDRESS=((1E0,32)), CUNMBR=1E0
Port EF
ESCON Director
Network Utility Definitions
ESCON Defs:
Device E0
Link EF
LAN number: 1 (use "list all" command to determine)

Figure 41. Host/Network Utility Parameter Relationships - LCS

Notes:

1. LCS uses a pair of subchannels, one for reading and one for writing. When configuring the subchannels used by the LCS interface, you actually need to specify only one subchannel address. LCS automatically assigns two adjacent subchannels for the LCS connection, one for the read (device address is odd) and one for the write (device address is even).
2. The device address specified in the Network Utility LCS interface definition must be within the range specified in the UNITADD parameter in the CNTLUNIT macro from the IOCP. For example, the UNITADD parameter in Figure 41 shows that 32 (decimal) device addresses starting at E0 (hex) are being reserved for the Network Utility definition. A device address of E0 has been specified for the Network Utility LCS interface. The Network Utility will automatically allocate E1 also. Since E0 and E1 are in the range E0 to FF hex, this is OK as long as no other device (or interface on this Network Utility) tries to use these same subchannels.
3. The value specified in the DEVICE statement in the host TCP/IP profile must be within the range specified in the ADDRESS parameter in the IODEVICE macro from the IOCP. For example, the DEVICE statement in the host TCP/IP profile in Figure 41 is 1E0 hex, which is in the range 1E0 to 1FF that the ADDRESS parameter in the IODEVICE statement specifies.

4. The values specified for the ADDRESS parameter in the IODEVICE macro and the UNITADD parameter in the CNTLUNIT macro are related ***by convention only***. In this example, the value for the ADDRESS parameter has been determined from the value for the UNITADD parameter by prepending a ***logical channel identifier*** (1 in this case) to the UNITADD value. This will often be the case. However, when defining the device address on the Network Utility LCS definition, use the UNITADD parameter and not the ADDRESS parameter to determine the valid range of values.

See “Chapter 18. Sample Host Definitions” on page 259 for more explanation and samples of host definitions for this interface type.

For a complete look at the configuration parameters needed for this scenario, see Table 19 on page 152.

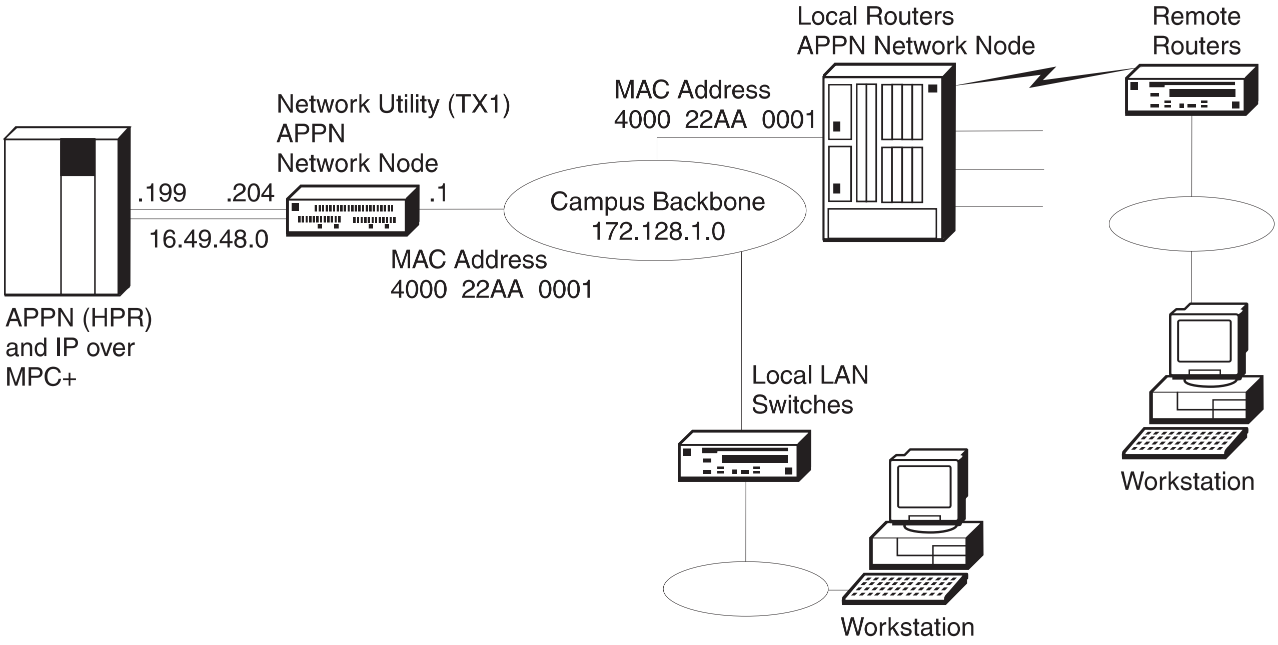


Figure 42. Channel Gateway (APPN & IP over MPC+)

Table 74. Channel Gateway (APPN & IP over MPC+). See page 216 for a description and 233 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes
Devices Adapters Slots	Slot 1: 2-Port TR Slot 2: ESCON	See "add device" on next row	1
Devices Adapters Ports	Slot 1/Port 1: Interface 0: TR Slot 2/Port 1: Interface 1: ESCON	Config>**add dev tok** Config>**add dev esc**	2
Devices Interfaces	Interface 0 MAC address: 400022AA0001	Config>**net 0** TKR config>**set phy 40:00:22:AA:00:01**	
Devices Channel Adapters ESCON Interfaces ESCON Interfaces	Interface 2 (new definition) Base Network Number: 1 Protocol Type: MPC+ (click on **Add** to create interface 2)	Config>**net 1** ESCON Config>**add mpc** (added as interface 2) ESCON Add Virtual> (continue in same session with next row)	3
Devices Channel Adapters ESCON Interfaces ESCON Subchannels	(highlight interface 2) Device Address: F0 Link Address: EF Subchannel type: Read (click on **Add** to define subchannel) Device Address: F1 Link Address: EF Subchannel type: Write (click on **Add** to define subchannel)	ESCON Add Virtual>**sub addr** ESCON Add MPC+ Read Subchannel>**dev f0** ESCON Add MPC+ Read Subchannel>**link ef** ESCON Add MPC+ Read Subchannel>**exit** ESCON Add Virtual>**sub addw** ESCON Add MPC+ Write Subchannel>**dev f1** ESCON Add MPC+ Write Subchannel>**link ef** (type **exit** twice and then **list all**)	4
System General	System name: NU_A Location: XYZ Contact: Administrator	Config>**set host** Config>**set location** Config>**set contact**	
System SNMP Config General	SNMP (checked)	Config>**p snmp** SNMP Config>**enable snmp**	
System SNMP Config Communities General	Community name: admin Access type: Read-write trap Community view: All	SNMP Config>**add community** SNMP Config>**set comm access write**	5

Table 74. Channel Gateway (APPN & IP over MPC+) (continued). See page 216 for a description and 233 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes
Protocols IP General	Internal address: 172.128.252.1 Router ID: 172.128.1.1	Config>**p ip** IP config>**set internal 172.128.252.1** IP config>**set router-id 172.128.1.1**	
Protocols IP Interfaces	Interface 0 (TR slot 1 port 1) IP address: 172.128.1.1 Subnet mask: 255.255.255.0 Interface 2 (MPC+ interface) IP address: 16.49.48.204 Subnet mask: 255.255.255.0	IP config>**add address** (once per i/f)	
Protocols IP OSPF General	OSPF (checked)	Config>**p ospf** OSPF Config>**enable ospf**	6
Protocols IP OSPF Area Configuration General	Area number: 0.0.0.0 Stub area (not checked)	OSPF Config>**set area**	
Protocols IP OSPF AS Boundary Routing	AS Boundary Routing (checked to enable) Import direct routes (checked to enable)	OSPF Config>**enable as** Import direct routes (Accept other defaults)	7
Protocols IP OSPF Interfaces	Interface 0 OSPF (checked)	OSPF Config>**set interface** Interface IP address: **172.128.1.1** Attaches to area: **0.0.0.0** (Accept other defaults)	
Protocols APPN General	APPN network node (checked to enable) Network ID: STFNET Control point name: NUGW	Config>**p appn** APPN config> **set node** Enable APPN Network ID: **STFNET** Control point name: **NUGW** (Accept other defaults)	
Protocols APPN Interfaces	(highlight Interface 0 Token Ring) (click on the configure tab) Define APPN port (checked to enable) Port name: TR001	APPN config>**add port** APPN Port Link Type: **TOKEN RING** Port name: **TR001** Enable APPN (Accept other defaults)	

Table 74. Channel Gateway (APPN & IP over MPC+) (continued). See page 216 for a description and 233 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
APPN
Interfaces
	(highlight Interface 0 Token Ring)
	(click on the Link stations tab)
	TRTG001 (new definition)
	General-1 Tab:
	Link station name: TRTG001
	General-2 Tab:
	MAC address of adjacent node:
	400022AA0011
	Adjacent Node Type:
	APPN Network Node
	(click on **Add** to create the Link station)
		`APPN config>`**`add link`**
		`Port name for the link station:` **`TR001`**
		`Station name:` **`TRTG001`**
		`MAC address of adjacent node:` **`400022AA0011`**
		(Accept other defaults)
			8

Protocols
APPN
Interfaces
	(highlight Interface 2 ESCON-MPC+)
	(click on the configure tab)
	Define APPN port (checked to enable)
	Port name: MPC001
		`APPN config>`**`add port`**
		`APPN Port Link Type:` **`MPC`**
		`Interface Number:` **`2`**
		`Port name:` **`MPC001`**
		`Enable APPN`
		(Accept other defaults)	

Protocols
APPN
Interfaces
	(highlight Interface 2 ESCON-MPC+)
	(click on the Link stations tab)
	MPCTG001 (new definition)
	General-1 Tab:
	Link station name: MPCTG001
	General-2 Tab:
	Adjacent Node Type:
	APPN Network Node
	(click on **Add** to create the Link station)
		`APPN config>`**`add link`**
		`Port name for the link station:` **`MPC001`**
		`Station name:` **`MPCTG001`**
		`Adjacent Node Type:` **`0`** `= APPN Network Node`
		(Accept other defaults)	

Notes:

1. **add dev** defines a single port, not an adapter.
2. The configuration program assigns an interface number to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the **add dev** command for each port you want to use, and the interface number (also known as ″net number″) is the output of the command.
3. When you “Add” the interface, a new interface will be generated and it will be assigned the next available interface number.
4. The values that you enter when configuring the subchannels must match values configured at the host. See “Chapter 18. Sample Host Definitions” on page 259 for examples of how to match these values.
5. You need a write-capable SNMP community only if you want to download configuration files from the configuration program directly into the router. SNMP is not required to TFTP a configuration file to the router.
6. You can also use RIP in place of OSPF.
7. You need to import direct routes into OSPF from the ESCON interface because OSPF is not enabled on the ESCON interface. Instead, the subnet on the ESCON interface is imported into OSPF in the Network Utility and then propagated to the network. This is necessary to prevent error messages from occurring at the host if the Network Utility sends the OSPF updates over the MPC+ connection. TCP/IP at the host does not (yet) support Link State Advertisements from an OSPF router.
8. The destination MAC address in this example is the local router on the right-hand side of the campus backbone in Figure 42 on page 233. This router is also configured to be an APPN network node.

Chapter 16. Data Link Switching

Overview

This section introduces Data Link Switching (DLSw) and summarizes the DLSw function implemented in Network Utility.

What is DLSw?

DLSw is an IBM-invented standard technology for transporting connection-oriented protocols, mainly SNA and NetBIOS, across IP backbone networks. DLSw routers on the edges of an IP network field link establishment requests from native SNA and NetBIOS end stations, search among peer DLSw routers for one serving the target end station, and then set up a path and relay application data between the end stations through the peer router.

The protocol that flows between DLSw routers is documented in RFC 1795, ″Data Link Switching: Switch to Switch Protocol.″ Clarifications about this protocol and multicast IP-based scalability enhancements are documented in RFC 2166, ″DLSw v2.0 Enhancements″.

Many DLSw implementations provide a *local DLSw* function that connects two links within a single router, as opposed to connecting them across an IP network to another DLSw router. Depending on the DLC types involved, this function may be equivalent to that of a FRAD or X.25 PAD.

Network Utility DLSw Function

The Network Utility DLSw implementation is nearly identical in function to that of the IBM 2210 and 2216 routers. It can handle the following end-station protocols:

- SNA
  - PU 4/5 to PU 2.0 (and IBM 5394 on SDLC)
  - T2.1 to T2.1
  - PU 4/5 to PU 4/5
- NetBIOS
  - Point-to-point sessions
  - Broadcast datagram traffic
- LAN Network Manager
  - LNM to bridge servers (e.g., LBS, CRS, REM)
  - LNM to 8235 intelligent hub
  - LNM to LAN Station Manager

Network Utility DLSw can communicate with end stations across the following data link control (DLC) types:

- 802.2 LLC

  LLC can be carried over any of these interface types:
  - Token-Ring
  - Ethernet (10 Mbps or 10/100 Mbps adapters)
  - FDDI

- PPP links enabled for remote bridging
- Frame Relay PVCs and SVCs enabled for remote bridging (RFC 1490/2427 bridged frame formats)
- ATM LAN emulation
- ATM native bridging (RFC 1483 bridged frame formats)

- SDLC

  DLSw can represent the primary station on a multipoint line, multiple secondary stations, or a single fully negotiable station on a point-to-point line.
- QLLC

  DLSw supports any combination of QLLC PVCs and SVCs on a single X.25 interface. It can handle parallel virtual circuits to the same remote DTE address, as well as incoming calls from non-configured SVCs.
- APPN

  You can configure APPN to attach to the DLSw function residing in the same Network Utility. This allows APPN to have links with any PU 2.0 or T2.1 SNA end station in the DLSw network, without requiring APPN to be present in remote (especially branch office) routers.
- Channel-LSA

  DLSw supports an internal interface to the ESCON and parallel channel LSA function residing in the same Network Utility. This allow the host to have links with any SNA end station in the DLSw network, without requiring separate channel gateway and central-site DLSw router products.

With remote DLSw (across IP to another router), Network Utility DLSw supports conversion from TCP DLSw frames to any of the supported DLC types. Local DLSw is supported only for specific combinations of DLC types, as shown here:

	LLC	SDLC	QLLC	APPN	Channel-LSA
LLC	(1)	x	x		x
SDLC	x	x	x	x	x
QLLC	x	x	x	x	x
APPN		x	x		x
CHANNEL	x	x	x	x	

```
Note:
1 - You should use bridging for local LLC-to-LLC connectivity.  The
    only exception supported by local DLSW is LLC to a Frame Relay
    bridge port that is configured as a Boundary Access Node (BAN) port.
```

The following list summarizes some of the other capabilities and features of IBM Network Utility DLSw.

- Dynamic compatibility to all DLSw protocol standards

  IBM DLSw supports RFC 1434+, RFC 1795 (DLSw Version 1), and RFC 2166 (DLSw Version 2). It dynamically detects the protocol level of each partner router with no pre-configuration, and can simultaneously handle partners at different protocol levels.
- Dynamic and on-demand partners

  IBM DLSw supports bringing up TCP connections to configured partners only when required, as well as discovering end stations served by non-configured partners, and bringing up those TCP connections on demand.
- Multicast IP discovery

  With the simple configuration of multicast IP addresses or groups, IBM DLSw can perform multicast searches for both end stations and partners. IBM DLSw provides a number of dynamic extensions to the DLSw Version 2 standard, including resource registration and simplified group configuration.

- Traffic prioritization

  There are configuration options allowing you to control not only SNA versus NetBIOS prioritization, but also individual circuit priorities. This is in addition to the Bandwidth Reservation System's (BRS) extensive support for interface-level traffic prioritization.

- Advanced filtering and static cache entries

  IBM DLSw includes extensive support for MAC address and NetBIOS name lists and static caching, allowing you to control what links are used for searching for resources as well as which remote partners are preferred.

- Load balancing and fault tolerance

  IBM DLSw can cache multiple remote partners and select among them on the basis of neighbor priority, largest frame size support, or first to reply. You can also use the neighbor priority feature to ensure that one central-site router serves only as a backup for another.

  For configurations involving duplicate MAC addresses, you can disable the neighbor priority feature, or set cache parameters to control the paths used to reach those MAC addresses.

Example Configurations

This section describes three sample configurations that use the Data Link Switching feature of the Network Utility. These configurations are:

- DLSw LAN Catcher
- DLSw LAN Channel Gateway
- DLSw X.25 Channel Gateway

DLSw LAN Catcher

This scenario is shown in Figure 43 on page 242. In this scenario, the SNA traffic in the remote sites uses DLSw to get back to the data center.

The Network Utility is in the data center on the backbone LAN segment. It is a DLSw partner with each remote router and as such requires a TCP session with each. The advantage to this approach is that all of the CPU cycles needed to manage these TCP sessions and to terminate the DLSw connections are concentrated in the Network Utility. Without the Network Utility, the local routers or the host gateway (if DLSw-capable) could be consumed by this workload.

From the host perspective, the SNA LLC2 traffic is bridged into the Network Utility from the host gateway. The host gateway is either an IBM 3745/46, an IBM 3746 with the Multiaccess Enclosure (MAE), or an IBM 2216.

You can take advantage of the 2-Port Token-Ring Adapter in the Network Utility by bringing in the IP-encapsulated SNA traffic on one port and delivering LLC2 SNA traffic onto the other Token-Ring port. Thus, you have twice the bandwidth available with an additional benefit of separating the IP and SNA traffic onto separate rings. Because the Network Utility provides LLC local acknowledgments (spoofing) to the host for each LLC connection, this removes a considerable amount of traffic from the campus backbone in large network environments.

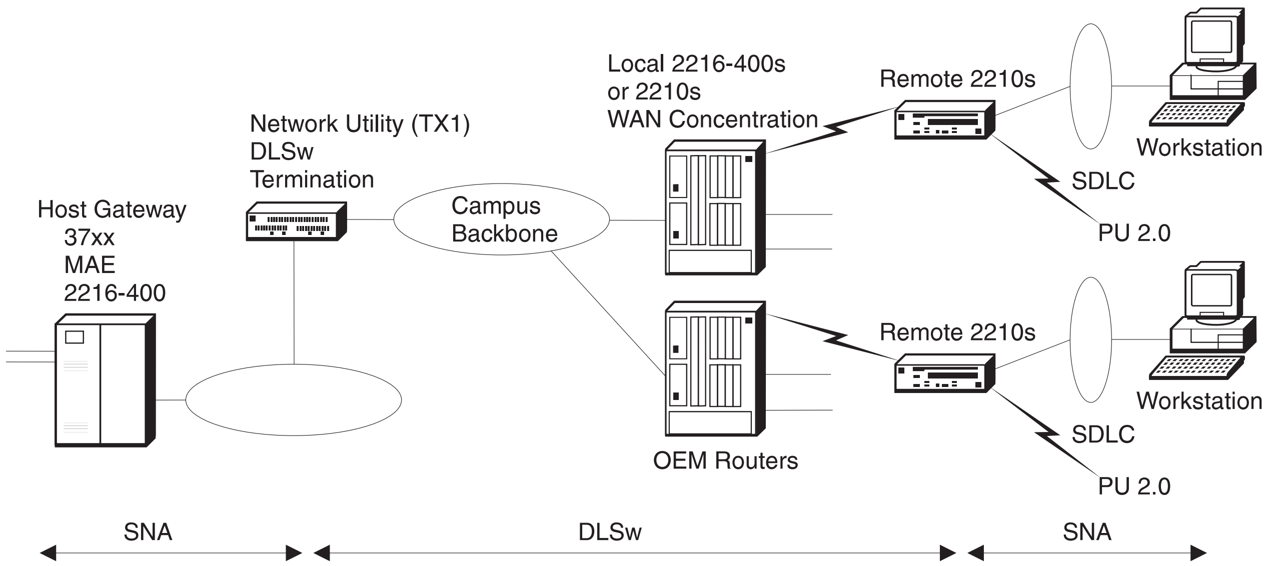


Figure 43. DLSw LAN Catcher

Keys to Configuration

For the most part, this is a standard DLSw configuration. However, you should be aware of the following points when configuring the Network Utility as a DLSw LAN Catcher:

- For this scenario, you should configure the Network Utility to allow TCP sessions from any of the remote routers. This is called DLSw dynamic neighbors. This keeps you from having to define the IP address of each DLSw partner on the Network Utility. The default value for dynamic neighbors is ″Enabled.″
- The Network Utility introduces a new parameter for IBM DLSw implementations that allows you to specify how explorer frames are forwarded. This is especially important in the outbound direction from the central site. The parameter is called *enable/disable forwarding explorers* and it gives you the flexibility to specify any of the following options:
  - Disable forwarding of explorer frames

    This option completely disables forwarding of explorer frames.
  - Forward explorer frames to the local TCP connection only

    If you want to block explorer frames from going out on WAN links, then you can specify this option. This is the default value for the Network Utility.
  - Forward explorer frames to all DLSw partners

    With this option, explorer frames are sent out to all DLSw partners.

For a complete look at the configuration parameters needed for the DLSw LAN Catcher scenario, see Table 76 on page 252.

DLSw LAN Channel Gateway

This scenario is shown in Figure 44 on page 243. As in the DLSw LAN catcher scenario, the Network Utility terminates the DLSw sessions from the remote routers. However, in this case, there is an ESCON Channel Adapter in the Network Utility. Instead of bridging the traffic from the DLSw function onto the LAN segment, this configuration passes it directly to the channel via an LSA loopback interface configured in the Network Utility.

This configuration also demonstrates the use of the Network Utility to support SNA traffic from the local campus to the host. This traffic is bridged off the campus backbone through the LSA loopback interface. All SNA devices in the network are configured with the same host destination MAC address which is the MAC address of the LSA loopback interface. This includes the devices at the main site as well as the devices in the remote sites.

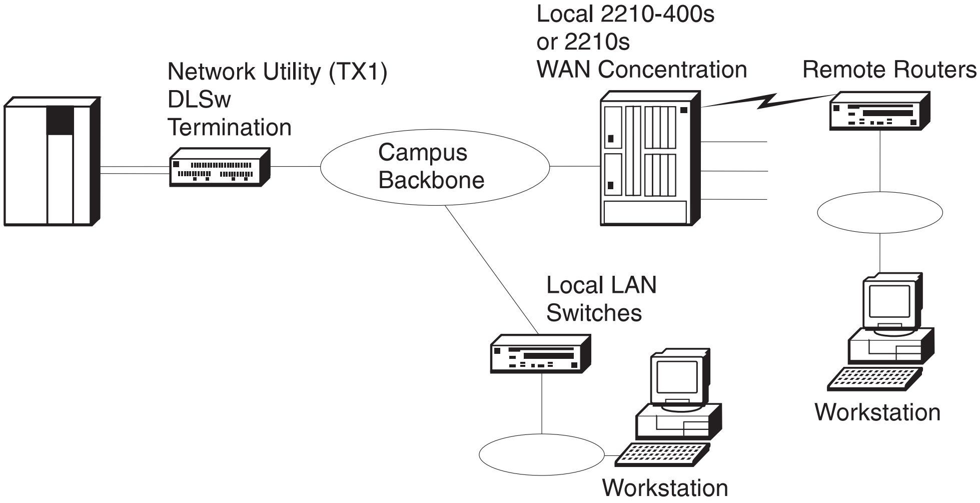


Figure 44. DLSw LAN Channel Gateway

Note: This example illustrates the use of the Network Utility as a channel gateway for DLSw traffic only. However, many of the functions illustrated in the Channel Gateway example configurations on page 208 could be combined with DLSw termination in a valid channel gateway configuration.

Keys to Configuration

Note the following points when configuring the Network Utility as a DLSw LAN Channel Gateway:

- An LSA interface must be configured and loopback must be enabled on this interface. Enabling loopback creates a virtual LAN inside the Network Utility. The only two devices on this LAN are the host and the DLSw termination point. A MAC address is defined on the LSA interface that represents the host on the channel. This is the destination MAC address that is configured in the downstream devices.

  Note: You can also define an LSA direct connection for the traffic to be bridged in from the local LAN segments. If you do this, then the devices on these segments will have a different destination MAC address from the remote devices because the LSA direct interface will have a different MAC address from the LSA loopback interface.
- When configuring DLSw, you need to open SNA SAPs for the LSA interface as well as the Token-Ring interface.
- The subchannel configuration for the LSA interface must match parameters configured in the host. See Table 70 on page 209 for a description of the subchannel parameters and “Chapter 18. Sample Host Definitions” on page 259 for example host definitions. This information will help you see how these parameters correlate.

- You need to configure a *local TCP connection*. This is done by defining a DLSw partner whose IP address is the internal address of the Network Utility. This is used for the traffic that is bridged from the local LAN segments into the host. This traffic gets bridged into the Network Utility into DLSw where the local TCP connection passes the traffic to the LSA loopback interface.
- The Network Utility currently supports a maximum of 2048 link stations per MAC address/SAP pair (for example, a destination MAC address of 400022AA0099 with SAP 04). If you need more then 2048 workstations, you have to define another LSA interface with a different SAP or a different MAC address. Remember that each LSA interface requires one subchannel of the 64 available on one ESCON channel adapter. You must also define the corresponding XCA major node to support each LSA interface.

X.25 Channel Gateway

This scenario is shown in Figure 45 on page 245. It uses Local DLSw in the Network Utility to map between X.25 addresses and MAC address/SAP pairs. The transport across the WAN is native Qualified Logical Link Control (QLLC), a protocol that allows SNA devices to communicate over X.25 networks. In the Network Utility, local DLSw performs protocol conversion between QLLC and LLC2 frames.

From the remote device perspective, there are two cases to consider:

1. A device on a LAN segment attached to the branch router

   On the workstation, the SNA application generates an LLC frame that it wants to send to the host. If the branch router is an IBM 2210, this LLC frame gets bridged into the 2210 DLSw function, which does three things:

   a. Protocol conversion from the LLC frame to a QLLC frame
   b. Maps the destination MAC address/SAP pair into the appropriate X.25 LCN (PVC) or DTE address (SVC)
   c. Passes the QLLC frame to X.25

   The X.25 PAD function in the branch router creates the LAPB link layer packets and sends them over the PVC (or SVC).

   If some product other than the IBM 2210 plays the role of branch router, it needs to perform these same functions but may do so without using local DLSw.
2. A device directly on the X.25 network (for example, an IBM 3174 Control Unit or an eNetwork Communications Server gateway machine attached via a Wide Area Connector Adapter)

   On these devices, SNA uses QLLC as a native DLC type. It generates a QLLC frame and sends it out over the configured PVC (or SVC).

In each of these cases, at the Network Utility, the LAPB packets are received over the X.25 circuit and passed to QLLC and then on to DLSw. DLSw does two things:

1. Protocol conversion from QLLC into an LLC2 frame
2. Mapping of the X.25 LCN (PVC) or DTE address (SVC) into the MAC address/SAP for the LSA local loopback interface

The traffic is then passed to the LSA loopback interface for transport across the ESCON channel.

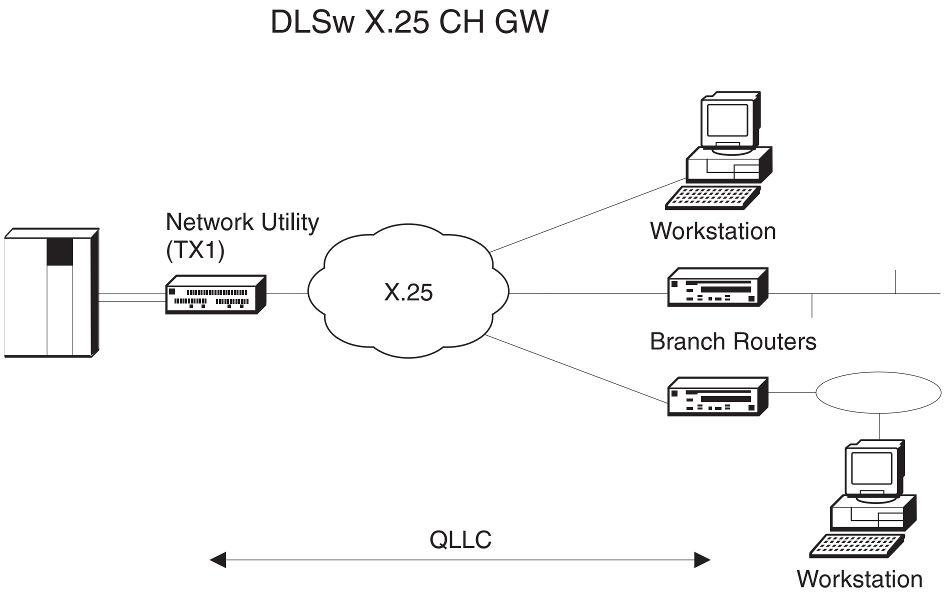


Figure 45. DLSw X.25 Channel Gateway

Keys to Configuration

The following list summarizes general configuration tasks you need to perform for this scenario. Please refer to other DLSw and LSA loopback configurations for details. The LSA loopback interface is configured the same as in "DLSw LAN Channel Gateway" on page 242.

- Add and configure the ESCON and LSA interfaces.
- Add and configure the X.25 interface. From the command line, use the **net** command in talk 6 to enter the X.25 Config subprocess, then use the following commands:
  - **set address** (to set the local DTE address)
  - **add protocol dlsw** (to add DLSw as an X.25 protocol)
  - **add pvc** or **add svc** (to add the individual PVCs or range of SVCs)
- Configure the IP internal address as in other examples.
- Configure DLSw
  - Configure general DLSw (enable, SRB segment, forward explorers locally).
  - Configure the DLSw local TCP connection.
  - Configure DLSw for LSA loopback (open SAPs on the LSA interface).

In addition to these general tasks, you need to configure Network Utility DLSw to map X.25 addresses to the LSA loopback MAC address. There are three ways to do this:

- Configure the X.25 stations individually at DLSw, each with its own destination MAC address. This option applies to both PVCs and SVCs.
- Configure a list of connection IDs, each of which has its own destination MAC address. Some X.25 stations can send a connection ID when they place a call, and Network Utility matches this value to the configured list. This option applies to SVCs only.
- Configure a default destination MAC address for incoming calls that do not contain a connection ID. This option applies to SVCs only.

The remainder of this section describes how to configure each of these three address mapping methods.

If the number of remote X.25 stations is relatively small, then you can configure each remote X.25 device in DLSw to be mapped to the LSA loopback MAC address. To do this using the command line, enter **talk 6** at the * prompt and type the following:

- **protocol dls**
  - **add qllc station** (once for each remote X.25 station). The system prompts you for:
    - Interface number (X.25 interface)
    - PVC or SVC
    - Logical channel number (for PVCs) or DTE address (for SVCs)
    - Source MAC and SAP (can be generated by DLSw)
    - Destination MAC and SAP (enter the LSA loopback MAC address)
    - PU type
    - XID block/num (if the PU type is 2)

To do this using the Configuration Program, do the following:

- Protocols/DLSw/Interfaces/Serial-X25/QLLC Stations
  - add a QLLC station (enter the same information as above)

If your remote X.25 stations can be configured to send a connection ID when they place a call[21], you can configure DLSw to map connection ID values to destination MAC addresses. To do this using the command line, enter **talk 6** at the * prompt and type the following:

- **protocol dls**
  - **add qllc destination** (once for each valid connection ID). The system prompts you for:
    - Connection ID
    - Destination MAC and SAP (enter the LSA loopback MAC address)

To do this using the Configuration Program, do the following:

- Protocols/DLSw/QLLC Destinations
  - add a QLLC destination (enter the same information as above)

Finally, if it is not feasible to configure each remote X.25 station or to use a connection ID, you can use the DLSw ANYCALL feature to accept any incoming X.25 call and map it to the LSA loopback MAC address. To do this using the command line, enter **talk 6** at the * prompt and type the following:

- **protocol dls**
  - **add qllc destination** (once, plus you can add specific connection IDs if you wish). The system prompts you for:
    - Connection ID (use the word ’ANYCALL’)
    - Destination MAC and SAP (enter the LSA loopback MAC address)

To do this using the configuration tool, do the following:

- Protocols/DLSw/QLLC Destinations
  - add a QLLC destination (enter the same information as above)

[21] QLLC products frequently present this parameter as a connection password.

Managing DLSw

This section introduces some of the ways in which you can monitor and manage the DLSw function.

Command-Line Monitoring

DLSw supports an extensive set of commands to display status, dynamically modify configuration parameters, and actively control the state of connections. These commands are described in detail in *MAS Protocol Configuration and Monitoring Reference Volume 1*, in the chapter ″Configuring and Monitoring DLSw.″ To access them, enter **talk 5** at the * prompt and **protocol dls** at the + prompt.

Some particularly useful commands for monitoring status are:

list tcp sess

Shows the status of all known TCP connections to partner routers. You can see the state of the TCP connections as they come up and go down, as well as the level of the DLSw protocol in use, and summary statistics on the number of DLSw circuits using each connection. If you configure DLSw to accept TCP connections only from dynamic (not configured) partners, this command displays the status of connections as initiated by remote routers. There will be no status if the remote routers are not actively bringing up TCP connections.

If you configure a ″local TCP connection″ to enable local DLSw function, this connection is flagged as such on the command output so that it can be distinguished from remote partner connections.

list dls sess all

Shows the status of all active DLSw sessions. A session, also called a circuit, is defined by a MAC and SAP address 4-tuple and corresponds to an SNA link, not an SNA LU-LU session. Sessions are normally driven up and down by SNA end stations, so the output of this command is dynamic. For every session, you see its identifying MAC and SAP addresses, state, which partner the session is connected through, and an identifier that you can use with the **list dls sess detail** command to get more information. Local DLSw sessions (those that involve only this router) show as two lines of output from this command.

Because a Network Utility may easily have hundreds or thousands of active sessions, you can use different variations of the **list dls session** command to display only a subset of them. Instead of the keyword ″all,″ you use different keywords to show only those circuits through a given partner, or only those in a given state, and so on. There are roughly 10 keywords defined to select sessions. The output of all these commands pauses when the screen fills, waiting for a keystroke from you to continue or quit. Press the space bar to view the next screen of output.

list dls mem

Shows the status of various pools of DLSw memory, as well as the memory congestion status for all active sessions.

list llc sess all

Shows 802.2 LLC-specific status information for all DLSw sessions that use LLC as the protocol between the router and the end station. These include sessions running over LAN, channel, ATM, and remotely bridged WAN interfaces. The command output includes more state information as well as the source route to the end station, if applicable.

list sdlc sess all
Shows SDLC-specific status information for all DLSw sessions that use SDLC as the protocol between the router and the end station. The command output includes SDLC addressing information as well as state information. If you are working with SDLC devices, this command is more useful than the generic **list dls sess**.

list qllc sess
Shows QLLC-specific status information for all DLSw sessions that use QLLC over X.25 as the protocol between the router and the end station. The command output includes QLLC addressing information as well as detailed state information. Because the router supports incoming dynamic SVCs, this command is essential to see the status of both configured and dynamic QLLC PVCs and SVCs.

DLSw supports dynamic modification under talk 5 of the vast majority of parameters you can configure under talk 6. DLSw follows the standard model where changes made under talk 5 have an immediate effect but do not survive a box reboot, while changes made under talk 6 take effect only after a box reboot. The talk 5 **list** commands show the values that are currently active in the running product.

The talk 5 commands **delete** and **disable** give you the power to tear down an existing DLSw connection. For example, you can use **delete dls** *session number* to clean up a hung session and allow the end stations to redrive it. **Delete/add** and **disable/enable** sequences are powerful methods to recycle configured TCP, SDLC, and QLLC connections.

Event Logging Support

DLSw has several hundred ELS messages defined, ranging from informational messages about normal events, to warnings of serious error conditions. Here are some of the types of DLSw events that can generate ELS messages:

- Initialization and configuration errors
- Partner TCP connection and capabilities frames sent or received
- Explorer frames sent or received for a particular MAC address or NetBIOS name
- Circuit setup/takedown frames sent or received
- DLC link setup/takedown frames sent or received
- Data frames sent or received on active circuits
- Pacing window changes on active circuits
- Memory allocation errors
- Unexpected protocol flows, frames discarded
- Frame flows do not match configuration

Although these messages are used primarily by software engineers to resolve problems, a user with a basic knowledge of the DLSw protocol and DLC link activation flows should be able to make sense of them and debug simple configuration mistakes. By activating these ELS messages and watching the output via talk 2, you should be able to at least answer the question ″Is anything happening?″

″DLS″ is one of the named *subsystems* within ELS. To activate the standard set of error messages, type **disp sub dls** from the event menu under either talk 6 or talk

5. To activate all DLSw messages, enter **disp sub dls all**. The corresponding commands to deactivate messages begin with **nodisp**. For general information on controlling and viewing ELS messages, see “Monitoring Event Messages” on page 90.

If you are trying to trace a link activation attempt, DLSw messages alone may not show the complete picture. You can activate the ELS messages for the underlying DLC type as follows:

LLC disp sub llc all

SDLC disp sub sdlc all

QLLC disp sub qllc all disp sub x253 all (X.25 layer 3, the packet layer)

Channel-LSA
disp sub lsa all

Refer to the *Event Logging System Messages Guide* (on CD-ROM and the 2216 Web Page) for a full list of individual messages and their meaning.

SNA Management Support

From a VTAM or NetView/390 operator console, you can control the links, PU, and LUs involved with DLSw as described in “NetView/390” on page 99.

Unlike APPN, Network Utility DLSw does not send SNA alerts. It does send traps (described in the following section) and trigger ELS messages that can generate traps. You can use the products mentioned in “IBM Nways Manager for AIX” on page 96 to convert those traps to alerts.

SNMP MIB and Trap Support

Network Utility DLSw provides full read-only and partial read-write support for the IETF standard DLSw MIB documented in RFC 2024. This large MIB gives visibility to most of the important configuration, status, and accounting information that products implementing RFC 1795 and 2166 should have. This information includes:

- Configuration
  - Node characteristics, for example, dynamic partners are enabled
  - Configured partner information
  - Configured directory/cache entries
- Status
  - Node up or down, for how long
  - Active TCP connections, for how long, dynamic partner information
  - Dynamic directory/cache information
  - Active circuits, for how long, DLC information
- Statistics and Accounting
  - Counts of TCP connections up and down (normal and error)
  - Data and control frames counts per partner
  - Counts of circuits up and down
  - Indices to underlying DLC MIBs for per-circuit frame counts
  - Pacing counts for active circuits

Network Utility DLSw supports all the traps defined in RFC 2024, reporting the following events:

- A TCP connection is terminated due to capabilities exchange failure or a DLSw protocol violation
- A TCP connection comes up or goes down
- A circuit comes up or goes down

DLSw all supports trap control data items so a management station can set the conditions under which a trap is generated.

In addition to RFC 2024, Network Utility DLSw supports′ IBM-specific DLSw MIB extensions for multicast IP-based groups and for QLLC stations.

Network Management Application Support

The Network Utility Java-based application implemented in the Nways Manager products discussed in “IBM Nways Manager Products” on page 96 provides integrated support for the standard DLSw MIB and the IBM-specific DLSw MIB extensions.

To view DLSw resources and their status using these products, you bring up specific panels that present key information from the DLSw MIB and from its underlying DLC-layer MIBs (LLC, SDLC, or X.25). You can also use integrated browser support to view the information in any of these MIBs.

You can control the emission of DLSw traps from the Nways Manager products, so that a given trap is generated always, never, or only under certain conditions.

Nways Manager for AIX can show you a DLSw topology view of your network, including DLSw connectivity, resources, and color-coded status. The topology is refreshed as new nodes are discovered. This application does not present the topology of DLSw IP multicast groups.

Chapter 17. DLSw Example Configuration Details

This chapter contains diagrams and configuration parameter tables for several of the example DLSw network configurations in “Chapter 16. Data Link Switching” on page 239. The parameter values shown are from real working test configurations.

For an explanation of the columns and conventions in the configuration parameter tables, see “Example Configuration Table Conventions” on page 125.

The Network Utility World Wide Web pages contain binary configuration files that match these configuration parameter tables. To access these files, follow the Download link from:

```
http://www.networking.ibm.com/networkutility
```

The configurations documented in this chapter are:

Table 75. Cross-Reference of Example Configuration Information

Configuration Description	Parameter Table
“DLSw LAN Catcher” on page 241	Table 76 on page 252
“DLSw LAN Channel Gateway” on page 242	Table 77 on page 256

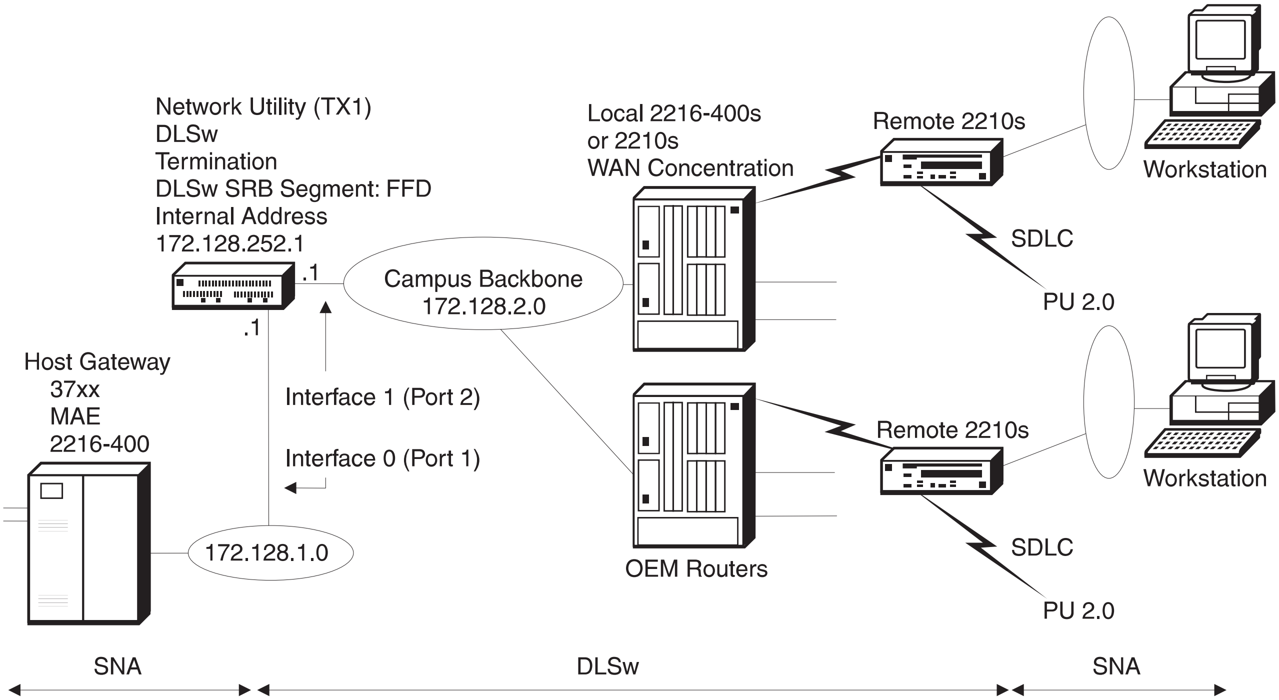


Figure 46. DLSw LAN Catcher

Table 76. DLSw LAN Catcher. See page 241 for a description and 251 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes
Devices Adapters Slots	Slot 1: 2-Port TR	See ″add device″ on next row	1
Devices Adapters Ports	Slot 1/Port 1: Interface 0: TR Slot 1/Port 2: Interface 1: TR	Config>**add dev tok** (once per port)	2
Devices Interfaces	Interface 0 MAC address: 400022AA0001 Packet size: 4399 Interface 1 MAC address: 400022AA0002 Packet size: 4399	Config>**net 0** TKR config>**set phy 40:00:22:AA:00:01** TKR config>**packet 4399** TKR config>**exit** Config>**net 1** TKR config>**set phy 40:00:22:AA:00:02** TKR config>**packet 4399**	
System General	System name: NU_A Location: XYZ Contact: Administrator	Config>**set host** Config>**set location** Config>**set contact**	
System SNMP Config General	SNMP (checked)	Config>**p snmp** SNMP Config>**enable snmp**	
System SNMP Config Communities General	Community name: admin Access type: Read-write trap Community view: All	SNMP Config>**add community** SNMP Config>**set comm access write**	3
Protocols IP General	Internal address: 172.128.252.1 Router ID: 172.128.1.1	Config>**p ip** IP config>**set internal** IP config>**set router-id**	
Protocols IP Interfaces	Interface 0 (TR slot 1 port 1) IP address: 172.128.1.1 Subnet mask: 255.255.255.0 Interface 1 (TR slot 1 port 2) IP address: 172.128.2.1 Subnet mask: 255.255.255.0	IP config>**add address** (once per i/f)	4
Protocols IP OSPF General	OSPF (checked)	Config>**p ospf** OSPF Config>**enable ospf**	5

Table 76. DLSw LAN Catcher (continued). See page 241 for a description and 251 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes
Protocols IP OSPF Area Configuration General	Area number: 0.0.0.0 Stub area (not checked)	OSPF Config>**set area**	
Protocols IP OSPF Interfaces	Interface 0 OSPF (checked) Interface 1 OSPF (checked)	OSPF Config> **set interface** Interface IP address **172.128.1.1** Attaches to area **0.0.0.0** (Accept other defaults) OSPF Config>**set interface** Interface IP address **172.128.2.1** Attaches to area **0.0.0.0** (Accept other defaults)	
Protocols DLSw General General	DLSw (checked) SRB segment: FFD Forward explorers: disabled	Config>**p dls** DLSw Config>**enable dls** DLSw Config>**set srb** DLSw Config>**disable forward all**	6
Protocols DLSw General Dynamic Neighbors	Dynamic neighbors (checked)	DLSw Config>**enable dynamic**	7
Protocols DLSw Interfaces	Interface 0 (TR slot 1 port 1) SAP type: SNA (SAPs 0,4,8,C)	DLSw Config>**open 0 sna**	8
Protocols Bridging General	Bridging (checked) DLSw (checked)	Config>**p asrt** ASRT config>**enable br** ASRT config>**enable dls**	9
Protocols Bridging Interfaces	Interface 0 (TR slot 1 port 1) Bridging port (checked) Interface supports: SRB Segment number: 001 MTU size: 4399	('enable br' assumed) ASRT config>**disable transp 1** ASRT config>**enable source 1** ASRT config>**delete port 2**	10

Notes:

1. **add dev** defines a single port, not an adapter.
2. The configuration program assigns an interface number to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the **add dev** command for each port you want to use, and the interface number (also known as ″net number″) is the output of the command.
3. You need a write-capable SNMP community only if you want to download configuration files from the configuration program directly into the router. SNMP is not required to TFTP a configuration file to the router.
4. Only Interface 1 needs to be configured for IP for DLSw to function correctly in this example. Interface 0 is configured here for IP, solely for box management purposes.
5. You can also use RIP in place of OSPF.
6. We disable the forwarding of remote explorers as a general filter, to prevent backbone LAN traffic from generating DLSw search messages on the WAN links to remote sites. This means that all circuits must be initiated by the remote routers. If your network requires the host to be able to initiate connections out to the remote sites, change this parameter to ″forward to all DLSw peers″.

   If the remote routers are IBM routers, you can configure them individually to control which search messages they want to receive, using MAC address and NetBIOS name lists. You can also configure whether each will bring up its TCP connection to Network Utility all the time or drop it when unused, using the *connectivity setup type* parameter.
7. Having dynamic neighbors enabled is the default value, so you do not have to change this panel or issue this command. We show it here to point out that this is the parameter that allows remote DLSw partners (neighbors) to establish TCP connections to this Network Utility without you having to define their IP addresses here. Each remote router needs to be configured with this Network Utility's internal IP address (172.128.252.1) as its partner address.
8. SAPs do not need to be opened on Interface 1 since that interface is only carrying IP traffic and not LLC traffic.
9. ″enable br″ automatically creates TB bridge ports for both token-ring interfaces. Bridge port numbers are 1 and 2, and are independent of adapter port numbers.
10. The disable and enable commands change bridge port 1 from TB to SRB. The ″delete port″ command turns off bridging on interface 1 (bridge port 2). Bridging would be required on this interface if we needed to support local end station traffic bridging from the Campus Backbone to the host.

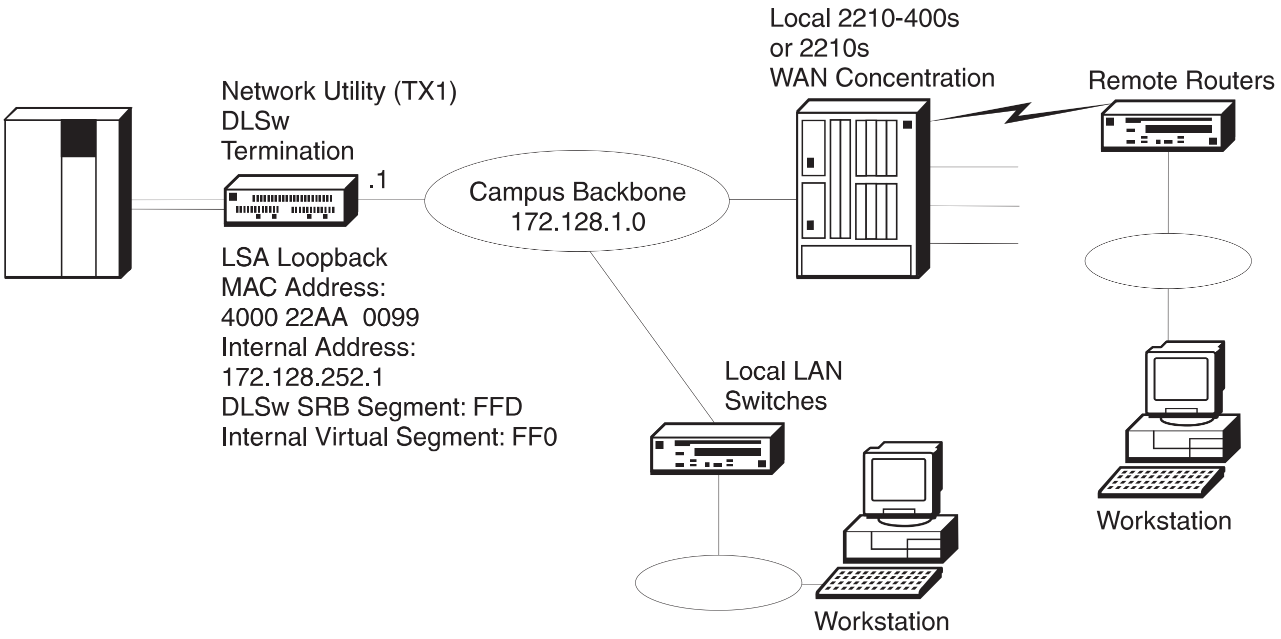


Figure 47. DLSw LAN Gateway

Table 77. DLSw LAN Gateway. See page 242 for a description and 255 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes
Devices Adapters Slots	Slot 1: 2-Port TR Slot 2: ESCON	See ″add device″ on next row	1

Devices
Adapters
Ports
	Slot 1/Port 1: Interface 0: TR
	Slot 2/Port 1: Interface 1: ESCON
		`Config>`**`add dev tok`**
		`Config>`**`add dev escon`**
			2

Devices
Interfaces
	Interface 0
	MAC address: 400022AA0001
		`Config>`**`net 0`**
		`TKR config>`**`set phy 40:00:22:AA:00:01`**	

Devices
Channel Adapters
ESCON Interfaces
ESCON Interfaces
	Base network number: 1
	Protocol type: LSA (do this first)
	Loopback (checked - do this second)
	LAN type: Token Ring
	Maximum data frame: 2052
	MAC address: 400022AA0099
		`Config>`**`net 1`**
		`ESCON Config>`**`add lsa`**
		(added as interface 2)
		`ESCON Add Virtual>`**`enable loopback`**
		`ESCON Add Virtual>`**`mac 40:00:22:AA:00:99`**
		`ESCON Add Virtual>`**`lan tok`**
		`ESCON Add Virtual>`**`maxdata 2052`**
		(continue in same session with next row)
			3

Devices
Channel Adapters
ESCON Interfaces
ESCON Subchannels
	Interface 2, Base net 1, Protocol LSA
	Device address: E4
	Subchannel type: read/write
	Link address: EF
		`ESCON Add Virtual>`**`subchannel add`**
		(cont'd)
		`ESCON Add LSA Subchannel>`**`device E4`**
		`ESCON Add LSA Subchannel>`**`link EF`**
		(Type **exit** twice and then **list all**)	

System
General
	System name: NUA_SC1C
	Location: XYZ
	Contact: Admin
		`Config>`**`set host`**
		`Config>`**`set location`**
		`Config>`**`set contact`**	

System
SNMP Config
General
	SNMP (checked)	`Config>`**`p snmp`**
		`SNMP Config>`**`enable snmp`**	

System
SNMP Config
Communities
General
	Community name: admin
	Access type: Read-write trap
	Community view: All
		`SNMP Config>`**`add community`**
		`SNMP Config>`**`set comm access write`**
			4

Table 77. DLSw LAN Gateway (continued). See page 242 for a description and 255 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes
Protocols IP General	Internal address: 172.128.252.1 Router ID: 172.128.1.1	Config>**p ip** IP config>**set internal** IP config>**set router-id**	
Protocols IP Interfaces	Interface 0 (TR slot 1 port 1) IP address: 172.128.1.1 Subnet mask: 255.255.255.0	IP config>**add address**	
Protocols IP OSPF General	OSPF (checked)	Config>**p ospf** OSPF Config>**enable ospf**	5
Protocols IP OSPF Area Configuration General	Area number: 0.0.0.0 Stub area (not checked)	OSPF Config>**set area**	
Protocols IP OSPF Interfaces	Interface 0 OSPF (checked)	OSPF Config>**set interface** Interface IP address **172.128.1.1** Attaches to area **0.0.0.0** (Accept other defaults)	
Protocols DLSw General General	DLSw (checked) SRB segment: FFD Forward explorers: local TCP connection only	Config>**p dls** DLSw Config>**enable dls** DLSw Config>**set srb** DLSw Config>**enable forward local**	6
Protocols DLSw General Dynamic Neighbors	Dynamic neighbors (checked)	DLSw Config>**enable dynamic**	7
Protocols DLSw TCP Connections	(add) Neighbor IP address: 172.128.252.1 (this is the router internal IP address)	DLSw Config>**add tcp** DLSw neighbor IP address: **172.128.252.1** (Accept other defaults)	8
Protocols DLSw Interfaces	Interface 0 (TR slot 1 port 1) SAP type: SNA (SAPs 0,4,8,C) Interface 2 (ESCON-LSA) SAP type: SNA (SAPs 0,4,8,C)	DLSw Config>**open 0 sna** DLSw Config>**open 2 sna**	9

Table 77. DLSw LAN Gateway (continued). See page 242 for a description and 255 for a diagram of this configuration.

Configuration Program Navigation	Configuration Program Values	Command-Line Commands	Notes

Protocols
Bridging
General
	General Tab:
	Bridging (checked)
	DLSw (checked)
	SRB Tab:
	Internal Virtual Segment: FF0
		`Config>`**`p asrt`**
		`ASRT config>`**`enable br`**
		`ASRT config>`**`enable dls`**
			10

Protocols
Bridging
Interfaces
	Interface 0 (TR slot 1 port 1)
	Bridging port (checked)
	Interface supports: SRB
	Segment number: 001
	MTU size: 2052
		('enable br' assumed)
		`ASRT config>`**`disable transp 1`**
		`ASRT config>`**`enable source 1`**
			11

Notes:

1. **add dev** defines a single port, not an adapter.
2. The configuration program assigns an interface number to all ports of an adapter automatically and you delete the ones you do not want to use. From the command line, you type the **add dev** command for each port you want to use, and the interface number (also known as ″net number″) is the output of the command.
3. The MAC address representing this LSA loopback interface is the target MAC address that all end stations in the DLSw network use to reach the host through this Network Utility.
4. You need a write-capable SNMP community only if you want to download configuration files from the configuration program directly into the router. SNMP is not required to TFTP a configuration file to the router.
5. You could choose to use RIP in place of OSPF.
6. We enable local forwarding to allow end stations on the local campus to reach the host. We disable the forwarding of remote explorers as a general filter, to prevent backbone LAN traffic from generating DLSw search messages on the WAN links to remote sites. This means that all remote circuits must be initiated by the remote routers. If your network requires the host to be able to initiate connections out to the remote sites, change this parameter to ″forward to all DLSw peers″.

   If the remote routers are IBM routers, you can configure them individually to control which search messages they want to receive, using MAC address and NetBIOS name lists. You can also configure whether each will bring up its TCP connection to Network Utility all the time or drop it when unused, using the *connectivity setup type* parameter.
7. Having dynamic neighbors enabled is the default value, so you do not have to change this panel or issue this command. We show it here to point out that this is the parameter that allows remote DLSw partners (neighbors) to establish TCP connections to this Network Utility without you having to define their IP addresses here. Each remote router needs to be configured with this Network Utility's internal IP address (172.128.252.1) as its partner address.
8. Adding the internal IP address as a neighbor is required to enable DLSw to carry traffic from the ESCON/LSA interface to the backbone LAN.
9. SAPs are opened on Interface 0 to enable the LLC flow to the local LAN switches, and are not required for remote DLSw to work.
10. ″enable br″ automatically creates a TB bridge port for the token-ring interface. The bridge port number is 1, and is independent of adapter port numbers and box interface numbers.
11. The disable and enable commands change bridge port 1 from TB to SRB. Bridging is required on this interface to support local end station traffic looping through DLSw from the Campus Backbone to the host.

Chapter 18. Sample Host Definitions

This appendix contains examples of host definitions for the Network Utility in the configurations used in this manual.

Specifically, definitions for the following environments are presented:

- LSA
- LCS
- MPC+

Additionally, the differences between a Network Utility with an ESCON channel adapter and a parallel channel adapter are highlighted.

For more information on defining the Network Utility to the host, refer to the *IBM 2216 Nways Multiaccess Connector Software User's Guide*, SC30-3886.

Overview

There are three steps to define a channel-attached Network Utility to the host:

1. Define the Network Utility to the host channel subsystem

   This will be done either from the I/O configuration program (IOCP) or the Hardware Configuration Definition (HCD), depending on your MVS version. (HCD requires MVS/ESA SP version 4.2 or later with APAR# OY67361.)

   The definition statements are slightly different for an ESCON channel-attached device than for a parallel channel-attached device. An example of these definitions is given in "Sample Host IOCP Definitions" on page 260.

2. Define the Network Utility as a control unit to the host operating system

   For most systems, the definitions are the same for an ESCON adapter as they are for a parallel channel adapter. Obviously, they depend on the operating system being used. An example of these definitions is given in "Defining the Network Utility in the Operating System" on page 263.

3. Define the Network Utility to the host TCP/IP or VTAM

   These definitions depend on whether you are defining an LSA (SNA), an LCS (TCP/IP), or an MPC+ (SNA and/or TCP/IP) interface on the Network Utility. Section "VTAM Definitions" on page 264 shows examples of the required VTAM definitions. Section "Host IP Definitions" on page 270 shows examples of the required TCP/IP definitions.

Definitions at the Channel Subsystem Level

You make definitions at this level via the IOCP or with HCD. If HCD is available, you will probably want to use it. HCD offers an improved method of defining system hardware configuration. With HCD several complex steps required for entering hardware configuration data can be accomplished using an interactive dialog. This chapter presents only the IOCP macros that would be generated from HCD.

Sample Host IOCP Definitions

An example of the definitions required in the host I/O Configuration Program (IOCP) for a Network Utility configured with an ESCON adapter is shown in Figure 48.

```
CHPID                PATH=((05)),TYPE=CNC
CNTLUNIT             CUNUMBR=1E0,PATH=05,CUADD=0,
                     UNITADD=((E0,32)),LINK=3C,UNIT=3172
IODEVICE             UNIT=3172,ADDRESS=((1E0,32)),
                     CUNUMBR=1E0
```

Figure 48. Sample Host IOCP Definitions for the Network Utility (ESCON)

The following sections describe the IOCP macros that you need for defining the Network Utility at the host.

RESOURCE Statement

This identifies the host logical partitions (LPARs) by name and number. This statement is not present if the host is not partitioned *as is the case in the example above*.

- PART=((name1,x),(name2,y)...(nameX,z))

  The name identifies the LPAR and is used in the rest of the channel path definition. The number is the corresponding LPAR number. The LPAR number is used in defining the subchannel on the Network Utility. If the host is not partitioned, the LPAR number is always 0.

Channel Path ID (CHPID) Statement

The CHPID identifies the type of channel connection and who uses it.

- PATH=x

  This uniquely identifies the channel path. This value is often called the ″CHPID number″.

- TYPE=CNC

  This indicates that the channel is an ESCON channel. The channel type is CNC for ESCON and BL for block multiplexor (Parallel Channel Adapter).

- SWITCH=x

  This identifies which ESCON Director is in this path. If no director is being used, this parameter is omitted.

- SHARED

  This indicates that the CHPID can be used by multiple LPARs simultaneously. If not present, only one LPAR can use the CHPID at a time.

- PARTITION=(name1,name2,...,nameX)

  This is one form of the PARTITION parameter and it contains an access list of LPARS that indicates which partitions have access to this channel. The names must be included in the RESOURCE statement.

- PARTITION=((name1,...,nameX),(name2,...,nameY))

  This is the other form of the PARTITION parameter. In this form, the first grouping of names is the access list of LPARs, as above. The second grouping is the list of candidate LPARs that an operator could configure to have access to the channel. The second grouping will have at least the same LPARs as the first grouping and it may specify additional LPARs also.

Control Unit (CNTLUNIT) Statement

This statement, along with the IODEVICE statement, defines the path from the host to the Network Utility. The CNTLUNIT and IODEVICE statements occur in pairs. If multiple LPARs are being defined to use a single CHPID, there must be a CNTLUNIT and IODEVICE statement for each LPAR.

- CUNUMBR=x

  This is an identifier for the control unit definition.
- PATH=x

  This number identifies the CHPID being used.
- UNIT=3172

  This identifies the type of control unit at the other end of the channel. The value is always 3172 when talking to a Network Utility. The IBM 3172 was the predecessor of the Network Utility ESCON channel function.
- CUADD=x

  This value identifies the control unit address of the Network Utility. The default is 0.
- UNITADD=((*addr*,*number*))

  This defines the range of addresses reserved for this control unit, where:

  addr is the hex address of the first subchannel assigned to this control unit

  number
  is the decimal number of subchannels being assigned to this control unit

  The example above defines a range of 32 control unit addresses, or subchannels, starting from E0 hex and going upwards. The device addresses specified on the Network Utility LCS, LSA, or MPC+ interface definition must be from within this range. The Network Utility can use a maximum of 64 subchannels.
- LINK=xx

  The value for the LINK parameter should be set to the port of the ESCON Director (ESCD) that the *Network Utility* is attached to. Because the ESCD is a switch, you can think of the link parameter as the phone number that the host will use to reach the Network Utility through the switch.

IODEVICE Statement

This statement, along with the CNTLUNIT statement, identifies the Network Utility connection to the host.

- ADDRESS=(*addr*,*number*)

  This parameter identifies the range of addresses to the rest of the host, where:

  addr is the hex address ***being assigned*** to the first address reserved

  number
  is the decimal number of subchannels reserved

  This address is different from the UNITADD. It is used in the TCP/IP profile (for LCS), the VTAM XCA Major Node Definition (for LSA), and the VTAM TRL (for MPC+) to identify the subchannels being used.
- CUNUMBR=x

This identifies the corresponding CNTLUNIT statement to this IODEVICE statement. While the value for this parameter has to be the same for both the CNTLUNIT and the IODEVICE macros, it does not have to relate to any other parameter. It is a good idea, however, to make it the same value specified in the ADDRESS parameter in the IODEVICE macro. The value for CUNUMBR has no significance outside the Channel Path Definition.

- UNIT=3172

  This identifies the type of device that is downstream. It should always be 3172 if the control unit is a Network Utility. The IOCP software in the host does not look at this field. If you are migrating from an IBM 3172 to the Network Utility, you might have a value of UNIT=SCTC in the existing IOCP statement. This should be changed to 3172 for the Network Utility.

- PARTITION=(name)

  This is the device candidate list and it contains a list of one or more LPARs that have access to the device. This list is a subset of the list of LPARs specified in the CHPID statement and it is used to restrict which LPARs in the channel candidate list are allowed to use these devices. If the host is not partitioned, this field will not be present.

Figure 49 shows an example of the IOCP statements for defining a Network Utility with a Parallel Channel Adapter (PCA).

```
CHPID                   PATH=((05)),TYPE=BL
CNTLUNIT                CUNUMBR=640,PATH=05
                        PROTOCL=S4,UNIT=3172
                        SHARED=N,UNITADD=((40,32))
IODEVICE                UNIT=3172,ADDRESS=((640,32))
                        STADET=N,CUNUMBR=640,TIMEOUT=Y
```

Figure 49. Sample Host IOCP Definitions for the Network Utility (PCA)

Note the following points concerning the IOCP statements for a Network Utility with a PCA.

- The TYPE is BL for Block Multiplexer
- PROTOCL parameter can be set to the following values, depending on the device capability:

  D Direct-Coupled Interlock (DCI) mode

  S Maximum 3.0 Mbps data streaming speed

  S4 Maximum 4.5 Mbps data streaming speed

  For the Network Utility, set the value to S4. The transfer mode and channel parameter must conform with the PCA setting for transfer mode and channel transfer speed.

- The UNIT parameter on the CNTLUNIT and IODEVICE statements must be set to 3172.
- When an ESCON Converter is the channel path, the CHPID TYPE parameter must be set to CVC, otherwise it is set to BL.

Defining the Network Utility in the Operating System

The following sections describe the definitions needed for various host operating systems.

Network Utility Definition for VM/SP

You must define the Network Utility to a VM/SP operating system by updating the real I/O configuration file (DMKRIO) with entries for the Network Utility in the RDEVICE and the RCTLUNIT macros. In the following example, 640 is the base unit address and the size of the address range is 32.

```
RDEVICE ADDRESS=(640,32),DEVTYPE=3088
RCTLUNIT ADDRESS=640,CUTYPE=3088,FEATURE=32-DEVICE
```

Network Utility Definition for VM/XA and VM/ESA

You must define the Network Utility to a VM/Extended Architecture (VM/XA or VM/ESA operating system by updating the real I/O configuration file (HCPRIO) with an entry for the Network Utility in the RDEVICE macro. In the following examples, 640 and 2A0 are base control unit addresses. The address range size, as defined in the UCW or IOCP, is 8 in both examples.

The following example is a VM/XA HCPRIO definition:

```
RDEVICE ADDRESS=(640,8),DEVTYPE=CTCA
```

The following example is a VM/ESA HCPRIO definition:

```
RDEVICE ADDRESS=(2A0,8),DEVTYPE=CTCA
```

Network Utility Definition for MVS/XA and MVS/ESA without HCD

You must define the Network Utility to an IBM Multiple Virtual Storage/Extended Architecture (MVS/XA) or MVS/ESA operating system by updating the MVS Control Program with an entry for the Network Utility in the IODEVICE macro.

For ESCON channels, an example IODEVICE macro is:

```
IODEVICE UNIT=3172,ADDRESS(540,8)
```

For parallel channels, an example IODEVICE macro is:

```
IODEVICE UNIT=CTC,ADDRESS(640,8)
```

The base control unit addresses are 540 and 640. The address range size, as defined in the UCW or IOCP, is 8 in both examples.

Network Utility Definition for MVS/ESA with HCD

The hardware configuration definition (HCD) component of MVS/ESA SP Version 4.2 and 4.3 with APAR #OY67361 offers an improved method of defining the system hardware configuration for Network Utility. You can accomplish the several complex steps required for entering hardware configuration data by using an interactive dialog with HCD.

The required configuration data for the Network Utility is:

- When using HCD with APAR # OY67361, define the Network Utility as (UNIT=3172). For example,

  ```
  IODEVICE UNIT=3172,ADDRESS(740,8)
  ```
- Without HCD, define the Network Utility for:
  - – Parallel channels as a 3088 device (UNIT = 3088 or CTC)

    ```
    IODEVICE       UNIT=CTC,ADDRESS(840,8)
    ```
  - – ESCON channels as a serial CTC device (UNIT = SCTC)

    ```
    IODEVICE       UNIT=SCTC,ADDRESS(A40,8)
    ```

Notes:

1. If you are using HCD for MVS Version 4 to define your ESCON host connection, you will need APAR # OY67361 to obtain the UIM support for the device definition (UNIT=3172).
2. When you are migrating your IOCP definition and operating system definitions to the HCD environment, it is important that you change all Network Utility device statements to device type (UNIT=3172).

Network Utility Definition for VSE/ESA

You must define the Network Utility to a VSE/ESA operating system by supplying an ADD statement for each channel unit address at initial program load (IPL) time. Code the device type on the ADD statement as CTCA, EML as shown in the following example:

```
ADD 640,CTCA,EML
```

The base control unit address is 640 in the example. For the number of channel unit addresses added, increment the IOTAB storage macro by this count.

VTAM Definitions

This section gives sample VTAM definitions for an XCA major node, an MPC+ local PU and Transport Resource List (TRL) major node, and an example of defining VTAM for APPN and DLUR support. It also shows an example of a switched major node for a PU in a TN3270 server. This section is not meant to be a complete reference on the subject. For more information on configuring VTAM, refer to the *CS OS/390 Resource Definition Reference*, SC31-8565.

VTAM XCA Major Node Definition

When defining a channel gateway using LSA to VTAM, a definition for an External Communications Adapter (XCA) is required. This definition is the same as that used for an IBM 3172. An example is shown in Figure 50 on page 265.

```
**********************************************************************
RAINETU VBUILD TYPE=XCA     1

**
**
RANETUP  PORT  ADAPNO=0,    2                          * X
               CUADDR=285,     3                       * X
               MEDIUM=RING,      4                     * X
               SAPADDR=4,          5                   * X
               TIMER=60
**
**********************************************************************
RANETUG1 GROUP DIAL=YES,CALL=INOUT,DYNPU=YES
*
RANETUL1 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP1 PU   ISTATUS=ACTIVE
RANETUL2 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP2 PU   ISTATUS=ACTIVE
RANETUL3 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP3 PU   ISTATUS=ACTIVE
RANETUL4 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP4 PU   ISTATUS=ACTIVE
RANETUL5 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP5 PU   ISTATUS=ACTIVE
RANETUL6 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP6 PU   ISTATUS=ACTIVE
RANETUL7 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP7 PU   ISTATUS=ACTIVE
RANETUL8 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP8 PU   ISTATUS=ACTIVE
RANETUL9 LINE ANSWER=ON,ISTATUS=ACTIVE
RANETUP9 PU   ISTATUS=ACTIVE
```

Figure 50. XCA Major Node Definition Sample for LSA Direct Connection

Notes:

1 `TYPE` must be `XCA`

2 `ADAPNO` is the LAN number for the Network Utility interface. This value is assigned to the Network Utility's LSA interface when it is created. The value can be obtained from the Network Utility by listing the configuration of the interface from the talk 6 menus, or it can be retrieved by entering the **list nets** command from the ESCON console in Talk 5. Note that a wrong value for this parameter is the single most common error in LSA configuration.

3 `CUADDR` specifies the subchannel to be used to communicate with the Network Utility. This value must be within the range of values specified in the IODEVICE statement in the IOCP definition.

4 This specifies the physical LAN topology to which the LSA interface is attached. This corresponds to the value specified for LANtype for the Network Utility interface. Valid values are `MEDIUM=RING` for Token-Ring, `MEDIUM=CSMACD` for Ethernet, and `MEDIUM=FDDI` for a Fiber Distributed Data Interface (FDDI) network.

5 `SAPADDR` is the Service Access Point number VTAM wishes to open on the Network Utility. Note that it is the SOURCE SAP, not the DESTINATION SAP. When more than one active XCA major node refers to the same LAN, all the XCA major nodes have to use different SAPs.

LINE Statement

The CALL field can be one of the following:

- IN means only remote devices may establish connections.
- OUT means only VTAM can initiate connections.
- INOUT connections may be initiated at either end.

If VTAM is going to dial out, the Switched Major Node definition must specify a destination with a PATH statement.

An asterisk in the first column indicates a statement has been commented out, and should be ignored. A character in the last column indicates the next line is a continuation of this line.

VTAM Definitions for an MPC+ Connection

An MPC+ connection requires entries in two VTAM control blocks:

- The Local Major Node
- The Transport Resource List (TRL) Major Node

Figure 51 shows a sample definition for a local SNA major node for a Network Utility MPC+ connection. This is the local PU that resides in VTAM that supports the channel connection defined in the TRL. The connection type must be APPN and you also need to enable HPR.

```
LOCNETU  VBUILD TYPE=LOCAL
MPCNETUP PU    TRLE=MPCNETU,
               XID=YES,
               CONNTYPE=APPN,
               CPCP=YES,
               HPR=YES
```

Figure 51. VTAM Local Major Node Definition

Notes:

1. TYPE must equal LOCAL on the VBUILD statement.
2. TRLE identifies the TRL being used. The name must match the name of an existing TRL.
3. XID indicates whether XIDs will be exchanged. It must be XID=YES.
4. CONNTYPE must be set to CONNTYPE=APPN since APPN is the only protocol that VTAM uses with an MPC+ connection.
5. CPCP specifies that CP-CP connections with APPN can be established over this MPC+ connection. This could be either set to YES or NO, depending upon your APPN topology.
6. HPR specifies that APPN HPR traffic can flow over this MPC+ connection. HPR is normally used by default, but setting this value to YES ensures it. This is important because an MPC+ connection requires RTP (and HPR).

Next, you need a transport resource list for the MPC+ connection from the Network Utility. An example definition is shown in Figure 52 on page 267.

```
        VBUILD TYPE=TRL
MPCNETU TRLE   LNCTL=MPC,
               MAXBFRU=9,
               READ=280,
               WRITE=281,
               MPCLEVEL=HPDT,
               REPLYTO=3.0
```

Figure 52. VTAM Transport Resource List (TRL) Definition

Notes:

1. TYPE must be TRL.
2. MPCNETU is the name that identifies the TRL. It must match what is specified in the TRLE= field in the local major node definition. (See Figure 51 on page 266.)
3. LNCTL identifies the connection type. It must be LCNTL=MPC.
4. MAXBFRU is the number of 4K pages per read subchannel.
5. READ/WRITE specifies the subchannels in the MPC+ group, and indicates their direction. The subchannel numbers must be in the range of addresses specified in the IODEVICE statement in the IOCP definition. There can be multiple READ and WRITE parameters in the TRLE statement but there must be at least one of each.

   Note: The designations READ and WRITE here are from the HOST perspective. In the Network Utility MPC+ definition, the designations are from the Network Utility perspective. Therefore, subchannels designated as READ on the host ***must*** be designated as WRITE on the Network Utility, and vice versa.
6. REPLYTO is the reply timeout value in seconds.

VTAM Definitions for APPN

If VTAM is configured for DLUS, then it must be an APPN network node. Configuring VTAM as an APPN network node requires certain parameters to be specified in the VTAM start-up parameters. These are shown in Figure 53 on page 268. Set the CONNTYPE to APPN and the NODETYPE to a Network Node (NN).

```
ASYDE=TERM,IOPURGE=5M,
CONFIG=I0,
CONNTYPE=APPN,
CPCP=YES,
CSALIMIT=0,
DYNADJCP=YES,
ENCRYPTN=NO,
GWSSCP=YES,
HOSTPU=ISTPUS18,
HOSTSA=18,
HPR=RTP,
NETID=USIBMRA,
NODETYPE=NN,
NOTRACE,TYPE=VTAM,IOINT=0
PPOLOG=YES
SORDER=APPN,
SSCPDYN=YES,
SSCPID=18,
SSCPNAME=RAI,
SSCPORD=PRIORITY,
SUPP=NOSUP,
TNSTAT,CNSL,
VRTG=YES
OSITOPO=LLINES,
OSIMGMT=YES
XNETALS=YES
```

Figure 53. VTAM Start-up Parameters

VTAM Static Definition of TN3270 Resources

VTAM definitions are required for the PUs used by the TN3270E Server. You need a switched major node definition for each PU in the TN3270E server. For example, each PU in the TN3270E server can support up to 253 LUs. If you need 500 3270 sessions, then you will need 2 PUs in the router and 2 PU definitions in VTAM.

Figure 54 shows an example of a VTAM switched major node definition for a TN3270E server PU that is connected via DLUR and APPN.

```
LOCNETU  VBUILD TYPE=SWNET
MNETUA  PU     ADDR=01,ISTATUS=ACTIVE,VPACING=0,                       *
               DISCNT=NO,PUTYPE=2,SSCPFM=USSSCS,USSTAB=US327X,         *
               IDBLK=077,IDNUM=02216,IRETRY=YES,MAXDATA=521,           *
               MAXOUT=7,MAXPATH=8,PASSLIM=7,PACING=0,ANS=CONTINUE
********************************************************************
PNETUA   PATH  PID=1,DLCADDR=(1,C,INTPU),DLCADDR=(2,X,07702216),       *
               DLURNAME=MNETUA
********************************************************************
JC7LU2   LU    LOCADDR=2
JC7LU3   LU    LOCADDR=3
JC7LU4   LU    LOCADDR=4
```

Figure 54. VTAM Definitions for a TN3270E Server PU (DLUR/APPN)

Figure 55 on page 269 shows an example of a VTAM switched major node definition for a TN3270E server PU that uses a subarea connection to the host.

```
LSAP08T VBUILD TYPE=SWNET
PUPS08T PU ADDR=01,IDBLK=077,IDNUM=12244,MAXOUT=7,PACING=0,VPACING=0,
              DLOGMOD=B22NNE,PUTYPE=ANY,
              SSCPFM=USSSCS,MAXDATA=2000,MODETAB=LMT3270
PT08LU2 LU LOCADDR=02,LOGAPPL=TSO
PT08LU3 LU LOCADDR=03,LOGAPPL=TSO
PT08LU4 LU LOCADDR=04,LOGAPPL=TSO
PT08LU5 LU LOCADDR=05,LOGAPPL=TSO
PT08LU6 LU LOCADDR=06,LOGAPPL=TSO
```

Figure 55. VTAM Definitions for a TN3270E Server PU (Subarea)

The following sections provide an overview of the statements in the Switched Major Node Definition.

VBUILD Statement

The TYPE field must be TYPE=SWNET.

PU Statement

This statement defines the type of data flow and the destination. The pertinent parameters are:

- ADDR is an identifier.
- MAXDATA is the maximum packet size VTAM will support over this interface. This value will be negotiated down with the Network Utility during the XID exchange.
- IDBLK/IDNUM identify the remote device when VTAM is communicating with PU 2.0 (dependent) devices.

LU Statement

These statements define the logical units (LUs) that can be contacted through this PU. The name on the left of each statement is the name that the host uses to address each LU. The LOCADDR is used by the Network Utility to identify the correct LU in VTAM.

PATH Statement

If VTAM is going to dial out, the Switched Major Node definition must specify a destination with a PATH statement. The path statement will be different depending on whether the TN3270E server attaches via a Subarea or a DLUR/APPN connection.

For a subarea connection, the format is:

```
PATH DIALNO=xxyyzzzzzzzzzzzz
```

where:

- *xx* is a place holder
- *yy* is the destination SAP number
- *zz* is the destination MAC address

The example in Figure 55 does not have a PATH statement because in this example, the downstream PU will contact VTAM instead of VTAM dialing out to the device.

The example in Figure 54 on page 268 shows a PATH statement for a TN3270E server PU that is using DLUR to connect to the host. Here, the PATH statement identifies the CP name of the Network Utility (MNETUA) via the DLURNAME parameter. This is needed in order for the LU6.2 conversation between the DLUR and DLUS to be established. Once this session has been established, the SSCP-PU session between VTAM and the TN3270E server PU will be established using the IDBLK/IDNUM value specified by DLCADDR=(2,X,07702216).

VTAM Dynamic Definition of TN3270 Resources

For the latest information, please refer to “Dynamic Definition of Dependent LUs” on page 146.

Host IP Definitions

Defining the Network Utility to the host for a TCP/IP connection requires you to make changes to the host TCP/IP profile. This section gives an overview of the relevant statements that need to be changed.

DEVICE Statement

This statement defines the subchannel pair being used by TCP/IP. The format is:

```
DEVICE  name  LCS  subchannel
```

where:

- *name* identifies the subchannel path being used. It has local significance only, and can be anything.
- *subchannel* identifies the even subchannel being used for this connection. This value comes from the IODEVICE statement in the IOCP definition. When specified, that subchannel and the next one are both being used.

A TCP/IP profile must contain one DEVICE statement for each subchannel pair being used.

LINK Statement

This statement identifies which LCS interfaces on the Network Utility are being used on a given subchannel pair. The format is:

```
LINK name lantype lannumber devicename
```

where:

- *name* identifies the LCS interface. It has local significance only, and can be anything.
- *lantype* identifies the type of LAN interface that the Network Utility LCS interface is emulating. The allowable values are:
  - IBMTR for Token-Ring
  - ETHERNET for Ethernet V2
  - 802.3 for Ethernet (IEEE 802.3)
  - ETHERor802.3 for either Ethernet format accepted
  - FDDI for FDDI

- *lannumber* identifies which LCS interface on the Network Utility is being used. The *lannumber* is generated sequentially for each *lantype* on the Network Utility when you add an LCS interface. The *lannumber* can be found by entering **list nets** from the ESCON console in Talk 5. Note that the *lannumber* is ***not*** the net number. Having the wrong *lannumber* is the single most common configuration error for an LCS interface.
- *devicename* correlates the LCS interface to a subchannel pair. It must match a previously defined DEVICE statement.

There can be multiple LINK statements associated with a single DEVICE statement. There must be an LCS interface on the Network Utility for each LINK statement.

HOME Statement

This statement specifies the IP address(es) of the host TCP/IP stack. The format is:

```
HOME     ipaddress1     link1
         ipaddress2     link2
```

where:

- *IpaddressX* specifies an IP address on the host.
- *LinkX* specifies which link is associated with this IP address.

There must be only one HOME address for each LINK statement. The HOME address must be in the same IP subnet as the IP address of the LCS interface in the Network Utility, but ***they must be different addresses***.

GATEWAY Statement

This statement identifies the IP routing information for the host. It is divided into three sections:

- Direct Routes are routes directly connected to the host. The subnet containing the Network Utility LCS interface is a direct route.
- Indirect Routes are routes that are accessible via routers. The subnets of the LANs on the Network Utility are indirect routes, for example.
- Default Route is the route to be used if the host doesn’t have a direct or indirect route to an IP address.

Direct Routes

The format for Direct Routes is:

```
network firsthop linkname pktsize submask subvalue
```

where:

- *network* is the non-subnetted part of the IP address.
- *firsthop* indicates the IP address of the next hop in the IP network. For Direct Routes, this should be an equal sign (=).
- *linkname* identifies which link the host should use to get to the addresses on this route. For routes accessible via the Network Utility, this should be the name from the LINK statement associated with the LCS interface on this subnet.
- *pktsize* is the maximum frame size to be used on the interface. It should be less than or equal to the packet size defined in the LCS configuration on the Network Utility. A value of DEFAULTSIZE indicates the default packet size will be used.

- *submask* specifies the subnet mask used on this link. The subnet mask should correspond to the subnet mask defined for the LCS interface in the IP configuration on the Network Utility. This field may also be set to HOST to identify a point-to-point connection. In this case, the network field should contain the full IP address of the LCS interface.
- *subvalue* specifies the subnetted part of the IP address, and together with the network field, should fully specify the IP subnet associated with this LCS interface.

Indirect Routes

The format for Indirect Routes is:

```
network firsthop linkname pktsize submask subvalue
```

where:

- *network* is the full address of the IP subnet.
- *firsthop* indicates the IP address of the next hop in the IP network. For Indirect Routes accessible via the Network Utility, this should be the IP address of the Network Utility LCS interface.
- *linkname* identifies which link the host should use to get to the addresses on this route. For routes accessible via the Network Utility, this should be the name from the LINK statement associated with the LCS interface on this subnet.
- *pktsize* is the same value as for Direct Routes.
- *submask* should either be 0 or blank if the network field contains the full subnet address.
- *subvalue* should be left blank if there is no subnet mask specified.

Default Routes

The format for Default Routes is:

```
network firsthop linkname pktsize submask subvalue
```

where:

- *network* should be DEFAULTNET.
- *firsthop* indicates the IP address of the next hop in the IP network. For Default Routes to the Network Utility, this should be the IP address of the Network Utility LCS interface.
- *linkname* identifies which link the host should use to get to the addresses on this route. For routes accessible via the Network Utility, this should be the name from the LINK statement associated with the LCS interface on this subnet.
- *pktsize* is the same value as for Direct Routes.
- *submask* should either be 0 or blank.
- *subvalue* should be blank.

START Statement

This statement causes the specified subchannels to be started. The format is:

```
START devicename
```

where *devicename* is the name on the DEVICE statement above.

There must be a START statement for every DEVICE statement if the customer wishes to activate the devices when TCP/IP is started. If the START statement is not here, the devices can be started using the OBEY file. Note that the name here is the one from the DEVICE statement, not the LINK statement. Note also that the Network Utility LCS interface will remain in the DOWN state until the START has been issued from TCP/IP.

Host TCP/IP Definitions for LCS

This section gives you examples of the above statements required if you are defining an LCS connection.

1. DEVICE statement:

   ```
   DEVICE LCS1 LCS 210
   ```

   where LCS1 is the device name being defined, LCS is the type of device, and 210 is the host read (Network Utility write) subchannel used for this definition.
2. LINK statement

   ```
   LINK ETHLCS1 802.3 0 LCS1
   ```

   where ETHLCS1 is the link name, 802.3 is the LAN type to which the LCS interface attaches on the Network Utility, 0 is the LAN number assigned by the Network Utility, and LCS1 is the name of the device (from the device statement above).

   Note: Remember that the LAN number is automatically assigned by the Network Utility when you define the LCS interface. You can obtain it by issuing a `list all` command from the ESCON Config> prompt in the talk 6 process on the Network Utility console.
3. HOME command

   ```
   HOME 9.24.106.72 ETHLCS1
   ```

   where 9.24.106.72 is the IP address of this LCS interface and ETHLCS1 is the name of the link.
4. GATEWAY command

   ```
   GATEWAY 9.24.106   9.24.106.1   ETHLCS1   4096  0
   ```

   where 9.24.106 is the IP address for the network, 9.24.106.1 is the IP address of the default router, ETHLCS1 is the link name defined by the LINK statement above, 4096 is the MTU size, 0 is the subnet mask, and the subnet value has been left blank.
5. Activate the TCP/IP profile

   To activate the device defined in step 1, issue the following command:

   ```
   start lcs1
   ```

Host TCP/IP Definitions for MPC+

The steps for configuring TCP/IP in the host for an MPC+ connection are the same as for an LCS connection. However, the command syntax for the device and link commands is slightly different. For an MPC+ connection, the syntax for the device command is:

```
DEVICE IPTRL1 MPCPTP
```

where IPTRL1 is the name of the TRL that this connection will use and MPCPTP specifies an MPC point-to-point link.

To define the link, the syntax is:

```
LINK LINK1 MPCPTP IPTRL1
```

where LINK1 is the link name and the other two parameters are the same as those used in the device statement.

Chapter 19. Virtual Private Networks

The Internet has become a popular, low-cost backbone infrastructure. Because of its universal reach, many companies have considered constructing a secure virtual private network (VPN) over the public Internet. The challenge in designing a VPN for today's global business environment will be to exploit the public Internet backbone for both intra-company and inter-company communication, while still providing the security and reliability of the traditional private, self-administered corporate network.

This chapter defines a VPN and explains the benefits of implementing a VPN. It also discusses security considerations and planning aspects and describes VPN solutions available in the market today.

VPN Introduction and Benefits

With the explosive growth of the Internet, companies are beginning to ask, *What is the best way for us to exploit the Internet for our business?* Initially, companies were providing corporate Web sites to promote their company's image, products and services. Today the Internet potential is limitless, and the focus has shifted to e-business—using the global reach of the Internet for easy access to key business applications and data that reside in traditional I/T systems. Companies can now securely and cost-effectively extend the reach of their applications and data across the world through the implementation of secure VPN solutions.

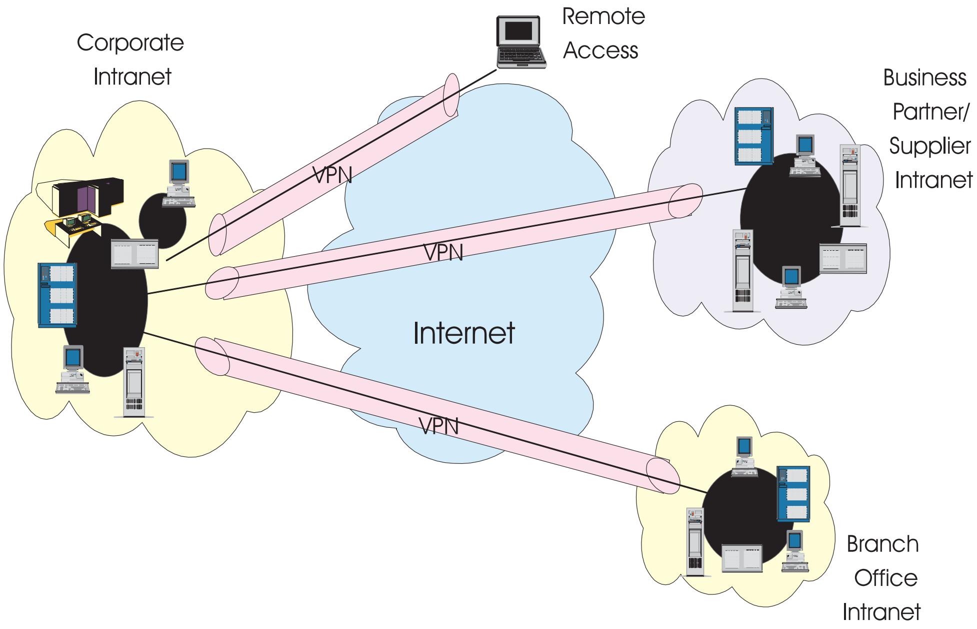


Figure 56. Virtual Private Networks

A VPN is an extension of an enterprise's private intranet across a public network such as the Internet, that creates a secure private connection, essentially through a private tunnel. VPNs securely convey information across the Internet by connecting remote users, branch offices and business partners into an extended corporate network, as shown in Figure 56. Internet Service Providers (ISPs) offer cost-effective access to the Internet (via direct lines or local telephone numbers), enabling companies to eliminate their current, expensive leased lines, long distance calls and toll-free telephone numbers.

The IETF’s IP Security Framework

Within the layered communications protocol stack model, the network layer (IP in the case of the TCP/IP stack) is the lowest layer that can provide end-to-end security. Network-layer security protocols provide blanket protection for all upper-layer application data carried in the payload of an IP datagram, without requiring a user to modify the applications.

The solutions are based on the IP Security architecture (IPSec) open framework, defined by the IPSec Working Group of the IETF. IPSec is called a framework because it provides a stable, long lasting base for providing network layer security. It can accommodate today’s cryptographic algorithms, and can also accommodate newer, more powerful algorithms as they become available. IPv6 implementations are required to support IPSec, and IPv4 implementations are strongly recommended. In addition to providing the base security functions for the Internet, IPSec furnishes flexible building blocks from which robust, secure VPNs can be constructed.

The IPSec Working Group has concentrated on defining protocols to address several major areas:

- Data origin authentication: verifies that each datagram was originated by the claimed sender and cannot be repudiated
- Data integrity: verifies that the contents of the diagram were not changed in transit, either deliberately or due to random errors
- Data confidentiality: conceals the clear text of a message, typically by using encryption
- Replay protection: ensures that an attacker cannot intercept a datagram and play it back at some later time without being detected
- Automated management of cryptographic keys and Security Associations (SAs): ensures that a company’s VPN policy can be conveniently and accurately implemented throughout the extended network with little or no manual configuration (these functions make it possible for a VPN’s size to be scaled to whatever size a business requires.)

The following list presents the principal IPSec protocols:

- IP Authentication Header (AH) provides data origin authentication, data integrity, and replay protection.
- IP Encapsulating Security Payload (ESP) provides data confidentiality, data origin authentication, data integrity, and replay protection.
- Internet Security Association and Key Management Protocol (ISAKMP) provides a method for automatically setting up security associations and managing their cryptographic keys.
- Oakley provides the cryptographic key management protocol used by ISAKMP.
- Internet Key Exchange (IKE) with shared key or digital signatures brings automation to key management, so that no manual key generation is required. During the Phase 1 negotiations, the cryptographic keys are exchanged and parties authenticate each other’s identity. At this point in time, ISAKMP function

uses Oakley cryptographic key management protocol to protect the ISAKMP messages exchanged between routers, in preparation for a secure data exchange.

- Public Key Infrastructure (PKI) is an arrangement by Certificate Authority (CA) that distributes and verifies keys for users.
- Certificate is a data area that binds each network user's encoded ID (digital signature) to its public/private key.
- Digital signature is a data area containing a user's encoded ID, which becomes part of a certificate.

Authentication Header

The IP Authentication Header (AH) provides connectionless integrity (that is, per-packet) and data origin authentication for IP datagrams. It also offers protection against replay. Data integrity is ensured by the checksum generated by a message authentication code (for example, MD5); data origin authentication is ensured by including a secret shared key in the data to be authenticated; and replay protection is provided by use of a sequence number field within the AH header. In the IPSec vocabulary, these three distinct functions are lumped together and simply referred to by the name authentication.

AH authenticates as much of the IP datagram as possible. Some fields in the IPheader change en-route and their value cannot be predicted by the receiver. These fields are called mutable and are not protected by AH. The mutable IPv4 fields are:

- Type of Service (TOS)
- Flags
- Fragment Offset
- Time to Live (TTL)
- Header Checksum

AH is identified by protocol number 51, assigned by the IANA. The protocol header (IPv4, IPv6, or Extension) immediately preceding the AH header contains this value in its Protocol (IPv4) or Next Header (IPv6, Extension) field.

AH processing is applied only to non-fragmented IP packets. However an IP packet with AH applied can be fragmented by intermediate routers. In this case, the destination first reassembles the packet and then applies AH processing to it. If an IP packet that appears to be a fragment (offset field is non-zero, or the More Fragments bit is set) is input to AH processing, it is discarded. This prevents the so-called overlapping fragment attack, which misuses the fragment reassembly algorithm in order to create forged packets and force them through a firewall.

Packets that failed authentication are discarded and are never delivered to upper layers. This mode of operation greatly reduces the chances of successful denial of service attacks, which aim to block the communication of a host or gateway by flooding it with bogus packets.

AH can be used in two modes: transport mode and tunnel mode. The transport mode is used by hosts instead of gateways. Gateways are not even required to support transport mode. Transport mode requires less processing overhead, but the mutable fields are not authenticated.

When protection of the IPv4 fields is required, tunneling should be used. The payload of the IP packet is considered immutable and is always protected by AH.

The tunnel mode is used whenever either end of an SA is a gateway. Thus, between two firewalls, the tunnel mode is always used. Although gateways are required to support tunnel mode only, often they also support transport mode. This mode is allowed when the gateway acts as a host, such as in cases when traffic is destined to itself. Examples are SNMP commands or ICMP echo requests.

In tunnel mode, the outer headers' IP addresses do not need to be the same as the inner headers' addresses. For example, two security gateways may operate an AH tunnel which is used to authenticate all traffic between the networks they connect together. This is a very typical mode of operation. Hosts are not required to support tunnel mode, but often they do.

Tunnel mode offers total protection of the encapsulated IP datagram and the possibility of using private addresses. However, there is an extra processing overhead associated with this mode.

Note: The original AH specification in RFC 1825 does not list tunnel mode as a requirement. Because of this, there are IPSec implementations based on that RFC that do not support AH in tunnel mode. This has implications in the ability to implement certain scenarios.

IP Encapsulating Security Payload

The IP Encapsulating Security Payload (ESP) provides data confidentiality (encryption), connectionless integrity (that is per-packet), data origin authentication and protection against replay. ESP always provides data confidentiality, and can also optionally provide data origin authentication, data integrity checking, and replay protection. If you compare ESP to AH, you will see that only ESP provides encryption, while either can provide authentication, integrity checking and replay protection.

When ESP is used to provide authentication functions, it uses the same algorithms used by the AH protocol. However, the coverage is different.

Combining the Protocols

Either ESP or AH may be applied alone, in combination with the other or even nested within another instance of itself. With these combinations, authentication and/or encryption can be provided between a pair of communicating hosts, between a pair of communicating firewalls or between a host and a firewall.

Internet Key Exchange (IKE)

An SA contains all the relevant information that communicating systems need in order to execute the IPSec protocols, such as AH or ESP. For example, an SA will identify the cryptographic algorithm to be used, the keying information and the identities of the participating parties. ISAKMP defines a standardized framework to support negotiation of SAs, initial generation of all cryptographic keys and subsequent refresh of these keys. Oakley is the mandatory key management protocol that is required to be used within the ISAKMP framework. ISAKMP supports automated negotiation of SAs and automated generation and refresh of cryptographic keys. The ability to perform these functions with little or no manual configuration of machines will be a critical element as a VPN grows in size.

Secure exchange of keys is the most critical factor in establishing a secure communications environment. No matter how strong your authentication and encryption are, if your key is compromised, they are worthless. Since the ISAKMP procedures deal with initializing the keys, they must be capable of running over links where no security is assumed. That is, they are used to bootstrap the IPSec protocols. Hence, the ISAKMP protocols use the most complex and processor-intensive operations in the IPSec protocol suite.

ISAKMP requires that all information exchanges are both encrypted and authenticated. No one can eavesdrop on the keying material, and the keying material will be exchanged only among authenticated parties.

VPN Customer Scenarios

This section discusses three of the most likely business scenarios that are well suited to the implementation of a VPN solution:

- Branch office connection network
- Business partner/supplier network
- Remote access network

The following sections provide a brief overview of each of these scenarios.

Branch Office Connection Network

The branch office scenario securely connects two trusted intranets within your organization. Your security focus is now on both protecting your company's intranet against external intruders and securing your company's data while it flows over the public Internet. This differs from the business partner/supplier network discussed next, where the focus is on enabling your business partners/suppliers access to data in your corporate intranet.

Suppose corporate headquarters wants to minimize the costs incurred from communicating with its own branches. Today, the company may use switched and/or leased lines, but it wants to explore other options for transmitting their internal confidential data that will be less expensive, more secure and globally accessible. By exploiting the Internet, branch office connection VPNs can easily be established to meet the company's needs.

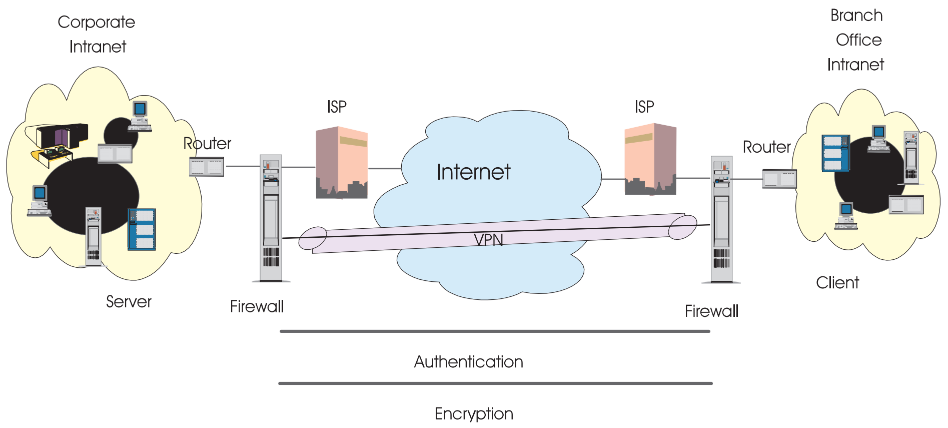


Figure 57. Branch Office Connection Network

As shown in Figure 57, one way to implement this VPN connection between the corporate headquarters and one of its branch offices is for the company to purchase Internet access from an ISP, such as IBM Global Services. IBM eNetwork routers or firewalls with integrated firewall functionality, or in some cases an IBM server with IPSec capability would be placed at the boundary of each of the intranets to protect the corporate traffic from Internet hackers. With this scenario, the clients and servers need not support IPSec technology, since the IPSec-enabled routers (or firewalls) would be providing the necessary data packet authentication and encryption. With this approach, any confidential information would be hidden from untrusted Internet users, with the router or firewall denying access to potential attackers.

With the establishment of branch office connection VPNs, the company's corporate headquarters will be able to communicate securely and cost-effectively with its branches, whether located locally or far away. Through VPN technology, each branch can also extend the reach of its existing intranet to incorporate the other branch intranets, building an extended, enterprise-wide corporate network.

This company can also easily expand this newly created environment to include its business partners, suppliers, and remote users through the use of open IPSec technology.

Business Partner/Supplier Network

Industry-leading companies are those that can communicate inexpensively and securely to their business partners, subsidiaries, and vendors. Many companies have chosen to implement switched and/or purchase leased lines to achieve this interaction. But this is often expensive, and geographic reach may be limited. VPN technology offers an alternative for companies to build a private and cost-effective extended corporate network with worldwide coverage, exploiting the Internet or other public network.

Suppose you are a major parts supplier to a manufacturer. Since it is critical that you have the specific parts and quantities at the exact time required by the manufacturing firm, you always need to be aware of the manufacturer's inventory status and production schedules. Perhaps you are handling this interaction manually today, and have found it to be time consuming, expensive and maybe even inaccurate. You would like to find an easier, faster and more effective way of communicating. However, given the confidentiality and time-sensitive nature of this information, the manufacturer does not want to publish this data on their corporate Web page or distribute this information monthly via an external report.

To solve these problems, the parts supplier and manufacturer can implement a VPN, as shown in Figure 58. A VPN can be built between a client workstation, in the parts supplier's intranet or directly to the server residing in the manufacturer's intranet. The clients can authenticate themselves either to the router or firewall protecting the manufacturer's intranet, directly to the manufacturer's server (validating that they are who they say they are), or to both, depending on your security policy. Then a tunnel could be established, encrypting all data packets from the client, through the Internet, to the required server.

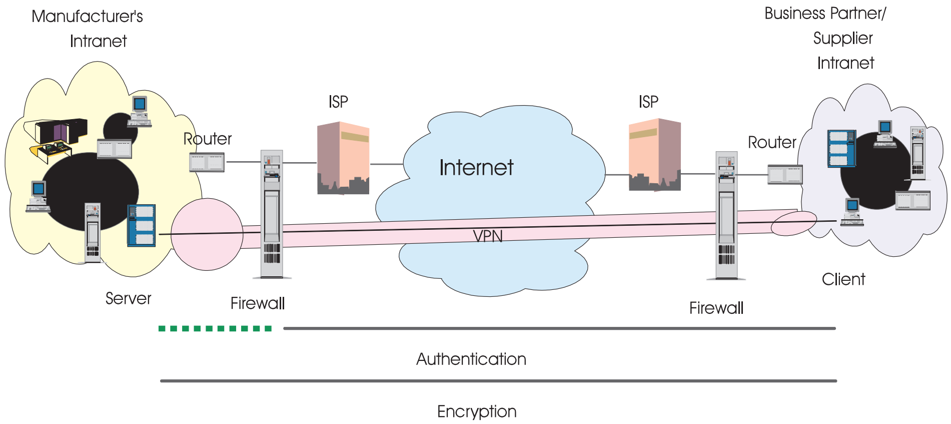


Figure 58. Business Partner/Supplier Network

One way to implement this scenario is for the companies to purchase Internet access from an ISP, such as IBM Global Services. Then, given the lack of security of the Internet, either an IPSec-enabled router, an IBM eNetwork firewall or an IBM server with IPSec capability can be deployed as required to protect the intranets from intruders. If end-to-end protection is desired, then both the client and server machines need to be IPSec-enabled as well.

Through the implementation of this VPN technology, the manufacturer would easily be able to extend the reach of their existing corporate intranet to include one or more parts suppliers (essentially building an extended corporate network) while enjoying the cost-effective benefits of using the Internet as their backbone. Now, with the flexibility of open IPSec technology, the ability for this manufacturer to incorporate more external suppliers is limitless.

When implementing a VPN, a set of security configuration criteria must be established. Decisions such as which security algorithms are to be used by each IPSec-enabled box and when the keys are to be refreshed are all aspects of policy management. With respect to key technology, almost all of today's currently popular security protocols begin by using public key cryptography. Each user is assigned a unique public key. Certificates, in the form of digital signatures, validate the authenticity of your identity and your encryption key.

These certificates can be stored in a public key database, such as a secure DNS, that can be accessible via a simple protocol, such as LDAP.

Remote Access Network

A remote user, whether at home or on the road, wants to be able to communicate securely and cost-effectively back to his/her corporate intranet.

Although many still use expensive long-distance and toll-free telephone numbers, this cost can be greatly minimized by exploiting the Internet. For example, you are at home or on the road, but need a confidential file on a server within your intranet. By obtaining Internet access in the form of a dial-in connection to an ISP such as IBM Global Services, you can communicate with the server in your intranet and access the required file.

One way to implement this scenario is to use an eNetwork VPN IPSec-enabled remote client and router or firewall, as shown in Figure 59. The client accesses the Internet via dial-up to an ISP, and then establishes an authenticated and encrypted tunnel between itself and the router or firewall at the intranet boundary.

By applying IPSec authentication between the remote client and the router or firewall, you can protect your intranet from unwanted and possibly malicious IP packets. And by encrypting traffic that flows between the remote host and the router or firewall, you can prevent outsiders from eavesdropping on your information.

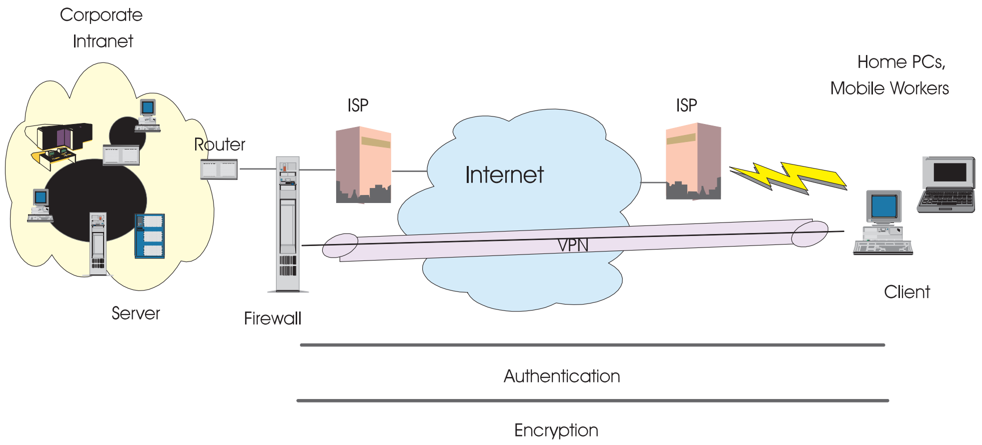


Figure 59. Remote Access Network

The three scenarios discussed in this section are the basis for the IPSec implementation and configuration examples described in this book. The next chapter provides step-by-step procedures for configuring IPSec tunnels with IBM Routers.

Policy Based Networking

Policy based networking is an architecture whereby the network devices determine if an action other than routing should be applied to received traffic. For example, should the traffic be secured by IP Security, or does the traffic have special Quality of Service (QoS) requirements? In order for network devices to make decisions based on policy, they need to be configured to do so. It is the configuration of multiple devices that is difficult to scale to a large network. Managing these multiple configurations can be made easier by developing a method of storing, retrieving, and distributing common configuration objects. The accepted method of storing, searching, and sharing configuration data is in a policy database.

In order to develop a policy, one must first define the requirements, such as traffic security and performance. An example of a traffic handling requirement is: *Traffic going between a branch office across the Internet to the corporate office should be secured*. This requirement is used to define what is known as the ***policy***. For the previously mentioned requirement, the network device must be able to identify which of the packets that it receives are coming from the branch office and are destined for the corporate office. A ***profile*** is created based on attributes such as IP source and destination addresses or protocols to match against the received packets. If the attributes of the packet match the attributes of the profile, then the packet is handled in a manner prescribed by an ***action*** definition. As one might assume, actions define such things as encryption and QoS methods.

All of the policies, profiles, and actions can be stored in a database. By using a database, certain profiles and actions can be reused and combined in different ways to create multiple policies without individually creating each policy in the device configuration. Change management is facilitated by the ability to make a single change in an object that will be propagated to any policy which uses that object. For example, multiple VPN tunnels could be using a common encryption method. If it were desired to change the encryption method for all tunnels, only one change would need to be made.

Four protocols will be able to use the policy database. These protocols are ones with repetitive data—repetitive within a device and possibly across the network. The following protocols are supported:

- RSVP
- DiffServ
- IKE
- IPSec

Every policy has a traffic profile and an availability period. You can specify that policy only applies to traffic arriving and leaving by specific interfaces. It is only necessary to identify the users if you are going to specify the IKE action.

An IPSec action may specify a drop, pass, or secure action. If the action is drop then all packets matching this policy are dropped. If the action is pass with no security, then all the packets are passed in the clear text. On the other hand, if the action is pass with security then all the packets are secured by means of an SA specified by this action. The IPSec action also contains the IP addresses of the tunnel end-points for the IPSec tunnel and IKE SAs.

Manually Defined Policies

One option for entering policies into the database is to manually configure each device. On IBM routers, the command line interface (talk 6) or configuration program is used to add objects such as policies, profiles, validity periods, IPSec actions and other security and performance related objects. As previously mentioned, some of these objects can be reused with different policies which reduces the amount of manual configuration. This may be acceptable and even desirable in a small network, but this method does not work well in large networks.

Policies from an LDAP Server

An alternative solution to configuring each network device is to input all policies into a central server and distribute the policies to the devices. The IETF has proposed that the central server is an LDAP server and that each network device is an LDAP client. For now, it is enough to understand that the LDAP server can store and distribute policies. In Figure 60, the policies on the right side are distributed by the LDAP Server.

For further discussion of this topic, point your Web browser to the following address:

`http://www.networking.ibm.com/support/networkutility/downloads`

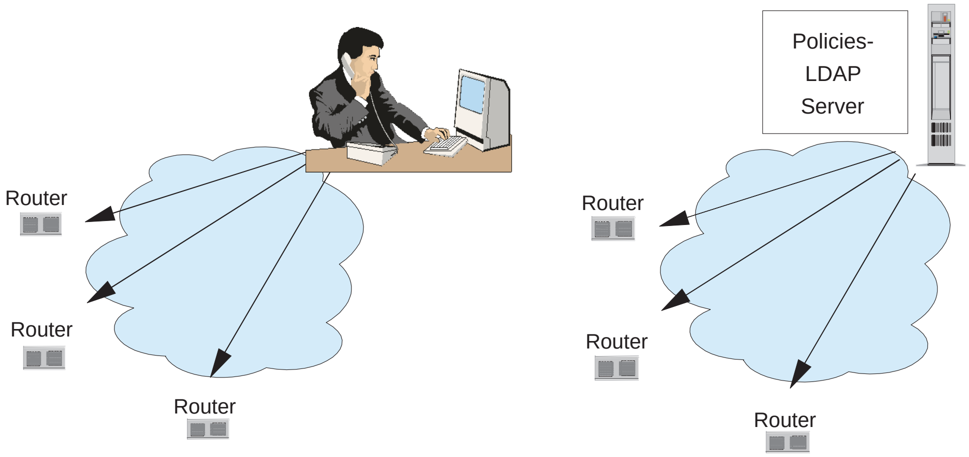


Figure 60. Policy distribution manually and by LDAP Server

IKE

IKE addresses concerns with manual tunnels for IPSec. Manual tunnels require difficult manual configuration of SA characteristics and keys.

IKE, previously known as ISAKMP/Oakley Key Resolution, defines a standardized framework to support automated negotiation of an SA, initial generation of all cryptographic keys, and subsequent refreshes of these keys. The negotiated SAs and keying materials are used to protect IKE exchanges and also other security functions, such as AH and ESP. Secure exchange of keys is the most critical factor in establishing a secure communications environment. Since IKE deals with initializing keys, it must be capable of being used over links where no security can be assumed. The IKE protocols use the most complex and processor-intensive operations in the IPSec protocol suite.

IKE requires that all information exchanges are both encrypted and authenticated. It has also been designed to protect against several well-known exposures:

- **Denial of Service:** The messages are constructed with unique cookies that can be used to quickly identify and reject invalid messages without the need to execute processor-intensive cryptographic operations.

- **Man-in-the-middle:** Protection is provided against the common attacks such as deletion of messages, modifications of messages, reflecting messages back to the sender, replaying of old messages and redirection of messages to unintended recipients.
- **Perfect Forward Secrecy (PFS):** Compromise of past keys provides no useful clues from breaking any other key whether it occurred before or after the compromised key.

IKE Pre-Shared Key and Digital Certificates

IKE has two phases. In phase I, the cryptographic operations are the most processor-intensive, since this phase is designed to exchange a ″master secret″ when there is no security in place. The master secret is used to derive the keys which will be used to protect users' traffic. Phase I is only concerned with establishing the protections suite for IKE messages themselves; it does not establish any SAs of keys for protecting user data. Phase I operations need only be done infrequently, and a single phase I negotiation can be used to support multiple phase II exchanges. Figure 61 shows the messages exchanged in Phase I. These 6 messages show the exchanges between two pairs in Main-mode.

Main Mode:

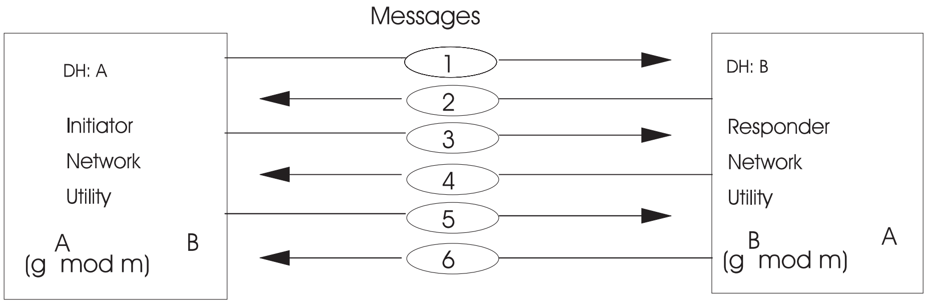


Figure 61. Phase I Message Exchange In IKE Main Mode

Message 1 is sent by the IKE peer that wishes to establish an ISAKMP tunnel. The first message is comprised of a standard IP header and UDP header. All ISAKMP messages are carried in a UDP packet with destination port 500. The UDP payload is comprised of an ISAKMP header, an SA payload and one or more proposal and transform payloads.

Message 2 contains the single proposal and transform that the responder wishes to accept.

Message 3 and message 4 exchange information from which the cryptographic keys will eventually be derived. All the information is exchanged in the clear. The messages contain a key exchange payload and the nonce payload. The key exchange payload contains the Diffie-Hellman (DH) public value. The exponent is the DH private value which is always kept secret. The nonce payload carries a large random number which is generated according to very strict mathematical guidelines. This payload is used to guarantee the connection exists and protect against replay attacks.

Both IKE devices now have each other's public DH values and their own keys. They can perform the DH calculation to generate a shared secret. The shared secret is the DH public value to the power of the private key. In Figure 61, the private DH values are A and B. In this case, the shared secret is a number equal in both routers which derived by using DH values A and B. The value of *g* is already agreed by the routers in messages 1 and 2.

From this point on, the keys can be generated with the information which is already exchanged and established. Now both the routers know:

- Two nonce values, N-i and N-r
- Its own private DH value
- Its partner’s public DH value, pk-i and pk-r
- The initiator and responder cookies
- The agreed hashing algorithm
- The shared secret—the result of the DH calculation

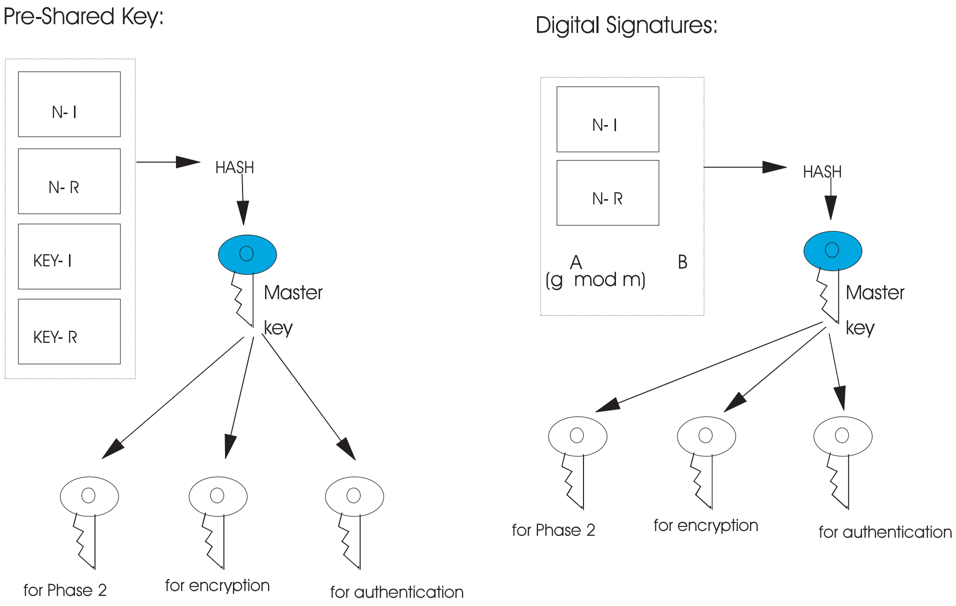


Figure 62. Generating the Keys

As seen in Figure 62, both devices now generate a master key, which is referred to as the SKEYID. This is the keying material for which the actual cryptographic keys will be derived. The method of generating the master keys depends on the authentication message agreed in message 2. The options are:

Pre-shared keys
The master key is derived from hashing the pre-shared key with the nonce from the initiator (N-I) and nonce from the responder (N-R).

Digital signature
The master’s key is derived from hashing the shared secret (result of the DH calculation) with the nonce from the initiator (N-I) and nonce from the responder (N-R).

The purpose of message 5 is to allow the responder to authenticate the initiator and message 6 allows the initiator to authenticate the responder. The format of the messages depends on whether the IKE peers have agreed to do authentication via pre-shared keys or digital signatures.

At this point, the phase I messages are complete in the main-mode. Each peer has authenticated itself to its peer, both have agreed on the characteristics of the ISAKMP SA and have derived the same set of keying material.

Aggressive Mode:
The other approach in phase I is the aggressive mode. Aggressive mode is less processor intensive, with only 3 message exchanges rather than six. However, aggressive mode is less secure.

The message 1 in aggressive mode is similar to message 1 of main mode in that it offers the peer a choice of ISAKMP SAs. It also includes the key exchange payload, the nonce payload and identity payloads. These would have been sent in messages 3 and 5 in main mode. This means that for aggressive mode, the identity of the initiator is sent in the clear, unlike in main mode when it is encrypted.

Message 2 in aggressive mode is the responder indicating which of the ISAKMP SAs he wishes to accept. In the response, he also includes the payloads that would have been present in message 2, message 4 and message 6 of main mode.

This means that the identity of the responder, his certificate and signature is sent in the clear, if authentication is via digital signatures. If authentication is via pre-shared keys, the identity and hash payloads are carried in the clear. Remember than in message 6 of main mode, the authentication material would have been encrypted before sending.

Message 3, which is encrypted, is sent to allow the responder to authenticate the initiator. The initiator sends the hash payload (pre-shared keys) or certificate and signature payload (digital signature mode) to the responder. The contents of the payloads is described in the discussion of main mode. The responder would then use the provided information to authenticate the initiator as described for message 5 in main mode.

Phase II exchanges negotiate the SAs and keys that will be used to protect user data exchanges. Phase II IKE messages are protected by the IKE SA generated in phase I. Phase II negotiations generally occur more frequently than phase I, typically once every few minutes, whereas phase I can be as far apart as once every day.

In phase II, message exchange starts with offers from the initiator. In message 1, as seen in Figure 63 on page 288, the initiator offers some options and information to calculate the shared-key. These are public DH (pk-i), initiator nonce values which are a big random number (N-i). All of this information is transferred in encrypted format.

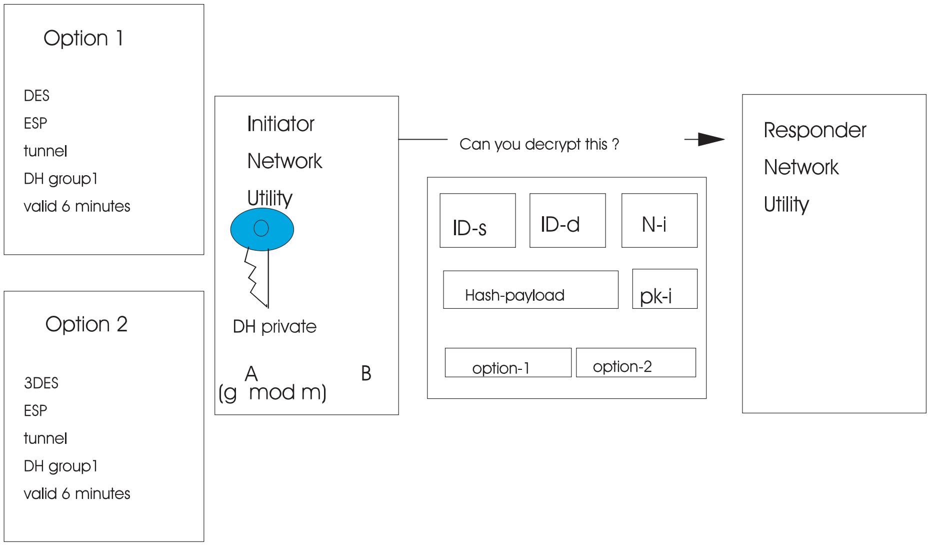


Figure 63. Phase II Message 1

In message 2, the responder provides the similar information (N-r, pk-r), to the initiator in addition to them. It selects one of the options which were provided to itself.

Now each Network Utility knows the following about the others:

- Each other's nonces, N-I and N-R
- Each other's public keys (i.e. public DH values)
- The security parameters index (SPI): an arbitrary 32-bit value that uniquely identifies one of the two security associations (inbound or outbound).
- The agreed protocols
- SKEYID_d calculated in phase 1. This is the hash of the master key, the two cookies and the DH shared secret.

The DH calculation is performed to generate the phase 2 shared secret. The keying material for traffic going from the responder to the initiator is the hash of SKEYID_d, the shared secret, protocol, SPI of the initiator, and two nonces.

All the necessary keying material has now been exchanged. Message 3 proves the connection is alive.

Tunneling Protocols

Refer to *Nways Multiprotocol Access Services Using and Configurating Features* for further explanation of the following protocols.

Layer 2 Tunneling

Layer 2 Tunneling Protocol (L2TP) is the IETF standard's track protocol for tunneling Point-to-Point Protocol (PPP) traffic across an IP network. L2TP uses a UDP transport for both tunnel setup messages and to transport PPP data between endpoints. L2TP is a client-server architecture with a client called a L2TP Access Concentrator (LAC) and the server called a L2TP Network Server (LNS).

Layer 2 Forwarding

Layer 2 Forwarding (L2F) is a tunneling protocol that was developed by Cisco Systems, Inc. It provides the same solutions as L2TP. When the IETF developed L2TP, they reused some of Cisco's L2F and Microsoft's PPTP. IBM routers implement L2F to interoperate with Cisco routers. Between the IBM routers, L2TP is used because it is an IETF standard.

L2F defines two devices—a NAS and a home gateway.

Point-to-Point Tunneling Protocol

Point-to-Point Tunneling Protocol (PPTP) has the same aim as L2TP: to tunnel PPP packets across an IP network. L2TP was developed by the IETF and is based heavily on PPTP and L2F (Cisco's equivalent). To establish a tunnel with a Microsoft device, IBM routers need to support PPTP.

PPTP is a client-server architecture with a client called PPTP network access concentrator (PAC) and the server called a PPTP network server (PNS). According to the architecture, the PAC is typically a workstation and the PNS, a server.

Voluntary Tunneling with PPTP

Voluntary tunneling is a client-initiated model. The client/PAC dials into the NAS, gets an IP address and establishes regular network access. Afterwards, it opens another dial-up session which establishes the PPTP tunnel. There are two scenarios in which IBM routers can be used with a voluntary tunnelling PPTP. The IBM router can either terminate the tunnel or initiate the tunnel. If it terminates the tunnel, the client initiating the tunnel must be PPTP capable and could use an NT or Windows device or any other device supporting PPTP. In the second scenario, a router could establish a tunnel back to a PPTP device, such as an NT server.

Compulsory Tunneling with L2TP

Compulsory tunneling is a router-initiated model. In this scenario, the client has no L2TP knowledge. The client dials into the LAC and the LAC initiates the L2TP tunnel back to the LNS. In this case, the LAC sends an incoming-call-request to the LNS. Since the client and the LAC have already negotiated LCP and partial authentication, the LAC passed this information to the LNS in what is called proxy-authentication. After call establishment, the LNS completes PPP authentication and network phase with the client through the newly formed tunnel.

VPN Event Logging Support (ELS)

The following four subsystems will help you debug and determine the status of your VPN configuration.

L2 Subsystem

The L2 Subsystem contains the ELS Messages for all of the layer 2 tunneling protocols including L2F, L2TP, and PPTP. This subsystem shows information about tunnel and call establishment and termination. It also shows information about packets received and transmitted through the L2 tunnels. Error messages resulting from failed negotiation of the tunnel are also displayed.

PLCY Subsystem

The ELS messages for the PLCY Subsystem tell about the status of the policy database refresh, how many rules where loaded into the database and information about errors that may have occurred when the policy database was being built. You may look at the packet information for policy database queries, what rules and actions resulted from these queries and any errors or other information pertinent to negotiations involving the policy database.

IPSP Subsystem

The IPSP Subsystem contains messages for the IPSec module in the router. The IPSP subsystem shows information about packet encryption and decryption, the algorithms that are being used and error messages resulting from packets being discarded because of failures.

IKE Subsystem

The IKE Subsystem shows information about the phase I and phase II negotiations which ultimately setup a secure IPSec Tunnel between two security gateways or hosts. Any errors that result from failed negotiations due to pre-shared key mismatch, security proposal mismatches or policy errors will be displayed.

Chapter 20. Virtual Private Networks Examples

In this chapter, you will find the following examples which demonstrate basic VPN solutions:

- IPSec Router to Router VPN Using Pre-Shared Keys
- Router to Router VPN Using Digital Certificates
- Voluntary PPTP Tunnel with IBM Router Termination
- IBM Network Utility Initiated Voluntary PPTP Tunnel
- IBM Network Utility Initiated Voluntary L2TP Tunnel
- L2TP Tunnel Terminated at an IBM Network Utility LNS

IPSec Router to Router VPN Using Pre-Shared Keys

This example uses IPSec with automatic key generation and pre-shared keys. In Figure 64, a secure tunnel is created from gateway to gateway. This tunnel will authenticate and encrypt traffic from specific hosts and will exclude all other traffic. The profile describes exactly which hosts at either end of the tunnel are allowed to pass data through the tunnel. The policy can allow a single host on either end, single or multiple subnets on either end, or any combination of the two. The limitation of gateway to gateway tunneling is that no authentication or encryption occurs on the LAN. This solution does not provide security on the LAN.

The network used for this example (Figure 64) consists of two token-ring segments connected by two IBM 2210 routers. In a real scenario, the serial link between the routers could be any private or public wide-area network (WAN).

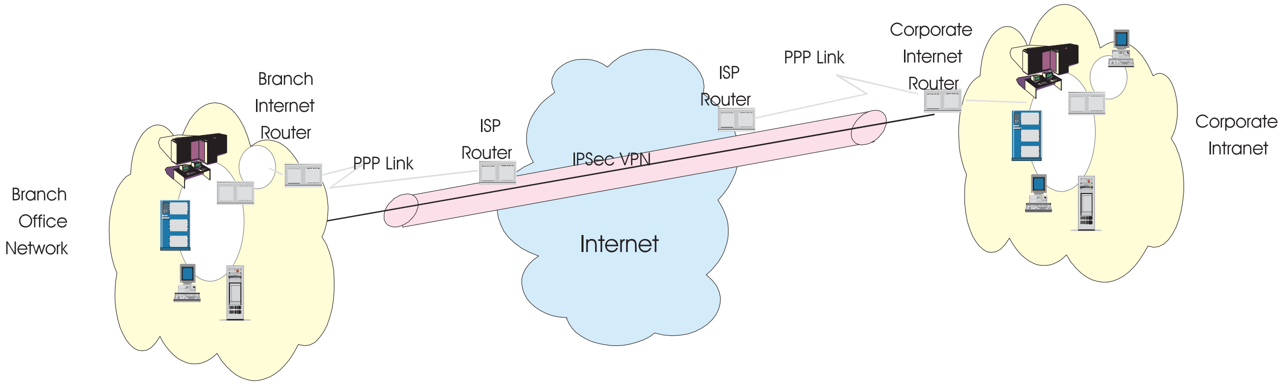


Figure 64. Physical Network Used for Example Configurations

In Figure 65 on page 292, the tunnel must authenticate and encrypt all traffic between Subnet 9.24.106.0 (via branch router VPNRTR2 and corporate router VPNRTR1) and Subnet 192.168.141.32. No other traffic from any other subnet may traverse the link between the two routers. Authentication guarantees that a tunnel is set up between the correct endpoints, and encryption protects the data from being interpreted as it crosses the WAN.

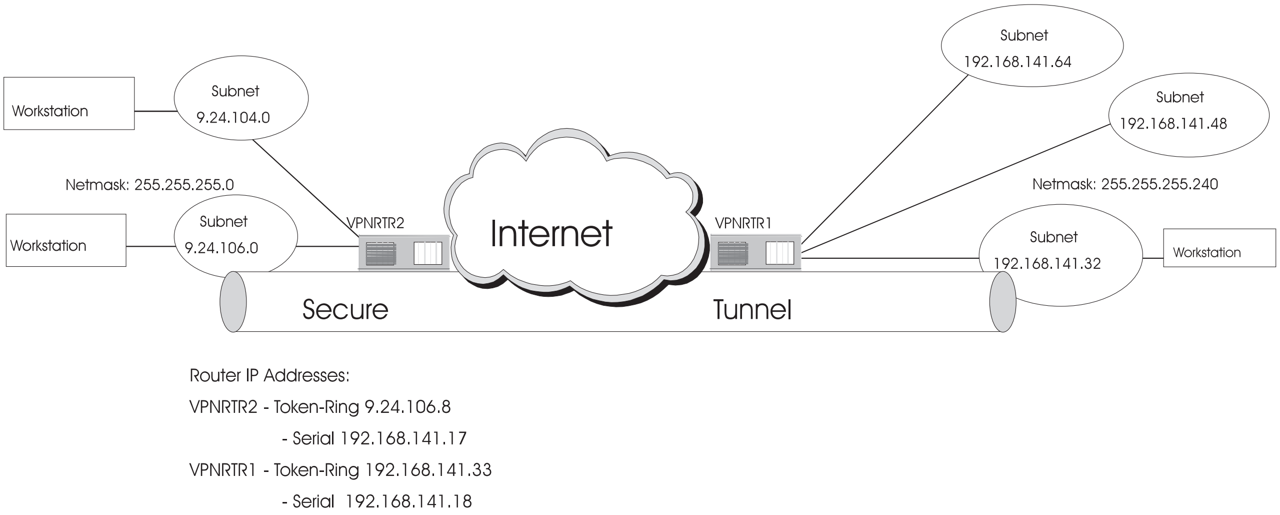


Figure 65. Sample Network Used for IPSec With Pre-Shared Keys

Create a Policy for the IPSec Tunnel for VPNRTR1

Follow these steps to configure the router:

1. Enable IP Security
2. Create the pre-shared key
3. Add a policy
4. Add a profile
5. Add a validity period
6. Add an IPSec action
7. Add an IPSec proposal
8. Add an ESP transform
9. Add an ISAKMP action
10. Add an ISAKMP proposal

Enable IP Security

From the Talk 6 command line interface, enable IP Security at the box level.

Table 78. Enable IP Security

```
VPNRTR1  *TALK 6
Gateway user configuration
VPNRTR1 Config>feature ipsec
IP Security feature user configuration
IPsec config>ipv4
VPNRTR1 IPV4-IPsec config>ENABLE IPSEC
It is necessary to restart the router for IPsec to be active.
VPNRTR1 IPV4-IPsec config>EXIT
VPNRTR1 IPsec config>EXIT
```

Create the Pre-Shared Key

A pre-shared key must be configured for every remote user. Because of this, pre-shared keys are not very scalable. However, pre-shared keys are refreshed on a regular basis, giving them an advantage over the manual tunnel method.

The Talk 6 **Add User** command is used to configure the keys.

Table 79. Add the PPP User

```
VPNRTR1 Config>FEATURE Policy
IP Network Policy configuration VPNRTR1 Policy config>ADD USER
Choose from the following ways to identify a user:      1
        1: IP Address
        2: Fully Qualified Domain Name
        3: User Fully Qualified Domain Name
        4: Key ID (Any string)
Enter your choice(1-4) [1]? 1
Enter the IP Address that distinguishes this user
 [0.0.0.0]? 192.168.141.17                           2
Group to include this user in []?
Authenticate user with 1:pre-shared key or 2: Public Certificate [1]? 1
Mode to enter key (1=ASCII, 2=HEX) [1]? 1
Enter the Pre-Shared Key (an even number of 2-128 ascii chars):
Enter the Pre-Shared Key again (3 characters) in ascii:       3



Here is the User Information you specified...

Name      = 192.168.141.17
Type      = IPV4 Addr
        Group     =
        Auth Mode =Pre-Shared Key
        Key(Ascii)=key                                   3
Is this correct? [Yes]:
```

1. The router needs to know how to recognize the remote IKE peer and the pre-shared key.
2. In this example, the chosen identifier was the IP Address. This must be the address of the tunnel endpoint of the remote router—in this example, the IP address of the WAN interface of VPNRTR2.
3. The key must be entered twice for validation and must be exactly the same for each router at the tunnel endpoints. In this example, the word **key** is used. You can use any key with up to 128 characters. However, you must enter exactly the same key twice in each router. The easiest way to do this is to type the key in a text editor and then cut and paste the key into the entry field of the Talk 6 prompts.

Add the Policy

A policy is the framework for describing how traffic entering or leaving the router is to be handled. Without access control, the router only makes routing decisions. Using the policy, the router makes decisions such as whether to pass the packet across the interface, whether the packet needs to be authenticated and whether the packet needs to be encrypted or decrypted. The policy ties other objects together. This example uses the **Add Policy** command first. This is probably the easiest way to create the first policy, since it prompts you to enter all the necessary information in the correct order.

If a policy creates a security tunnel, only the packets matching the profile will be encrypted and forwarded. Other packets not matching the profile are passed in the clear unless explicitly dropped by another policy. When more than one policy exists on a router, the policies are evaluated according to priority number.

Refer to Figure 66 on page 294 to see how an incoming packet is evaluated against multiple policies. If the packet matches the profile for Policy #1, it is sent to IPSec for processing. If the packet does not match the profile for Policy #1, it is evaluated against the profile for Policy #2. This process continues until the incoming packet is evaluated against all policies. If the packet did not match any profile, it would be sent in clear text to the routing protocol for processing.

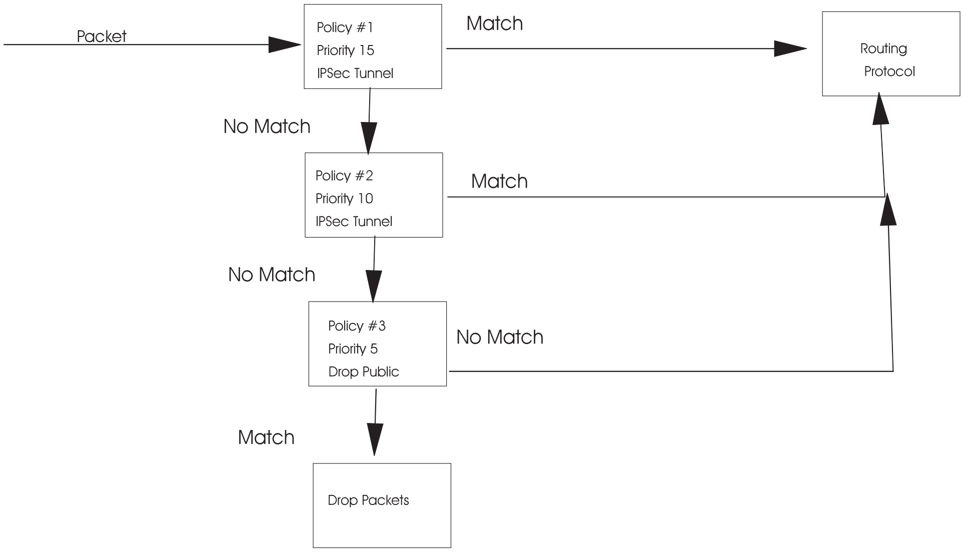


Figure 66. Effect of Multiple Policies

Table 80. Create a New Policy

```
VPNRTR1 Policy config>ADD POLICY
Enter a Name (1-29 characters) for this Policy []? ike-pre-32-106
Enter the priority of this policy (This number is used to determine
the policy to enforce in the event of policy conflicts) [5]? 15           1
List of Profiles:
        0: New Profile                 2
```

1. The priority of the policy specifies the order in which multiple policies will be evaluated. The higher numbered policies are evaluated before lower numbered policies. See Figure 66 for the flow of packet evaluation when multiple policies are defined.

2. After a policy is added, you must create a profile to associate with the policy. Since no profile has been created on this router, the prompt is given to create a new profile.

Add the Profile

The profile describes the criteria used to determine if a packet should be acted on by the policy. These criteria include source and destination address, protocol, port type, and Differentiated Services (DS/TOS) byte.

Table 81. Add the Profile

```
List of Profiles:
        0: New Profile
Enter a Name (1-29 characters) for this Profile []? 32-106       1
Source Address Format (1:NetMask, 2:Range, 3:Single Addr) [1]?       2
Enter IPV4 Source Address [192.168.141.32]?
Enter IPV4 Source Mask [255.255.255.240]?
Destination Address Format (1:NetMask, 2:Range, 3:Single Addr) [1]?
Enter IPV4 Destination Address [9.24.106.0]?
Enter IPV4 Destination Mask [255.255.255.0]?

Protocol IDs:
     1)  TCP
     2)  UDP
     3)  All Protocols
     4)  Specify Range

Select the protocol to filter on (1-4) [3]?        3
Enter the Starting value for the Source Port [0]?
Enter the Ending value for the Source Port [65535]?
Enter the Starting value for the Destination Port [0]?
Enter the Ending value for the Destination Port [65535]?
Enter the Mask to be applied to the Received DS-byte [0]?       4
Enter the value to match against after the Mask has
been applied to the Received DS-byte [0]?
Configure local and remote ID's for ISAKMP? [No]: YES        5
Enter local identification to send to remote
     1)  Local Tunnel Endpoint Address
     2)  Fully Qualified Domain Name
     3)  User Fully Qualified Domain Name
     4)  Key ID (any string)
Select the Identification type (1-4) [1]?
Any user within profile definition allowed access? [Yes]:
Limit this profile to specific interface(s)? [No]:


...continued on next screen
```

1. The descriptive name 106-32 is used here.
2. In this example, any host on one subnet should be able to access any host on the other subnet.
3. You may configure the profile to allow only certain protocols and certain ports if you want to limit services further. For instance, you can allow Telnet only and not FTP by allowing only TCP port 23.
4. The DS byte is related to QoS or prioritization. You can select which traffic matches the profile by priority level.
5. This step of configuring local and remote IDs for ISAKMP is optional unless your peer must identify you with something other than your IP address.

Table 82. Confirm the Profile

```
Here is the Profile you specified...


Profile Name    = 32->106
        sAddr:Mask= 192.168.141.32: 255.255.255.240 sPort=    0 : 65535
        dAddr:Mask= 9.24.106.0 : 255.255.255.0   dPort=    0 : 65535
        proto     =             0 : 255
        TOS       =           x00 : x00
        Remote Grp=All Users
Is this correct? [Yes]:
List of Profiles:
        0: New Profile
        1: 32->106

Enter number of the profile for this policy [1]? 1
```

Add the Validity Period

The validity period is the time period during which the policy is valid. You may configure the validity period to specify a time duration or the months of the year, days of the week and hours of the day that the policy is valid. This flexibility enables the network administrator to specify when a policy is valid. For example, requirements could be ″all the time″ or ″only this year during January and February″ or ″only Monday through Friday 9AM to 5PM.″ These requirements can be translated to configuration values using the **Add Validity Period** command.

Table 83. Add the Validity Period

```
List of Validity Periods:              1
        0: New Validity Period

Enter number of the validity period for this policy [0]?
Enter a Name (1-29 characters) for this Policy Valid Profile []? always
Enter the lifetime of this policy. Please input the
information in the following format:
                yyyymmddhhmmss:yyyymmddhhmmss OR '*' denotes forever.
 [*]? *
During which months should policies containing this profile
be valid.  Please input any sequence of months by typing in
the first three letters of each month with a space in between
each entry, or type ALL to signify year round.
 [ALL]?
During which days should policies containing this profile
be valid.  Please input any sequence of days by typing in
the first three letters of each day with a space in between
each entry, or type ALL to signify all week
 [ALL]?
Enter the starting time (hh:mm:ss or * denotes all day)
 [*]?


Here is the Policy Validity Profile you specified...


Validity Name   = always                      2
        Duration  = Forever
        Months    = ALL
        Days      = ALL
        Hours     = All Day
Is this correct? [Yes]:
List of validity periods:
      0: New Validity Period
      1: always
Enter number of the validity period for this policy [1]?
```

1. Since you are creating a new policy, you are prompted to create a validity period. In some instances, you can reuse a validity period that has been previously created. You will see in later examples in this chapter that an existing validity period is used. This concept is true of all policy database objects. When appropriate, any object can be reused.
2. The validity period in our example has been configured to be in effect at all times.

Add the IPSec Action

In addition to a profile and a validity period, a policy must also be associated with either an IPSec action, a Manual IPsec or a DiffServ Action. In this scenario, an IPSec action is configured.

An IPSec action may specify either a drop, pass or secure action. If the action is drop, then all packets matching the profile used by this policy are dropped. If the action is pass with no security, then all packets are passed in the clear. If the action is pass with security, then all packets are secured by means of the SA specified by this action. The IPSec action also contains the IP addresses of the tunnel endpoints for the IPSec tunnel and IKE SAs.

Table 84. Add the IPSec Action

```
Should this policy enforce an IPSEC action? [No]: y
IPSEC Actions:
        0: New IPSEC Action

Enter the Number of the IPSEC Action [0]?
Enter a Name (1-29 characters) for this IPsec Action []? tunnel_vpnrtr1-vpnrtr2
List of IPsec Security Action types:
     1)  Block (block connection)
     2)  Permit

Select the Security Action type (1-2) [2]?
Should the traffic flow into a secure tunnel or in the clear:
     1) Clear
     2) Secure Tunnel
 [2]?
Enter Tunnel Start Point IPV4 Address
 [192.168.141.18]?
Enter Tunnel End Point IPV4 Address (0.0.0.0 for Remote Access)
 [0.0.0.0]? 192.168.141.17
Does this IPSEC tunnel flow within another IPSEC tunnel? [No]:        1
Percentage of SA lifesize/lifetime to use as the acceptable minimum [75]?
Security Association Refresh Threshold, in percent (1-100) [85]?
Options for DF Bit in outer header (tunnel mode):          2
     1)  Copy
     2)  Set
     3)  Clear
Enter choice (1-3) [1]?
Enable Replay prevention (1=enable, 2=disable) [2]?          3
Do you want to negotiate the security association at
system initialization(Y-N)? [No]: y          4
```

1. You are then asked if the negotiated IPSec tunnel flows into another tunnel. This relates to the tunnel-in-tunnel feature which was first shipped in V3.2 of the code. A scenario for tunnel-in-tunnel is shown in Figure 67:

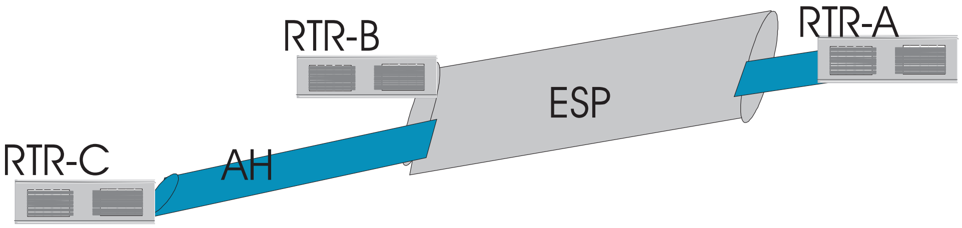


Figure 67. Tunnel in a Tunnel

All traffic going between RTR-A and RTR-C should be authenticated, but all traffic between RTR-A and RTR-B must be encrypted. The distinction of tunnel-in-tunnel is that tunnels start at the same point but finish at different points. For this example, the answer is no.

2. When the IPSec header is created, many of the fields of the IP header are copied from the header of the packet being secured. You can control how the **don’t fragment** field is set. You can copy from the original packet, set the DF bit or, if it is turned on in the original packet, turn it off. Having the DF bit set has implications for IPSec. Consider the diagram below.

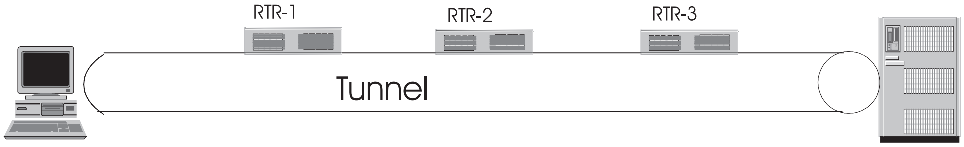


Figure 68. Understanding the DF Bit

The traffic is flowing from the device on the left to the device on the right. RTR-2 needs to fragment the packet but cannot do so because the DF bit is set. RTR-2 will generate an ICMP packet too big message and send it to the sender of the packet, RTR-1. RTR-1 then needs to inform the sender that the packet is too large. This can cause problems for RTR-1 because either (a) the ICMP packet might not contain enough of the original packet to be able to determine who the sender is or (b) the IP address might be encrypted. If RTR-1 cannot determine who the sender is, he will store tunnel information and wait for another packet to arrive for that tunnel. When that packet arrives, RTR-1 will then generate the ICMP packet too big message if necessary. So consider carefully the setting of the DF bit.

In this example, we have set the DF bit to copy (the default).

3. **Enable replay prevention** defines whether the sequence numbers should be checked on received packets.
4. This parameter controls whether this SA should be created at system start-up. Specifying **no** indicates that this SA should only be negotiated when packets are received which match the policy.

Following this step we are prompted to select an IPSec proposal. Since no previous proposal exists, the only option is to create a new one.

Add an IPSec Proposal

The IPSec proposal contains the information about which ESP, AH, (or both) transform to propose or check against during phase 2 ISAKMP negotiations. Refer to “IKE” on page 284 for an explanation of phase 2 negotiations. If you require pfs (perfect forward secrecy), then the IPSec proposal identifies which DH (Diffie-Hellman) group to use. The transforms that the IPSec proposal references are sent or checked against the order in which they are specified. The first ESP or AH transform in the list must be the one that is most appropriate to use. If more than one transform is in the list, then each one is compared to the peer’s list of transforms to find a match. If none of the configured transforms match the peer’s list, then the negotiation fails. The IPSec proposal may list a combination of AH and ESP transforms, but the only valid combinations are:

- List of AH only (tunnel or transport mode)
- List of ESP only (tunnel or transport mode)
- List of AH (transport mode) and list of ESP (tunnel mode)
  - AH (transport mode) + ESP (transport mode) defines transport mode
  - AH (tunnel or transport mode) + ESP (tunnel or transport mode) defines tunnel mode

Table 85. Adding the IPSec Proposal

```
You must choose the proposals to be sent/checked against during phase 2
negotiations.  Proposals should be entered in order of priority.

List of IPSEC Proposals:         1
        0: New Proposal

Enter the Number of the IPSEC Proposal [0]?
Enter a Name (1-29 characters) for this IPsec Proposal []? esp-prop1
Does this proposal require Perfect Forward Secrecy?(Y-N)? [No]:
Do you wish to enter any AH transforms for this proposal? [No]:
Do you wish to enter any ESP transforms for this proposal? [No]: y         2
```

1. Because you have specified an IPSec action, you are prompted to create a new proposal.
2. Type **y** to enter an ESP transform and you will be prompted to add the transform.

Add an IPSec Transform

The attributes of the IPSec transform contain information about the IPSec encryption and authentication parameters and also specify how often the keys are refreshed. The transform is either AH (authentication only) or ESP (encryption, authentication, or both) and may be configured to operate in either tunnel or transport mode.

Table 86. Add the IPSec Transform

```
List of ESP Transforms:
        0: New Transform

Enter the Number of the ESP transform [0]?
Enter a Name (1-29 characters) for this IPsec Transform []? esp-trans1
List of Protocol IDs:
     1)  IPSEC AH
     2)  IPSEC ESP

Select the Protocol ID (1-2) [1]? 2
List of Encapsulation Modes:
     1)  Tunnel                              1
     2)  Transport

Select the Encapsulation Mode(1-2) [1]?
List of IPsec Authentication Algorithms:          2
0)  None
     1)  HMAC-MD5
     2)  HMAC_SHA

Select the ESP Authentication Algorithm (0-2) [2]?
List of ESP Cipher Algorithms:
     1)  ESP DES
     2)  ESP 3DES
     3)  ESP CDMF
     4)  ESP NULL

Select the ESP Cipher Algorithm (1-4) [1]?           3
Security Association Lifesize, in kilobytes (1024-65535) [50000]?           4
Security Association Lifetime, in seconds (120-65535) [3600]?


Here is the IPSec transform you specified...


Transform Name  = esp-trans1
        Type =ESP   Mode =Tunnel     LifeSize=   50000 LifeTime=    3600
        Auth =SHA   Encr =DES
Is this correct? [Yes]: y
List of ESP Transforms:
        0: New Transform
        1: esp-trans1

Enter the Number of the ESP transform [1]?
Do you wish to add another ESP transform to this proposal? [Yes]: n
```

1. Transport mode is an SA defined between two end stations. Tunnel mode is used when at least one of the devices is a security gateway (for example, a router). We have selected tunnel mode since our SA is between two routers.
2. These are the authentication methods. HMAC_SHA is more secure than HMAC_MD5.
3. Select the encryption method. Note that ESP 3DES is not allowed outside the United States.
4. Set the SA lifetime/lifesize. When the SA expires, IKE will perform another phase II calculation to refresh the keys. The default of 3600 seconds has been set. This means that someone who could intercept one of your packets would have only 1 hour to break the code. In one case, a group of university students demonstrated that single DES encryption can be broken in 22 hours.

After adding the IPSec transform, you are prompted to confirm the IPSec Proposal.

Table 87. Confirm the IPSec Proposal

```
Here is the IPSec proposal you specified...


Name  = esp-prop1
        Pfs   = N
        ESP Transforms:
                esp-trans1
Is this correct? [Yes]: y
List of IPSEC Proposals:
        0: New Proposal
        1: esp-prop1

Enter the Number of the IPSEC Proposal [1]?
Are there any more Proposal definitions for this IPSEC Action? [No]:
```

After the IPSec transform and proposal are created, we can finish the IPSec Action. We are given a confirmation screen and prompted to select the action to be associated with the policy.

Table 88. Confirm the IPSec Action

```
Here is the IPSEC Action you specified...

IPSECAction Name = tunnel_vpnrtr1-vpnrtr2
        Tunnel Start:End         = 192.168.141.18 : 192.168.141.17
        Tunnel In Tunnel         =            No
        Min Percent of SA Life   =            75
        Refresh Threshold        =            85 %
        Autostart                =           Yes
        DF Bit                   =          COPY
        Replay Prevention        =      Disabled
        IPSEC Proposals:
                esp-prop1
Is this correct? [Yes]:
IPSEC Actions:
        0: New IPSEC Action
        1: tunnel_vpnrtr1-vpnrtr2

Enter the Number of the IPSEC Action [1]?
```

Add ISAKMP Action

Since a secure IPSec action was specified, you are automatically prompted to create an ISAKMP action. In most cases, one ISAKMP action and one ISAKMP proposal is sufficient for all security policies. The algorithms and methods that you select will most likely be strategic, enterprise-wide parameters. For example, your company will make a decision based on corporate security requirements such as choosing between maximizing encryption levels or striking a balance between privacy and performance. As with any security design, your original configuration should be audited and monitored to insure that it is properly supporting your intentions. The ISAKMP action specifies the key management information for phase I. Refer to “IKE” on page 284 for an explanation of the phase I and phase II negotiations.

Table 89. Add the ISAKMP Action

```
ISAKMP Actions:
        0: New ISAKMP Action

Enter the Number of the ISAKMP Action [0]?
Enter a Name (1-29 characters) for this ISAKMP Action []? ike-1

List of ISAKMP Exchange Modes:           1
     1)  Main
     2)  Aggressive

Enter Exchange Mode (1-2) [1]?
Percentage of SA lifesize/lifetime to use as the acceptable minimum [75]?
ISAKMP Connection Lifesize, in kilobytes (100-65535) [5000]?       2
ISAKMP Connection Lifetime, in seconds (120-65535) [30000]?
Do you want to negotiate the security association at
system initialization(Y-N)? [Yes]:          3
```

1. The exchange modes relate to the level of security during phase 1 negotiations. Aggressive mode is quicker since it has a fewer number of messages exchanged, but it is less secure because the identity of the initiator is sent in the clear. The action is selected to occur in main mode.
2. Connection Lifesize and Connection Lifetime control when a new SA will be negotiated. This refreshing of keys can take several seconds, so the smaller the numbers, the more often a refresh occurs. As with many choices concerning security, a good balance is required between performance and security requirements.
3. We selected to negotiate the SA at initialization in order to improve the performance of the initial transactions that occur across the tunnel.

Add ISAKMP Proposal

The ISAKMP proposal specifies the encryption and authentication attributes of the phase I SA. It also specifies which DH group to use to generate the keys and the life of the phase I security.

Table 90. Add the ISAKMP Proposal (Screen 1 of 2)

```
You must choose the proposals to be sent/checked against during phase 1
negotiations.  Proposals should be entered in order of priority.
List of ISAKMP Proposals:
        0: New Proposal

Enter the Number of the ISAKMP Proposal [0]? 0
Enter a Name (1-29 characters) for this ISAKMP Proposal []? ike-prop1

List of Authentication Methods:
     1)  Pre-Shared Key
     2)  RSA SIG

Select the authentication method (1-2) [1]? 1

List of Hashing Algorithms:
     1)  MD5
     2)  SHA

Select the hashing algorithm(1-2) [1]? 1

List of Cipher Algorithms:
     1)  DES
     2)  3DES

Select the Cipher Algorithm (1-2) [1]? 1

...continued
```

Table 91. Add the ISAKMP Proposal (Screen 2 of 2)

```
Security Association Lifesize, in kilobytes (100-65535) [1000]?
Security Association Lifetime, in seconds (120-65535) [15000]?

List of Diffie Hellman Groups:
     1)  Diffie Hellman Group 1
     2)  Diffie Hellman Group 2

Select the Diffie Hellman Group ID from this proposal (1-2) [1]?       1


Here is the ISAKMP Proposal you specified...

Name = ike-prop1
        AuthMethod = Pre-Shared Key
        LifeSize   = 1000
        LifeTime   = 15000
        DHGroupID  = 1
        Hash Algo  = MD5
        Encr Algo  = DES CBC
Is this correct? [Yes]:
List of ISAKMP Proposals:
        0: New Proposal
        1: ike-prop1

Enter the Number of the ISAKMP Proposal [1]?
Are there any more Proposal definitions for this ISAKMP Action? [No]:
```

1. Select 1 for pre-shared keys. If we wanted to use certificates for authentication, we would have selected RSA-SIG.

Once all of the objects necessary for a secure tunnel policy have been created, a summary of the policy is presented. The defined policy—ike-pre-32-106—has a priority of 15 and will set up a secure tunnel between the 192.168.141.32 subnet and the 9.24.106.0 subnet. The IPSec action specifies a secure tunnel which will always be in effect as specified by the valid period. The packets allowed to enter the tunnel are determined by the profile which describes the two subnets. The authentication and encryption methods are specified in the ISAKMP action and ISAKMP proposal.

Table 92. Confirm the ISAKMP Action

```
Here is the ISAKMP Action you specified...


ISAKMP Name     = ike-1
        Mode                    =            Main
        Min Percent of SA Life  =              75
        Conn LifeSize:LifeTime  =            5000 : 30000
        Autostart               =             Yes
        ISAKMP Proposals:
                ike-prop1
Is this correct? [Yes]: y
ISAKMP Actions:
        0: New ISAKMP Action
        1: ike-1

Enter the Number of the ISAKMP Action [1]?
```

Confirm the Policy

After confirming the ISAKMP Action and Proposal, you may wish to configure a DiffServ Action. In that case, a summary of the policy is presented for confirmation.

Table 93. Confirm the Policy

```
Do you wish to Map a DiffServ Action to this Policy? [No]:
Policy Enabled/Disabled (1. Enabled, 2. Disabled) [1]?

Here is the Policy you specified...


Policy Name     = ike-pre-32-106
        State:Priority =Enabled    : 15
        Profile        =32-106
        Valid Period   =always
        IPSEC Action   =tunnel_vpnrtr1-vpnrtr2
        ISAKMP Action  =ike-1
Is this correct? [Yes]:   Y
```

This policy will evaluate all packets entering the router and forward those packets matching the profile to IPSec for encryption. If no further policies are created, then all packets not matching the profile will be routed in the clear to the appropriate interface.

However, for the purpose of this scenario, only traffic between the two subnets should cross the VPN. To accomplish this, a policy to drop all traffic that does not come from one of the two specified subnets will have to be created.

Create a Policy on VPNRTR1 to Drop Public Traffic

These are the steps to create the policy to drop public traffic that is not from either of the subnets specified in the IPSec tunnel policy. The steps to configure this policy are similar to the tunnel policy except there are fewer steps:

1. Add the policy
2. Add the profile

3. Specify the interfaces
4. Add the validity period
5. Add the IP security action
6. Confirm the policy

Add the Policy

Table 94. Add Policy to Drop Public Traffic

```
VPNRTR1 Config>FEATURE Policy
IP Network Policy configuration
VPNRTR1 Policy config>ADD POLICY
Enter a Name (1-29 characters) for this Policy []? dropAllPublicTraffic
Enter the priority of this policy (This number is used to
determine the policy to enforce in the event of policy conflicts) [5]?         1
```

1. In the policy above, a priority of 15 was assigned. The next time, a priority of 5 will be assigned. Therefore, incoming packets will be evaluated against the profile of the tunnel policy first. If they do not match that profile, they will be evaluated against this profile. The result will then be that all packets not meeting the tunnel profile will match this profile and therefore be dropped.

Add the Profile

This profile is designed to match all traffic.

Table 95. Add Policy to Match All Traffic

```
List of Profiles:
        0: New Profile
        1: 32->106

Enter number of the profile for this policy [1]? 0
Enter a Name (1-29 characters) for this Profile []? allPublicTraffic
Source Address Format (1:NetMask, 2:Range, 3:Single Addr) [1]?
Enter IPV4 Source Address [0.0.0.0]?
Enter IPV4 Source Mask [0.0.0.0]?
Destination Address Format (1:NetMask, 2:Range, 3:Single Addr) [1]?
Enter IPV4 Destination Address [0.0.0.0]?
Enter IPV4 Destination Mask [0.0.0.0]?

Protocol IDs:
     1)  TCP
     2)  UDP
     3)  All Protocols
     4)  Specify Range

Select the protocol to filter on (1-4) [3]?
Enter the Starting value for the Source Port [0]?
Enter the Ending value for the Source Port [65535]?
Enter the Starting value for the Destination Port [0]?
Enter the Ending value for the Destination Port [65535]?
Enter the Mask to be applied to the Received DS-byte [0]?
Enter the value to match against after the Mask has
been applied to the Received DS-byte [0]?
Configure local and remote ID's for ISAKMP? [No]:
```

Specify the Interface Pairs

Since no source and destination IP Addresses were specified, which interfaces the policy applies to must be specified.

Table 96. Define Interfaces to Block Public Traffic

```
The Source and/or Destination Address information you specified
includes all addresses.  You must specify an Interface Pair
with this profile to further qualify what traffic you wish to filter
to this policy.
Limit this profile to specific interface(s)? [No]: yes
Interface Pair Groups:
        0: New Ifc Pair
Number of Ifc Pair Group [1]? 0
Enter a Group Name (1-29 characters) for this Interface Pair []? inOutPublic
Ingress Interface IP Address (255.255.255.255 = any ingress)
 [255.255.255.255]?
Egress Interface IP Address (255.255.255.255 = any egress)
 [255.255.255.255]? 192.168.141.18
Interface Pair Groups:
        0: New Ifc Pair
        1) Group Name: inOutPublic
                  In:Out=255.255.255.255 : 192.168.141.18
```

Table 97. Verify Specified Interfaces

```
Number of Ifc Pair Group [1]? 0
Enter a Group Name (1-29 characters)for this Interface Pair []?inOutPublic
Ingress Interface IP Address (255.255.255.255 = any ingress)
 [255.255.255.255]? 192.168.141.18
Egress Interface IP Address (255.255.255.255 = any egress)
 [255.255.255.255]?
Interface Pair Groups:
        0: New Ifc Pair
        1) Group Name: inOutPublic
                  In:Out=255.255.255.255 : 192.168.141.18
                  In:Out= 192.168.141.18 : 255.255.255.255

Number of Ifc Pair Group [1]? 1


Here is the Profile you specified...


Profile Name    = allPublicTraffic
        sAddr:Mask=       0.0.0.0 : 0.0.0.0   sPort=    0 : 65535       1
        dAddr:Mask=       0.0.0.0 : 0.0.0.0   dPort=    0 : 65535       2
        proto     =             0 : 255
        TOS       =           x00 : x00             3
        Remote Grp=All Users
        1.  In:Out=255.255.255.255 : 192.168.141.18
        2.  In:Out= 192.168.141.18 : 255.255.255.255
Is this correct? [Yes]:
```

1. All zeros indicates that traffic from any source matches the profile.
2. All zeros indicates that traffic with any destination matches the profile.
3. Leaving the default TOS at x00 indicates traffic at any priority level will match the profile.

Add the Validity Period

Once the interfaces are specified, the profile and the validity period must be selected. A new validity period does not need to be created since the previous configured **always** description can be used.

Table 98. Reuse the ALWAYS Validity Period

```
List of Profiles:
        0: New Profile
        1: 32->106
        2: allPublicTraffic

Enter number of the profile for this policy [1]? 2
List of Validity Periods:
        0: New Validity Period
        1: always

Enter number of the validity period for this policy [1]? 1
```

Add the IPSec Action

Describe a security action to drop all traffic that matches the **allPublicTraffic** profile. The fact that this policy is set at a lower priority than the tunnel policy will cause the correct traffic to enter the tunnel and all other traffic to be dropped. In other words, each incoming packet is tested against the tunnel profile first and against the public profile last.

Table 99. Add the IPSec Action to Drop Public Traffic

```
Should this policy enforce an IPSEC action? [No]: yes
IPSEC Actions:
        0: New IPSEC Action
        1: tun-32->106

Enter the Number of the IPSEC Action [1]? 0
Enter a Name (1-29 characters) for this IPsec Action []? dropTraffic
List of IPsec Security Action types:
     1)  Block (block connection)
     2)  Permit

Select the Security Action type (1-2) [2]? 1


Here is the IPSec Action you specified...


IPSECAction Name = dropTraffic
        Action   = Drop
Is this correct? [Yes]: yes
IPSEC Actions:
        0: New IPSEC Action
        1: tun-32->106
        2: dropTraffic

Enter the Number of the IPSEC Action [1]? 2
Do you wish to Map a DiffServ Action to this Policy? [No]:
Policy Enabled/Disabled (1. Enabled, 2. Disabled) [1]? 1
```

Confirm the Policy is Correct

Confirm the policy by entering **Yes**.

Table 100. Confirm the Policy to Drop Traffic

```
Here is the Policy you specified...


Policy Name     = dropAllPublicTraffic
        State:Priority =Enabled    : 1
        Profile        =allPublicTraffic
        Valid Period   =always
        IPSEC Action   =dropTraffic
Is this correct? [Yes]:   Y
```

This completes the configuration on VPNRTR1. Be sure to make a copy of the configuration in IBD, the configuration program, or send to a TFTP server.

Create a Policy for the IPSec Tunnel for VPNRTR2

Refer to Figure 65 on page 292 for the network diagram and IP addresses for this example. These are the steps to configure the router:

1. Enable IP security
2. Create the pre-shared key
3. Add a policy
4. Add a profile
5. Add a validity period
6. Add an IPSec action
7. Add an IPSec proposal
8. Add an ESP transform

The steps for creating the VPNRTR2 policy are the same as those for VPNRTR1 with the following differences:

- The profile for VPNRTR2 tunnel policy reverses the sAddr:Mask and the dAddr:Mask used with VPNRTR1.
- The profile for VPNRTR2 drop policy reverses the specified interfaces.
- The IPSec Action for VPNRTR2 reverses the Tunnel Start:End points.
- The user defined for VPNRTR2 is the tunnel endpoint of VPNRTR1.

Note: Make sure that the pre-shared key is identical in both routers. An easy way to do this is to cut and paste the keys. Also, be aware that if the key entered is longer than the character width of the Telnet session screen, you may not see the entire key when the confirmation screen is presented.

Table 101 on page 310 shows the output of the Talk 6 **list all** command after completing the configuration of VPNRTR2. The values that are different from the VPNRTR1 policy are annotated below each figure.

Table 101. List All of the Policy Database Objects for VPNRTR2

```
VPNRTR2 Policy config>LIST ALL

Configured Policies....

Policy Name     = ike-pre-106->32             1
        State:Priority =Enabled    : 15
        Profile        =106->32               2
        Valid Period   =always
        IPSEC Action   =ike-1
        ISAKMP Action  =ike-1

Policy Name     = dropAllPublicTraffic
        State:Priority =Enabled    : 5
        Profile        =allPublicTraffic
        Valid Period   =always
        IPSEC Action   =dropTraffic

Configured Profiles....

Profile Name    = 106->32      3
        sAddr:Mask=     9.24.106.0 : 255.255.255.0   sPort=     0 : 65535
        dAddr:Mask= 192.168.141.32 : 255.255.255.240 dPort=    0 : 65535
        proto     =              0 : 255
        TOS       =            x00 : x00
```

1. The name of the policy is only for reference. Please use a meaningful name.
2. Use a meaningful name for the profile.
3. The profile addresses are the reverse of the router at the opposite tunnel endpoint.

Table 102. List Policy Database Objects for VPNRTR2 (Screen 1 of 4)

```
Remote Grp=All Users

Profile Name    = allPublicTraffic
        sAddr:Mask=        0.0.0.0 : 0.0.0.0         sPort=    0 : 65535
        dAddr:Mask=        0.0.0.0 : 0.0.0.0         dPort=    0 : 65535
        proto     =              0 : 255
        TOS       =            x00 : x00
        Remote Grp=All Users
        1.  In:Out=255.255.255.255 : 192.168.141.17       1
        2.  In:Out= 192.168.141.17 : 255.255.255.255

Configured Validity Periods

Validity Name   = always
        Duration  = Forever
        Months    = ALL
        Days      = ALL
        Hours     = All Day

Configured DiffServ Actions....
No DiffServ Actions configured
```

1. The address of the WAN port.

Table 103. List Policy Database Objects for VPNRTR2 (Screen 2 of 4)

```
Configured IPSEC Actions....

IPSECAction Name = ike-1
        Tunnel Start:End         = 192.168.141.17 : 192.168.141.18           1
        Tunnel In Tunnel         =             No
        Min Percent of SA Life   =             75
        Refresh Threshold        =             85 %
        Autostart                =             No
        DF Bit                   =           COPY
        Replay Prevention        =       Disabled
        IPSEC Proposals:
                esp-prop1

IPSECAction Name = dropTraffic
        Action   = Drop

Configured IPSEC Proposals....

Name  = esp-prop1
        Pfs   = N
        ESP Transforms:
                esp-trans1
```

1. Remember, for the IPSec action of this router, the tunnel start and end points must be exactly reversed from the router at the opposite end point.

Table 104. List Policy Database Objects for VPNRTR2 (Screen 3 of 4)

```
Configured IPSEC Transforms....

Transform Name  = esp-trans1
        Type =ESP   Mode =Tunnel     LifeSize=   50000 LifeTime=    3600
        Auth =SHA   Encr =DES

Configured ISAKMP Actions....

ISAKMP Name     = ike-1
        Mode                     =             Main
        Min Percent of SA Life   =               75
        Conn LifeSize:LifeTime   =             5000 : 30000
        Autostart                =              Yes
        ISAKMP Proposals:
                ike-prop1
```

Table 105. List Policy Database Objects for VPNRTR2 (Screen 4 of 4)

```
Configured ISAKMP Proposals....
Name = ike-prop1
        AuthMethod = Pre-Shared Key
        LifeSize   = 1000
        LifeTime   = 15000
        DHGroupID  = 1
        Hash Algo  = MD5
        Encr Algo  = DES CBC

Configured Policy Users....
Name      = 192.168.141.18        1
Type      = IPV4 Addr
        Group     =
        Auth Mode =Pre-Shared Key
        Key(Ascii)=key
Configured Manual IPSEC Tunnels....

                             IPv4 Tunnels
------------------------------------------------------------------------------

   ID        Name        Local IPv4 Addr  Rem IPv4 Addr    Mode    State
 ------  ---------------  ---------------  ---------------  -----  --------
VPNRTR2 Policy config>
```

1. The configured policy user for VPNRTR2 will be the IP Address of VPNRTR1.

Create a Policy on VPNRTR2 to Drop Public Traffic

These are the steps to create the policy to drop public traffic that is not from either of the subnets specified in the IPSec tunnel policy.

Note: For exact, step-by-step instructions, refer to “Create a Policy on VPNRTR1 to Drop Public Traffic” on page 305. The only difference is in the **Specify the Interfaces** step where the interface address will be the IP Address of the WAN port for VPNRTR2.

1. Add the policy
2. Add the profile
3. Specify the interfaces
4. Add the validity period
5. Add the IP security action
6. Confirm the policy

Monitoring/Troubleshooting the Policies

The policy database takes the policy and generates the rules which are needed for IPSec. The policy has defined that traffic going from x.x.x.x to x.x.x.x need to be secured using tunnel *tunnelname*. In the past, packet filters would also need to be configured to confirm that traffic from x.x.x.x to x.x.x.x had been secured by tunnel *tunnelname*. The policy feature creates this filter for you. If you want to know what policies have been generated, go into the policy feature from talk 5 and enter **list policy generated**. The router will list all the policies that you have defined. Select the appropriate number, and the router will tell you what rules were generated for that policy.

Table 106. List the Policy Generated

```
VPNRTR2 *TALK 6
VPNRTR2 Config>FEATURE Policy
IP Network Policy configuration
VPNRTR2 Policy console>LIST POLICY GENERATED
1: (Enabled,Valid)      dropAllPublicTraffic
2: (Enabled,Valid)      ike-pki-106-32
Number of Policy to display [0]? 2
Rules generated for policy ike-pki-106-32:
Rule 1.  ike-pki-106-32.p1in
Rule 2.  ike-pki-106-32.p1out
Rule 3.  ike-pki-106-32.p2in
Rule 4.  ike-pki-106-32.traffic
Rule 5.  ike-pki-106-32.inBoundTunnel
```

If you want to know what these rules are, there are two commands: one gives you a summary and one gives you the details. **List rule basic** gives you the basic information about a rule—the priority, how it was generated and how it has been used.

Table 107. List Rule Basic

```
VPNRTR2 Policy console>LIST RULE BASIC
1: (Enabled,Valid)      ike-pki-106-32.p2in
2: (Enabled,Valid)      ike-pki-106-32.p1out
3: (Enabled,Valid)      ike-pki-106-32.p1in
4: (Enabled,Valid)      ike-pki-106-32.traffic
5: (Enabled,Valid)      ike-pki-106-32.inBoundTunnel
11: (Enabled,Valid)     dropAllPublicTraffic
Number of Rule to display (0 for All) [0]? 1
Policy Name: ike-pki-106-32.p2in
Loaded from: Local
State:       Enabled and Valid
Priority:    94
Hits:        0
Profile:     106->32.p2in
Validity:    always
IPSEC:       ike-1
```

List rule complete shows you the details of the rule. This rule is used to confirm that traffic from 192.168.141.18 to 192.168.141.17 was secured using the correct tunnel definition.

Table 108. List Rule Complete

```
VPNRTR2 Policy console>LIST RULE COMPLETE
1: (Enabled,Valid)      ike-pki-106-32.p2in
2: (Enabled,Valid)      ike-pki-106-32.p1out
3: (Enabled,Valid)      ike-pki-106-32.p1in
4: (Enabled,Valid)      ike-pki-106-32.traffic
5: (Enabled,Valid)      ike-pki-106-32.inBoundTunnel
11: (Enabled,Valid)     dropAllPublicTraffic
Number of Rule to display (0 for All) [0]? 1
Policy name:                           ike-pki-106-32.p2in
Policy Loaded from:                    Local Configuration
Policy state:                          Enabled and Valid
Policy Priority:                       94

Profile Name    = 106->32.p2in
        sAddr:End = 192.168.141.18 : 192.168.141.18  sPort=  500 : 500
        dAddr:End = 192.168.141.17 : 192.168.141.17  dPort=  500 : 500
        proto     =            17 : 17
        TOS       =           x00 : x00
        Remote Grp=All Users

Validity Name   = always
        Duration  = Forever
        Months    = ALL
        Days      = ALL
        Hours     = All Day

IPSECAction Name = ike-1
        Tunnel Start:End         = 192.168.141.17 : 192.168.141.18
        Tunnel In Tunnel         =            No
        Min Percent of SA Life   =            75
        Refresh Threshold        =            85 %
        Autostart                =            No
        DF Bit                   =          COPY
        Replay Prevention        =      Disabled
        IPSEC Proposals:
        ----------------
        1:Name = esp-prop1
                Pfs  = N
                ESP Transforms:
                --------------
                1:Name = esp-trans1
                        Mode     = Tunnel
                        LifeSize = 50000
                        LifeTime = 3600
                        Authent  = SHA          Encr =DES
VPNRTR2 Policy console>
```

Other useful commands:

- >TALK 5
- >+FEATURE IPSec
- IPSec>IKE
- VPNRTR2 IKE>LIST TUNNEL
- VPNRTR2 IKE>LIST ALL
- VPNRTR2 IKE>STATS

Router to Router VPN Using Digital Certificates

If you are going to perform authentication using digital certificates you will need to have a certificate authority (CA). This is typically a software package running on a PC or UNIX platform. Note that in this release only one CA is supported, so all of your certificates for the entire network have to be issued by the same instance of the software package. Many companies sell CA software—for example, Entrust Technologies, Inc. and VeriSign.

To obtain a certificate, the router must have a private and public key. These are generated when the certificate request is issued from talk 5. Once the keys are generated, the router forms a certificate-request packet. This contains the router's public key and an identifier. The request is then sent to a TFTP server running somewhere in the network. The certificate-request must then be passed to CA and be read and processed by the CA. The CA will issue a certificate. The certificate contains the router's public key, the identifier sent by the router and a validity period. The certificate is signed by the CA's private key.

The router must then retrieve this certificate, either via TFTP or via LDAP. When the router downloads the certificate, the private key that is the partner to the public key in the certificate must still be in the router's running memory. The downloaded certificate is useless if the router has lost its matching private key. This means that from the time you issue the certificate request to the time the certificate downloads, you must not restart or reload the router, clear the cache, or issue a new certificate request. Any of these operations destroy the private key. The keys and certificate should be saved as soon as the certificate has been retrieved.

The router also needs to have a copy of the CA's certificate. When the router verifies a peer's certificate, it must confirm that the peer's certificate was signed by the CA's private key. To be able to do this, it must have the CA's certificate which contains the CA's public key. Each router performing IKE must download the CA's certificate using either TFTP or LDAP. This certificate must also be saved.

This example explains how to configure IBM routers for IP security with automatic key negotiations using digital signatures to provide authentication. The tunnel is from router to router. This tunnel will authenticate and encrypt traffic from specific hosts and will exclude all other traffic. The ***profile*** describes exactly which hosts at either end of the tunnel are allowed to pass data through the tunnel. The ***policy*** can allow a single host on either end, single or multiple subnets on either end, or any combination of the two. The limitation of router to router tunneling is that no authentication or encryption occurs on the LAN. This solution does not provide security on the LAN.

Refer to Figure 64 on page 291 for the physical network connectivity. Refer to Figure 65 on page 292 for the logical network diagram and IP Addressing. Documentation and screen captures will be given for only the parameters that are different from the example in “IPSec Router to Router VPN Using Pre-Shared Keys” on page 291.

Note: A user does not need to be defined in this case. The authentication will be provided by the digital certificate.

Create a Policy for the IPSec Tunnel for VPNRTR1

The steps for this example will be very similar to those described in "Create a Policy for the IPSec Tunnel for VPNRTR1" on page 292. To create this configuration, begin with the step titled **Enable IPSecurity** and continue the steps exactly until the end of the step titled **Add ISAKMP Action**. The next step, **Add ISAKMP Proposal**, will be different.

Except as noted in the following steps, these steps are the same as for the pre-shared keys example given in "Create a Policy for the IPSec Tunnel for VPNRTR1" on page 292. When using this method, do not use the **Add User** command to create a user and the keys. When creating the profile, you can configure it the same as the previous example, but be sure to consider the note.

1. Enable IP security
2. Add a policy
3. Add a profile

   Note: When adding the profile, you are prompted to configure IDs for ISAKMP. You must do this so that the other peer can identify you. The method chosen here must match the **subject-alt-name** type and information entered in the **CERT-REQ** command shown in Table 111 on page 318. The information must also match what is sent to the Certificate Authority as shown in Figure 70 on page 320.

4. Add a validity period
5. Add an IPSec action
6. Add an IPSec proposal
7. Add an ESP transform
8. Add an ISAKMP action

The following steps will be different than those in the pre-shared keys example. You will begin by adding a new ISAKMP Proposal to specify the authentication method RSA SIG. RSA SIG is a term for digital certificates. Next, you will request and load a router certificate and a CA certificate.

1. Add an ISAKMP proposal
2. Configure TFTP server for loading certificates
3. Request a router certificate
4. Load router certificate
5. Save the router certificate
6. Obtain a CA certificate
7. Load the CA certificate
8. Save the CA certificate

Add an ISAKMP Proposal

The ISAKMP proposal specifies the encryption and authentication attributes of the phase I SA. It also specifies which Diffie-Hellman group to use to generate the keys and the life of the phase I security.

Table 109. Add the ISAKMP Proposal for Digital Certificates

```
VPNRTR1 Policy config>ADD ISAKMP-PROPOSAL
Enter a Name (1-29 characters) for this ISAKMP Proposal []? cert1      1

List of Authentication Methods:        2
     1)  Pre-Shared Key
     2)  RSA SIG

Select the authentication method (1-2) [1]? 2

List of Hashing Algorithms:
     1)  MD5
     2)  SHA

Select the hashing algorithm(1-2) [1]?

List of Cipher Algorithms:
     1)  DES
     2)  3DES

Select the Cipher Algorithm (1-2) [1]?
Security Association Lifesize, in kilobytes (100-65535) [1000]?
Security Association Lifetime, in seconds (120-65535) [15000]?

List of Diffie Hellman Groups:
     1)  Diffie Hellman Group 1
     2)  Diffie Hellman Group 2

Select the Diffie Hellman Group ID from this proposal (1-2) [1]?


Here is the ISAKMP Proposal you specified...

Name = cert1
        AuthMethod = RSA SIG
        LifeSize   = 1000
        LifeTime   = 15000
        DHGroupID  = 1
        Hash Algo  = MD5
        Encr Algo  = DES CBC
Is this correct? [Yes]:
```

1. The ISAKMP proposal will be given a name.
2. Specify to use digital certificates.

Configure TFTP Server for Loading Certificates

The **Load Certificate** command requires that a TFTP server be pre-defined. Use the **Add Server** command to assign a name and IP address. It is a good idea to check connectivity between the router and the TFTP sever before attempting the load certificate operation.

Table 110. Add the TFTP Server Description for Loading Certificates

```
VPNRTR1 Config>FEATURE IPSec
IP Security feature user configuration
VPNRTR1 IPsec config>PKI
VPNRTR1 PKI config>ADD SERVER
Name ? (max 65 chars) []? TFTPServer
Enter server IP Address []? 9.24.106.146
Transport type (Choices: TFTP/LDAP)  [TFTP]?
VPNRTR1 PKI config>EXIT
```

Request a Router Certificate

Before requesting a certificate, it is important to make sure the clock of the router to which you will load the certificate is close to but no later than the clock of the CA system. Refer to “Chapter 19. Virtual Private Networks” on page 275 for an explanation of Certificate Authorities. When a CA issues a certificate, it will be time-stamped with a valid period expressed as beginning and ending time and date. The time of the router must be after the start time of the certificate and before the end time. If the certificate is being issued by a host not under your control, then the only way you can learn the time stamp of the certificate is to request it a try to load it to the router. If the time for the router is outside of the validity period, the following message will be displayed in the ELS log.

```
PKI.009 Validity check: failed Current date 1999/3/5, Time 9:38.21.
Cert valid date: 1999/3/5 10:14:38 -- 1999/6/5 10:14:38
```

This message informs you that the router time is earlier than the valid certificate time. If this occurs, check the router time using the T 6 command **time list** to display the time and the command **time set** to adjust the time.

The **CERT-REQ** command is used to create a certificate request which will be sent to the CA.

Table 111. Request the Certificate

```
VPNRTR1  *TALK 5
VPNRTR1 +FEATURE IPSec
VPNRTR1 IPSP>PKI
VPNRTR1 PKI Console>CERT-REQ
Enter the following part for the subject name
   Country Name(Max 16 characters) []? us
   Organization Name(Max 32 characters) []? cert
   Organization Unit Name(Max 32 characters) []?
   Common Name(Max 32 characters) []? VPNRTR1        1
Key modulus size (512|768|1024)
 [512]?
Certificate subject-alt-name type:     2
   1--IPv4 Address
   2--User FQDN
   3--FQDN
Select choice [1]?
Enter an IPv4 addr) []? 192.168.141.18       3
Generating a key pair. This may take some time. Please wait ...
Cert Request format: 1--DER;2--PEM      4
 [1]? 2
PKCS10 message successfully generated
Enter tftp server IP Address []? 9.24.106.146
Remote file name (max 63 chars) [/tmp/tftp_pkcs10_file]? test.req
Memory transfer starting.
.Memory transfer completed - successfully.
Certificate request TFTP to remote host sucessfully.     5
Generated private key stored into cache
Please download router certificate and save
both router certificate and its private key ASAP.
VPNRTR1 PKI Console>
```

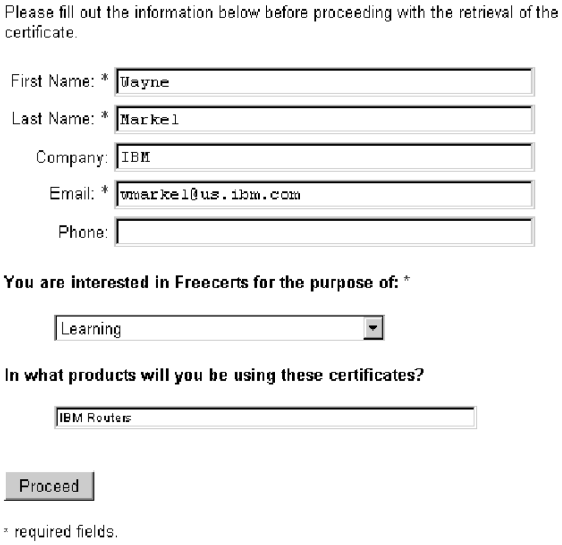
1. This name should match the actual configured router system name.
2. The type must match the ID type specified in the profile.
3. This must be the local tunnel endpoint address. The IP address of the serial interface is directly on the Internet.
4. The certificate request format must match the format that the CA uses to create the certificate. DER is digital format and PEM is ASCII format.

5. The certificate request is now on the TFTP server as test.req.

Obtaining the Certificates from the CA

The certificate request should be sent to the CA server which will verify the request and issue a certificate. The certificate contains the router public key and the information that you entered. The CA signs the certificate with a private key and it becomes trusted digital information.

Open the *test.req* document in Word Pad as shown in Figure 69.



Please fill out the information below before proceeding with the retrieval of the certificate.

First Name: * Wayne

Last Name: * Markel

Company: IBM

Email: * wmarkel@us.ibm.com

Phone:

You are interested in Freecerts for the purpose of: *

Learning

In what products will you be using these certificates?

IBM Routers

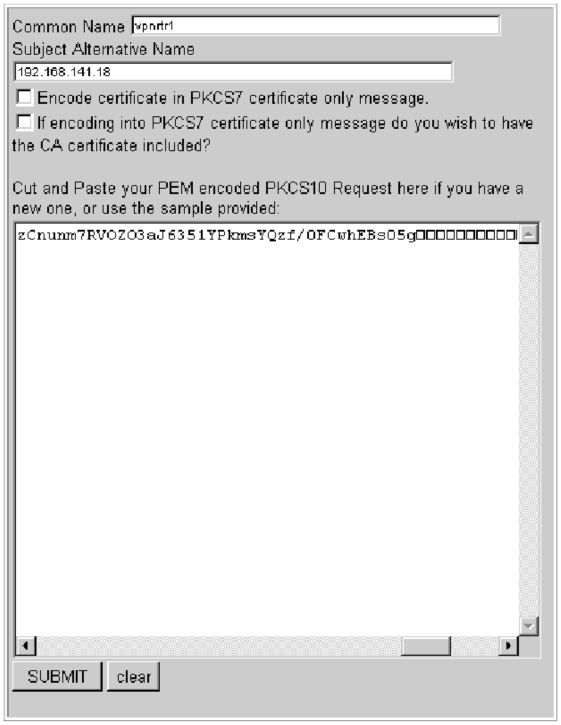
* required fields.

Figure 69. Certificate Request Created by the Router

For this example, Entrust Technologies was used although you could use any Certificate Authority. The steps to obtain the certificate may be different than the steps listed here.

Remember that when doing a cut and paste, only cut and paste the header and footer and characters in between. The header should begin with dashes and the footer should end with dashes.

On the company's Web Site, select **Request a VPN Certificate**. Fill out the disclaimer form and click **PROCEED**. On the next form, scroll down to the input area as shown in Table 111 on page 318, and fill in the common name (which in this example is VPNRTR1). Uncheck the box labeled **Encode certificate in PKCS7 certificates only message**. Enter the alternative name which should match reference 2 in Table 111 on page 318 for which we entered 192.168.141.18. Delete the pre-entered text in the cut and paste area. Using Word Pad, open the *test.req* file and cut and paste the certificate request into the window provided on the Web page. The certificate is pasted with no carriage returns.



Common Name vpnrtr1

Subject Alternative Name

192.168.141.18

□ Encode certificate in PKCS7 certificate only message.

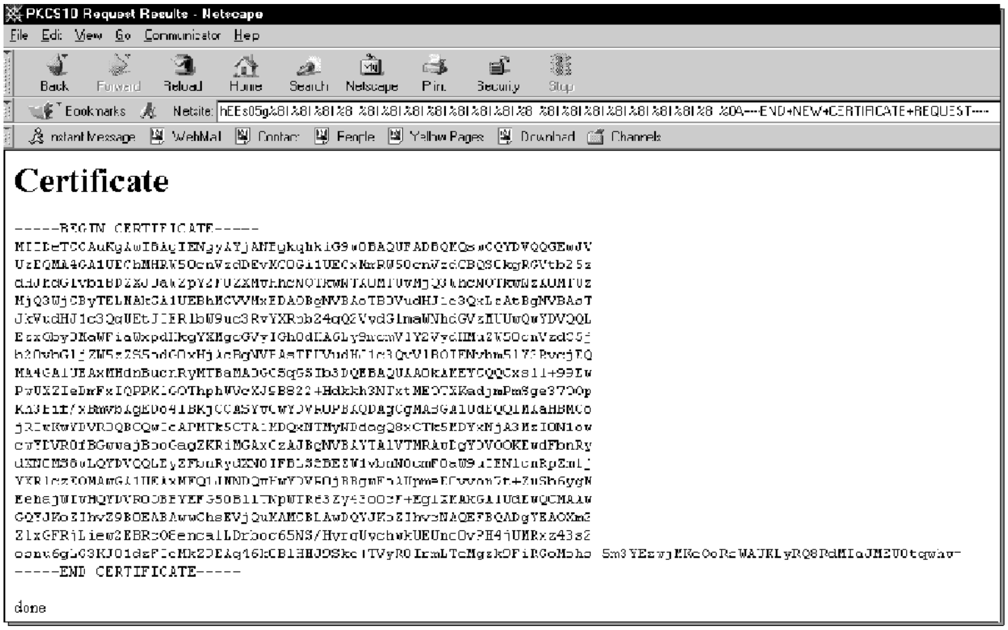
□ If encoding into PKCS7 certificate only message do you wish to have the CA certificate included?

Cut and Paste your PEM encoded PKCS10 Request here if you have a new one, or use the sample provided:

zCnunm7RVOZO3aJ6351YPkmsYQzf/0FCwhEBs05g□□□□□□□□□□

Figure 70. Fill Out the Certificate Request Form

A certificate will be returned in the Web browser as shown in Figure 71.



PKCS10 Request Results - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Stop
Bookmarks
Netsite: hEEs05g%81%81%81%8 %81%81%81%81%81%81%81%8 %81%81%81%81%81%81%81%8 %0A----END+NEW+CERTIFICATE+REQUEST----
Instant Message WebMail Contact People Yellow Pages Download Channels

Certificate

```
-----BEGIN CERTIFICATE-----
MIIDeTCCAuKgAwIBAgIENgyAYjANBgkqhkiG9w0BAQUFADBQMQswCQYDVQQGEwJV
UzEQMA4GA1UEChMHRW50cnVzdDEvMCOGA1UECxMmRW50cnVzdCBQSCkgRGVtb25z
dHJhdGlvbiBDZXJ0aWZpY2F0ZXMvRhcNOTkwNTAUMTUvMjQ3WhcNOTkwNzAUMTUz
MjQ3WjCByTELMAkGA1UEBhMCVVMxEDAOBgNVBAoTB0VudHJ1c3QxLzAtBgNVBAsT
JkVudHJ1c3QgUEtJIER1bW9uc3RyYXRpb24gQ2VydGlmaWNhdGVzMUUwQwYDVQQL
EzxGby9KaWFiaWxpdHkgYXMgcGVyIGh0dHA6Ly9ucmV1Y2VydHMuZW50cnVzdC5j
b20vbGljZW5zZS5odG0xHjAcBgNVBAsTFEVudHJ1c3QvV1BOIENvbm51Y2RvcjEQ
MA4GA1UEAxMHdnBucnRyMTBaMA0GCSqGSIb3DQEBAQUAAOkAMEYCQQCxs11+99Ew
PwUXZIeDmFxIQPRKiGOThphWVcXJ9B822+Hdkkh3NTxtME0TXKadjmPm9ge37D0p
Ka3E1f/xBmwbAgED041BKjCCASYwCwYDVR0PBAQDAgCgMA8GA1UdEQQIMAaHBMCo
jRIwKwYDVR0QBCQwIcAPMTk5CTA1MDQxNTMyNDdagQ8xCTk5MDYxNjA3MzIONlow
cwYDVR0fBGwwajBooGagZKRiMGAxCzAJBgNVBAYTA1VTMRAwDgYDVQQKEwdFbnRy
dXNCMS8wLQYDVQQLEyZFbnRydXN0IFBLS2BEEW1vbnN0cmF0aW9uIEN1cnRpZml
YXR1czEOMAwGA1UEAxMFQ1JNNDQwFwYDVR0jBBgwFnAUpw=ECvvon7t+ZuSb6ygN
EehejWIvHQYDVR0OBBYEFG50B11TNpWIR63Zy43oO3F+Eg1ZMAkGA1UdEwQCMAAw
GQYJKoZIhvZ9B0EABAwwChsEVjQuMAMCBLAwDQYJKoZIhvcNAQEFBQADgYEAOXm2
ZlxGFRjLiew2EBRz0EencalLDrboc65NS/HvrqUvchwkUEUncOv?H4jUMRxz43s2
oonv6gLC3KJ01dzFIcMkZDEAq16kCB1HHJ0Skc1TVyR0IrmLTcMgzkOFiRGoMohs 5m3YEzvjMKcOoRcWAJKLyRQ8RdMIaJMZU0tqwav-
-----END CERTIFICATE-----
```

done

Figure 71. The Router Certificate is Returned to the Browser

Cut and paste the certificate into a new text document in Word Pad. Delete spaces at the end of the first, next to last, and last line. Save the certificate in the TFTP Server Upload Directory. For this example, the certificate file was named, *cert.txt*.

Load Router Certificate

The certificate should now be retrieved via LDAP or TFTP. The following scenario uses TFTP to retrieve the certificate. As shown in Table 112, use the **Load Certificate** command to retrieve the router's certificate. Take the default option for the type of certificate since you are retrieving the router's certificate. You are then asked if the certificate is in digital format (option 1) or ASCII format (option 2). Select option 2. Next you are asked for the server name. This is the name of the TFTP server which you added from talk 6. Lastly, you will be asked for the name of the file on the server. The router then retrieves the certificate and stores it in its runtime memory.

Table 112. Load the Router Certificate

```
VPNRTR1 PKI Console>LOAD CERTIFICATE
Enter the type of the certificate:
Choices:  1-Root CA Cert, 2-Router Cert
Enter (1-2): [2]?
Encoding format:
Choices: 1-DER 2-PEM
Enter (1-2): [1]? 2
Server info name []? TFTPServer
Remote file name on tftp server (max 63 chars) [/tmp/default_file]? cert.txt

Attempting to load certificate file. Please wait ...
Memory transfer starting.
.Memory transfer completed - succesfully.
Router Certificate  loaded into run-time cache
VPNRTR1 PKI Console>
```

Save the Router Certificate

Immediately save the certificate and associated keys. You will have to repeat the certificate process if you fail to save the certificate and the router restarts. You will be asked which certificate you are saving, what name you wish to call it and if you wish this certificate to be loaded into the router's memory when the router is started.

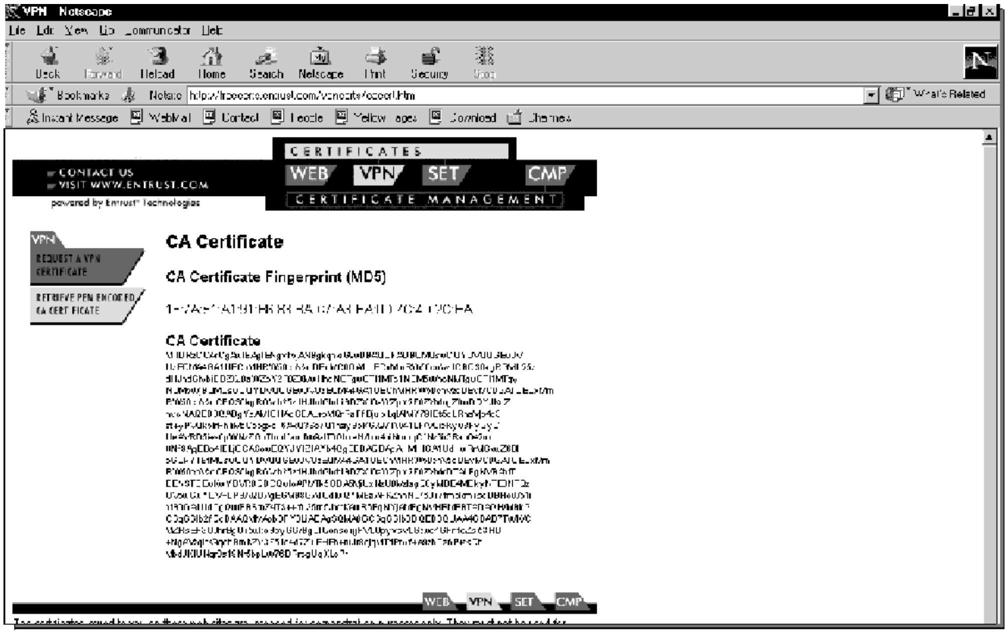
Table 113. Save the Router Certificate

```
VPNRTR1 PKI Console>CERT-SAVE
Enter type of certificate to be stored into SRAM:
     1)Root certificate;
     2)Box  certificate with private key;
Select the certificate type (1-2) [2]?
SRAM Name for certificate and private key []? r1cert.txt
Load as default router certificate at initialization? [No]: y
Both Router Certificate and private key saved into SRAM successfully
VPNRTR1 PKI Console>
```

Obtain a CA Certificate

The router now has its private and public keys, which were generated just before the certificate request was issued. You have just retrieved the router's certificate. Now you need the CA's certificate so you can verify the validity of an IKE peer's certificate. One part of confirming the validity is to check that the CA signed the peer's certificate, therefore the CA's certificate is needed. The peer's certificate must be signed by the same CA because there is no mechanism in place to check a certificate issued by another CA.

Next, **Retrieve PEM Encoded Certificate** was selected on the Entrust Web site. As shown in Figure 72, a CA certificate was returned to the browser. For this particular test, no header or footer was sent with the CA certificate.


VPN - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Stop
Bookmarks Netsite What's Related
Instant Message WebMail Contact People Yellow Pages Download Channels
CERTIFICATES
WEB VPN SET CMP
CERTIFICATE MANAGEMENT
CONTACT US
VISIT WWW.ENTRUST.COM
powered by Entrust Technologies
VPN
REQUEST A VPN CERTIFICATE
RETRIEVE PEM ENCODED CA CERTIFICATE

CA Certificate

CA Certificate Fingerprint (MD5)

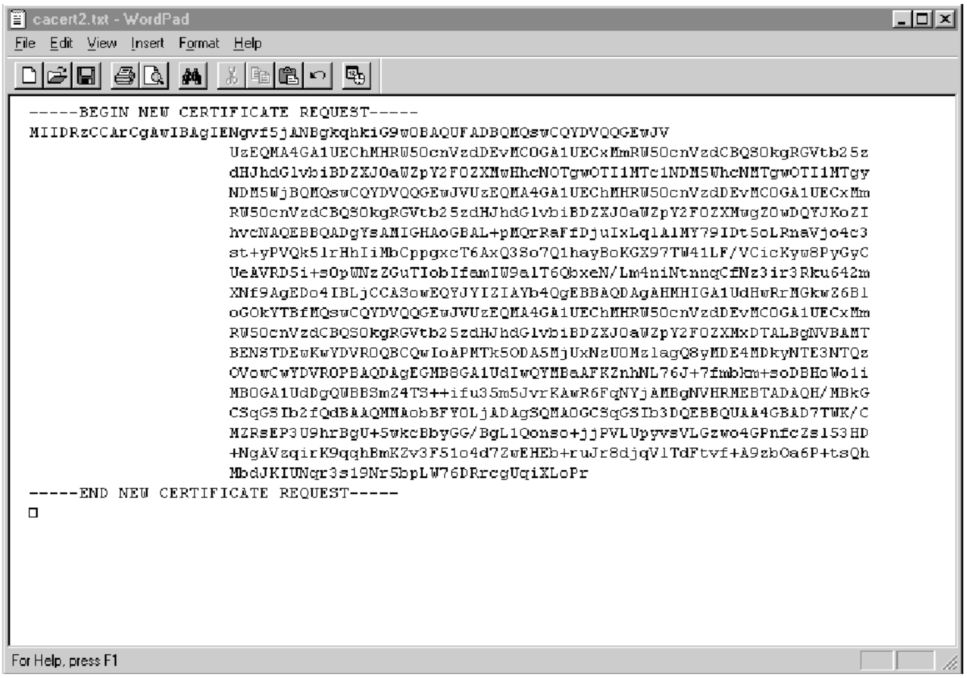
[illegible]

CA Certificate

[illegible]

Figure 72. The CA Certificate is Returned to the Web Browser

For this example, the text following ″CA Certificate″ was pasted into a new Word Pad text document. No header or footer was sent with the CA certificate. (This may vary depending on how you get the CA certificate.) In order for the router in this example to accept the certificate, the header and footer from the router certificate that was already received had to be pasted into the document and the CA certificate text was pasted in between the header and footer text as shown in Figure 73 on page 323.



```
-----BEGIN NEW CERTIFICATE REQUEST-----
MIIDRzCCArCgAwIBAgIENgvf5jANBgkqhkiG9w0BAQUFADBQMQswCQYDVQQGEwJV
                    UzEQMA4GA1UEChMHRW50cnVzdDEvMC0GA1UECxMmRW50cnVzdCBQS0kgRGVtb25z
                    dHJhdGlvbiBDZXJ0aWZpY2F0ZXMwHhcNOTgwOTI1MTc1NDM5WhcNMTgwOTI1MTgy
                    NDM5WjBQMQswCQYDVQQGEwJVUzEQMA4GA1UEChMHRW50cnVzdDEvMC0GA1UECxMm
                    RW50cnVzdCBQS0kgRGVtb25zdHJhdGlvbiBDZXJ0aWZpY2F0ZXMwgZ0wDQYJKoZI
                    hvcNAQEBBQADgYsAMIGHAoGBAL+pMQrRaFfDjuIxLqlAlMY79IDt5oLRnaVjo4c3
                    st+yPVQk5lrHhIiMbCppgxcT6AxQ3So7Q1hayBoKGX97TW41LF/VCicKyw8PyGyC
                    UeAVRD5i+sOpWNzZGuTIobIfamIW9alT6QbxeN/Lm4niNtnnqCfNz3ir3Rku642m
                    XNf9AgEDo4IBLjCCASowEQYJYIZIAYb4QgEBBAQDAgAHMHIGA1UdHwRrMGkwZ6Bl
                    oGOkYTBfMQswCQYDVQQGEwJVUzEQMA4GA1UEChMHRW50cnVzdDEvMC0GA1UECxMm
                    RW50cnVzdCBQS0kgRGVtb25zdHJhdGlvbiBDZXJ0aWZpY2F0ZXMxDTALBgNVBAMT
                    BENSTDEwKwYDVR0QBCQwIoAPMTk5ODA5MjUxNzU0MzlagQ8yMDE4MDkyNTE3NTQz
                    OVowCwYDVR0PBAQDAgEGMB8GA1UdIwQYMBaAFKZnhNL76J+7fmbkm+soDBHoWo1i
                    MB0GA1UdDgQWBBSmZ4TS++ifu35m5JvrKAwR6FqNYjAMBgNVHRMEBTADAQH/MBkG
                    CSqGSIb2fQdBAAQMMAobBFYOLjADAgSQMA0GCSqGSIb3DQEBBQUAA4GBAD7TWK/C
                    MZRsEP3U9hrBgU+5wkcBbyGG/BgL1Qonso+jjPVLUpyvsVLGzwo4GPnfcZsl53HD
                    +NgAVzqirK9qqhBmKZv3F51o4d7ZwEHEb+ruJr8djqVlTdFtvf+A9zbOa6P+tsQh
                    MbdJKIUNqr3s19Nr5bpLW76DRrcgUqiXLoPr
-----END NEW CERTIFICATE REQUEST-----
□
```

Figure 73. Add the Header and Footer to the CA Certificate

Save the certificate as a document in the TFTP server upload directory. For this example, the file was named *cert.txt*.

Load the CA Certificate

The CA’s certificate can also be loaded via TFTP using the **Load Certificate** command and selecting option 1 as the type of certificate.

Table 114. Load the Root Certificate to Cache

```
VPNRTR1 PKI Console>LOAD CERTIFICATE
Enter the type of the certificate:
Choices:  1-Root CA Cert, 2-Router Cert
Enter (1-2): [2]? 1
Encoding format:
Choices: 1-DER 2-PEM
Enter (1-2): [1]? 2
Server info name []? TFTPServer
Remote file name on tftp server (max 63 chars) [/tmp/default_file]? cacert.txt

Attempting to load certificate file. Please wait ...
Memory transfer starting.
.Memory transfer completed - successfully.
Root CA Certificate  loaded into run-time cache
VPNRTR1 PKI Console>
```

Save the CA Certificate

Table 115. Save the Root Certificate to the Router Configuration

```
VPNRTR1 PKI Console>CERT-SAVE
Enter type of certificate to be stored into SRAM:
     1)Root certificate;
     2)Box  certificate with private key;
Select the certificate type (1-2) [2]? 1
SRAM Name to store Root Certificate? []? cacert
Load as default root certificate at initialization? [No]: y
Root Certificate saved into SRAM successfully.
VPNRTR1 PKI Console>
```

Once you have saved the CA certificate, you have completed configuration of the VPNRTR1 tunnel policy.

Create a Policy on VPNRTR1 to Drop Public Traffic

The steps to configure this policy are exactly the same as the step in the example shown in “Create a Policy on VPNRTR1 to Drop Public Traffic” on page 305:

1. Add the policy
2. Add the profile
3. Specify the interfaces
4. Add the validity period
5. Add the IP security action
6. Confirm the policy

This completes the configuration of VPNRTR1. Save copies of the configuration because you will not want to take the time to do this again.

Create a Policy for the IPSec Tunnel for VPNRTR2

To create the IP Security tunnel, follow these steps:

1. Enable IP security
2. Add a policy
3. Add a profile
4. Add a validity period
5. Add an IPSec action
6. Add an IPSec proposal
7. Add an ESP transform
8. Add an ISAKMP action
9. Add an ISAKMP proposal
10. Configure TFTP server for loading certificates
11. Request a router certificate
12. Load router certificate
13. Save the router certificate
14. Obtain a CA certificate
15. Load the CA certificate
16. Save the CA certificate

The steps above are the same as the steps shown in “Create a Policy for the IPSec Tunnel for VPNRTR1” on page 316 with the following differences:

- The profile for VPNRTR2 tunnel policy reverses the sAddr:Mask and the dAddr:Mask used with VPNRTR1.
- The IPSec Action for VPNRTR2 reverses the Tunnel Start:End points.

Create a Policy on VPNRTR2 to drop public traffic

The steps for creating this policy are the same as the steps in “Create a Policy on VPNRTR2 to Drop Public Traffic” on page 312.

Monitoring/Troubleshooting from Talk 5

Monitoring operations and statistics for this example are the same as for the pre-shared keys example. Refer to “Monitoring/Troubleshooting the Policies” on page 312.

Voluntary PPTP Tunnel with IBM Router Termination

Refer to “Point-to-Point Tunneling Protocol” on page 289 for additional information about PPTP.

IBM routers support PPTP in order to provide interoperability with Microsoft Windows devices which support only PPTP. Microsoft has announced intentions to implement L2TP on NT 5.0.

Figure 74 on page 326 is an example of a remote access VPN using PPTP voluntary tunneling. The IBM router will be configured as the endpoint of a PPTP tunnel. The client, a Windows/98, Windows/95 or Windows NT Dial-Up Networking (DUN) client will dial into the ISP router. The client will establish a PPP connection and be given an IP address on the 9.24.104.0 subnet. At this point, the client has IP connectivity to anywhere in the Internet IP cloud including the WAN interface of the corporate Internet router. The client will then establish a tunnel to 192.168.141.18, which is the IP address of the corporate Internet router. The userid and password for the PPTP tunnel is *sg245281* and the IP address is 192.168.141.38. These are assigned by the corporate router. Once the tunnel is established, connectivity is the same as you would have if you dialed directly into a Remote Access Server on the corporate LAN.

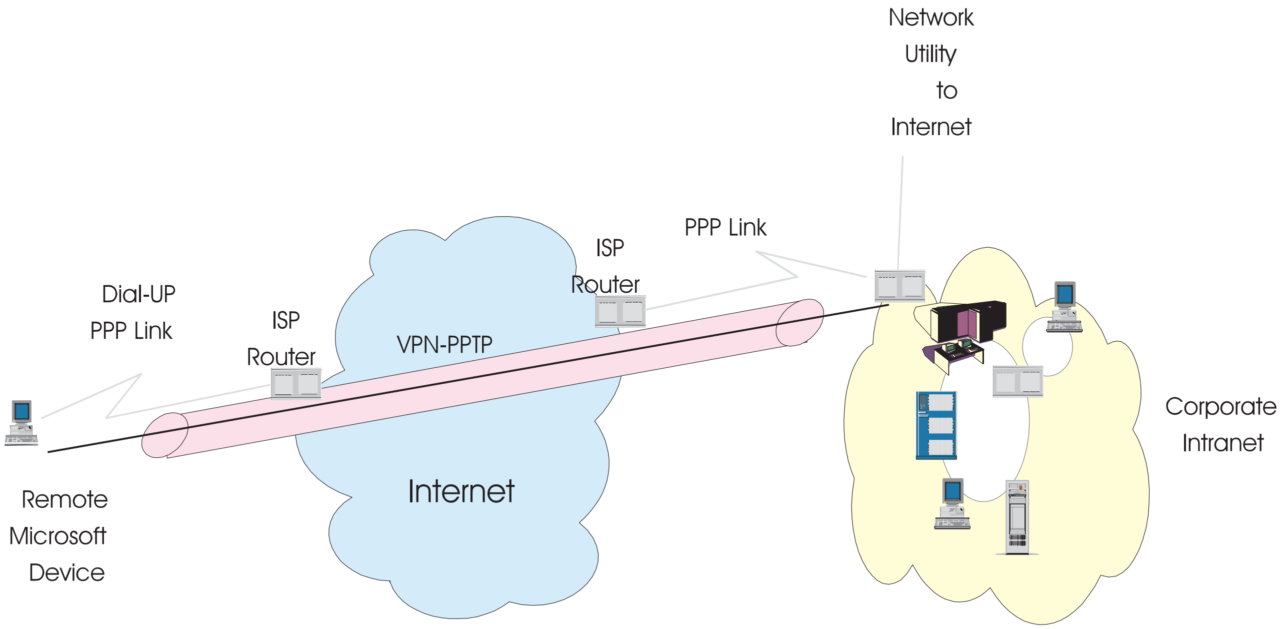


Figure 74. Workstation to Gateway PPTP Tunnel

Configuration of the Network Utility

Before completing the following steps, make sure the Network Utility has the proper interfaces configured. Also configure IP so that the PPP interface has either a static or dynamic route to the Internet. The Intranet interface should not be advertised on the Internet. Refer to Figure 75 on page 327 for the IP addressing of the network that was used for this example.

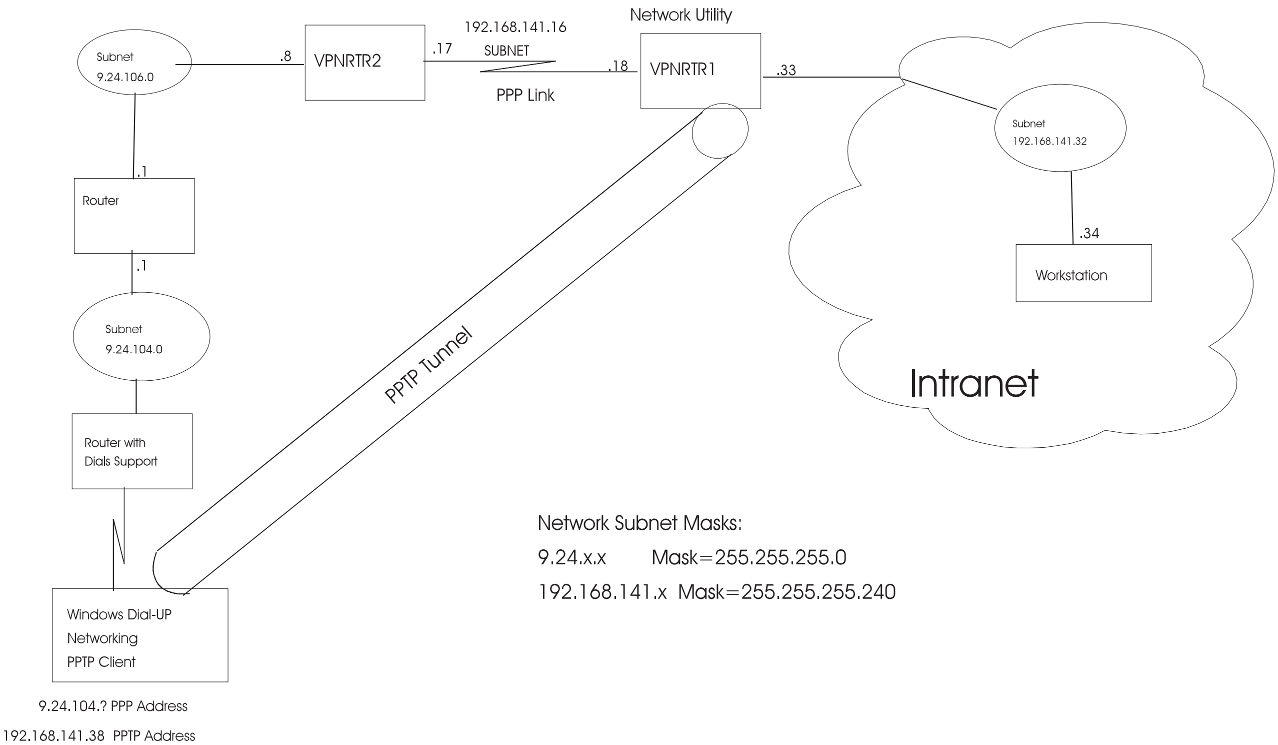


Figure 75. IP Addressing Scheme

To configure the corporate Internet router as illustrated in Figure 74 on page 326, follow these steps:

- Enable PPTP
- Add L2-Nets
- Enable mschap and mppe
- Add PPP USER
- Enable arp subnet routing
- Configure the Dial-UP Networking (DUN) client

Enable PPTP

Table 116. Enable PPTP

```
VPNRTR1  *TALK 6
VPNRTR2 Config>FEATURE Layer-2-Tunneling
VPNRTR2 Layer-2-Tunneling Config>ENABLE PPTP

 Restart system for changes to take effect.
```

Add Layer 2 Nets

Add the number of Layer 2 Nets needed to support the maximum number of simultaneous connections. In this example, 3 nets are added. You do not need to enable IPX or transparent bridging.

Table 117. Add a Layer-2 Net

```
VPNRTR1 *TALK 6
VPNRTR2 Config>FEATURE Layer-2-Tunneling
VPNRTR2 Layer-2-Tunneling Config>ADD L2-NETS
Additional L2 nets:  [0]? 3       1
Add unnumbered IP addresses for each L2 net? [Yes]:
Adding device as interface 6
Defaulting Data-link protocol to PPP
Adding device as interface 7
Defaulting Data-link protocol to PPP
Adding device as interface 8
Defaulting Data-link protocol to PPP
Enable IPX on L2T interfaces?(Yes or [No]):
Enable transparent bridging on L2T interfaces?(Yes or [No]):
Bridge configuration was not changed.

Restart router for changes to take affect.
VPNRTR2 Layer-2-Tunneling Config>
```

1. Add the number of nets to equal the planned maximum number of simultaneously attached PPTP clients.

Enable mschap and mppe

Microsoft Windows Dial-UP Networking (DUN) PPTP clients use MPPE to perform encryption. This protocol needs to be enabled on the L2Net. L2Nets configured as inbound from anyone (the default) take their PPP defaults from a template in the layer feature. The **encapsulator** command takes you to a prompt from where all the PPP defaults can be tuned.

To use MPPE, you must enable MS-CHAP. When you enable MPPE, you will be asked if MPPE is operating in mandatory or optional mode. If it is operating in mandatory mode, you must negotiate MPPE. Mandatory mode forces the router to renegotiate MPPE each time a new connection is requested, even when the sender has previously established MPPE between itself and the router. If MPPE is operating in optional mode, you are not obligated to negotiate MPPE. Optional mode results in the router maintaining MPPE between itself and the sender after the initial negotiation and does not renegotiate MPPE for each new connection. You will then be asked if the keys are stateful or stateless. If the keys are stateless, the key changes every time a packet is sent, whereas with stateful the key is only generated when 255 packets have been sent. Stateless is advised for lossy networks and should be used for PPTP connections. The router will know if the client is using stateless or stateful mode since part of the MPPE header indicates whether the keys have been refreshed.

Microsoft also has their own compression algorithm, MPPC. MPPE is negotiated as an MPPC option. If you wish to do compression and you are using MPPE, you must use MPPC. In this case, you cannot use the Stac-LZS algorithm which is normally available for PPP links. If you chose not to use MPPC, the router code partially enables the function to allow MPPE to be negotiated. If you do chose to use MPPE and MPPC, they are decoded in one pass since these protocols share the same PPP header.

Table 118. Enable MSCHAP and MPPE

```
VPNRTR1 Layer-2-Tunneling Config>ENCAPSULATOR
Point-to-Point user configuration
VPNRTR1 PPP-L2T Config>ENABLE MSCHAP
Rechallenge Interval in seconds (0=NONE)  [0]?
Enabling MSCHAP
VPNRTR1 PPP-L2T Config>ENABLE MPPE
mandatory or optional [optional]?
stateful or stateless [stateful]? stateless      1
Enabling encryption

** Note ** : To view the MPPE configuration, please enter a 'list ccp'
             command since MPPE is negotiated within the CCP protocol.
VPNRTR1 PPP-L2T Config>
```

1. If the keys are stateless, the key changes every time a packet is sent, whereas with stateful the key is only generated when 255 packets have been sent. Stateless is advised for lossy networks and should be used for PPTP connections.

Add PPP User

For this example, two users are configured so that at least two simultaneous connections are tested. Each user has been assigned a static IP address. This is the simplest but also the least flexible and least scaleable way to assign IP addresses to PPP clients. Other methods of assigning IP addresses are to use an IP address pool or to use DHCP services.

First the user named *sg245281* is added. The password entry will not appear on the screen.

Table 119. Add the PPP User

```
VPNRTR2 Config>ADD PPP-USER
Enter name:  []? sg245281
Password:
Enter again to verify:
Allow inbound access for user? (Yes, No): [Yes]
Will user be tunneled? (Yes, No): [No]
Is this a 'DIALs' user? (Yes, No): [Yes]
Type of route?  (hostroute, netroute): [hostroute]
Number of days before account expires [0-1000] [0]?
Number of grace logins allowed after an expiration [0-100] [0]?
IP address: [0.0.0.0]? 192.168.141.38      1
Enter hostname: []?
Allow virtual connections? (Yes, No): [No]
Give user default time allotted ? (Yes, No): [Yes]
Enable callback for user? (Yes, No): [No]
Will user be able to dial-out ? (Yes, No): [No]
Set ECP encryption key for this user? (Yes, No): [No]
Disable user ? (Yes, No): [No]

     PPP user name: sg245281
   User IP address: 192.168.141.38
     Netroute Mask: 255.255.255.255
          Hostname:       Virtual Conn: disabled
      Time alotted: Box Default
     Callback type: disabled
          Dial-out: disabled
        Encryption: disabled
            Status: enabled
    Login Attempts: 0
    Login Failures: 0
  Lockout Attempts: 0
    Account Expiry:    Password Expiry:
Is information correct? (Yes, No, Quit): [Yes]

User 'sg245281' has been added
VPNRTR2 Config>
```

1. Manually assigned IP address for PPTP client

Add another ppp user named *salesman* using the same parameters, and then list all ppp users.

Table 120. List PPP-USERS

```
VPNRTR2 Config>LIST PPP-USERS addr
List  (Name, Verb, User, Addr, VCon, Call, Time, Dial, Encr): [User] addr

PPP user name      User IP address     Netroute Mask       Hostname
-----------------  ------------------  ------------------  ---------------
salesman           192.168.141.39      255.255.255.255     <undefined>
sg245281           192.168.141.38      255.255.255.255           <undefined>
2 PPP records displayed.
```

Enable Arp Subnet Routing

Arp subnet routing is also referred to as proxy arp. If a host on the corporate network transmits a datagram to a PPTP host which has an IP address on the same subnet, the sender will not send the datagram to the default route, but will expect to see an entry in its own ARP cache. If there is no entry in the ARP cache, the sender will send ARP broadcasts directed at the destination IP. Since the destination IP address (the remote PPTP client) is not on the physical network, it will never respond. Arp subnet routing allows the local router to respond to the ARP broadcast on behalf of the remote client. The datagram then gets copied by the router and forwarded across the Internet.

Table 121. Enable Arp Subnet Routing

```
VPNRTR2 Config>PROTOCOL
Protocol name or number [IP]?
Internet protocol user configuration
VPNRTR1 IP config>ENABLE ARP-SUBNET-ROUTING
VPNRTR1 IP config>
```

Configure the DUN Client

To establish a PPTP connection using a Microsoft platform, you need two DUN sessions—one to the Internet—the ISP's router—and another to the Network Utility. You will first launch the PPP dial-up connection which establishes the Internet connection, and then launch the PPTP connection to create the tunnel to the Network Utility. The PPP interface of the IBM Network Utility must be accessible on the Internet.

Note: You must have DUN 1.2 or later installed. To see if you have Version 1.2 or later, open a **Make a New Connection** window in the DUN folder. Check for the presence of Microsoft VPN Adapter in the **Select a Device** drop down window.

To configure the Microsoft PPTP client, follow these steps:

- Add a DUN client. Configure it to use its modem to dial into the ISP router.
- Add a second DUN client. Configure it to use the VPN adapter to connect to the IP address of the corporate router's WAN interface. When you click on **Make a New Connection**, you will be asked for details about the adapter. You should use the Microsoft VPN adapter. The next screen will ask for the hostname or IP address. In this box, enter the IP address of the IBM router that is reachable via the IP cloud. When you launch that DUN connection, it will ask for a userid and password which should match the details configured with the **add ppp-user** command in the router configuration as shown in "Configuration of the Network Utility" on page 326.

When using the manually defined ppp-user, you must have a static IP address for each manually configured user.

You can define ppp-user/ip addr for each user and specify on the DUN to use sever assigned IP address. Otherwise, you can have one ppp-user and specify on the DUN to use the locally-assigned static IP address.

In the DUN client under the **Properties/Server Type/TCP/IP** settings, you can specify whether or not to use the default route on the remote network. What you specify here will depend on whether or not the PPTP client is accessing only resources on the remote subnet or whether it needs to have connectivity to other nets as well.

Monitoring

To verify that the configuration is correct, you can do the following ping tests. First, start the PPP link on the DUN client and ping the Internet interface of the Network Utility. The ping should be successful. Next try to ping the Intranet interface. The ping should fail. Now start the PPTP tunnel by launching the PPTP DUN definition. You should now be able to ping all hosts on the Intranet.

From the Talk 5 prompt, issue the **NETWORK 6** command and then the **LIST ALL** command to see an extensive amount of information about the PPP connection. The most useful for troubleshooting are the statistics, user id, and IP address of the connection.

As shown in Table 122, use the **CALL STATE** command. Before we issued the **CALL STATE** command, we established sessions with our two ppp-users.

Table 122. Display Layer-2 Sessions

```
VPNRTR1 Layer-2-Tunneling Console> CALL STATE
CallID | Serial # | Net # |    State    | Time Since Chg | PeerID | TunnelID
 55285 |        0 |     8 | Established |    0:37:10     |      0 |     6084
 38142 |        0 |     7 | Established |    0: 4:35     |      0 |    24721
VPNRTR1 Layer-2-Tunneling Console>
```

For monitoring and troubleshooting, use the following commands:

- VPNRTR1 Layer-2-Tunneling Console> **TUNNEL TRANSPORT**
- Issue the **TALK 2** command after setting up ELS with the **DISPLAY SUBSYSTEM L2 ALL ALL**

In Table 123, the output from the **TALK 2** command shows the two PPP nets with the CallID matching the information.

Table 123. Output of Talk 2 With Event Set for Display Subsystem L2

```
00:41:55   L2.024: PPTP PAYLOAD SEND 38 bytes, net=7, callid=38142
00:41:55   L2.041: SND PPTP:F=3081,L=54,Tid=0,Cid=0,NS=115,NR=117,O=0
00:41:55   L2.040: RCV PPTP:F=3081,L=38,Tid=24721,Cid=38142,NS=118,NR=115,O=0
00:41:55   L2.022: PPTP PAYLOAD RCVD 38 bytes, net 7, callid=38142
00:42:00   L2.024: PPTP PAYLOAD SEND 38 bytes, net=8, callid=55285
00:42:00   L2.041: SND PPTP:F=3081,L=54,Tid=0,Cid=0,NS=264,NR=274,O=0
00:42:00   L2.040: RCV PPTP:F=3081,L=38,Tid=6084,Cid=55285,NS=275,NR=264,O=0
00:42:00   L2.022: PPTP PAYLOAD RCVD 38 bytes, net 8, callid=55285
00:42:03   L2.084: PPTP Tunnel 6084/0 EVENT Rcv-ECHO,state=Established
```

See Table 124 for a partial listing of the **LIST ALL** command.

Table 124. Partial Output of List All Command

```
VPNRTR1 +NETWORK 8
Point-to-Point Console
       VPNRTR1 PPP 8>LIST ALL

Interface Statistic        In                           Out
-------------------        --                           ---
Packets:                   81                           70
Octets:                    3316                         2581
..
.Remote Username:           sg245281
.
..IPCP Option                Local                        Remote
-----------                -----                        ------
IP Address                 0.0.0.0                      192.168.141.38
Compression Slots          None                         None
```

IBM Network Utility Initiated Voluntary PPTP Tunnel

This example, which is illustrated in Figure 76, is a PPTP scenario where the IBM router initiates a PPTP tunnel terminated by a Microsoft NT Remote Access Server (RAS) which is a PPTP peer. The NT server has two adapters—one with an IP address which is reachable via the IP cloud and one on the private network. The NT host does not have a dynamic routing protocol configured, but does have IP forwarding capability.

The scenario will provide connectivity for IP hosts in the branch to hosts on a single subnet within the corporate network. The NT RAS is located in what is sometimes referred to as the DMZ. It is accessible from the Internet and not protected by the corporate firewall. The RAS itself becomes a firewall for the corporate subnet which will be accessed across the Internet.

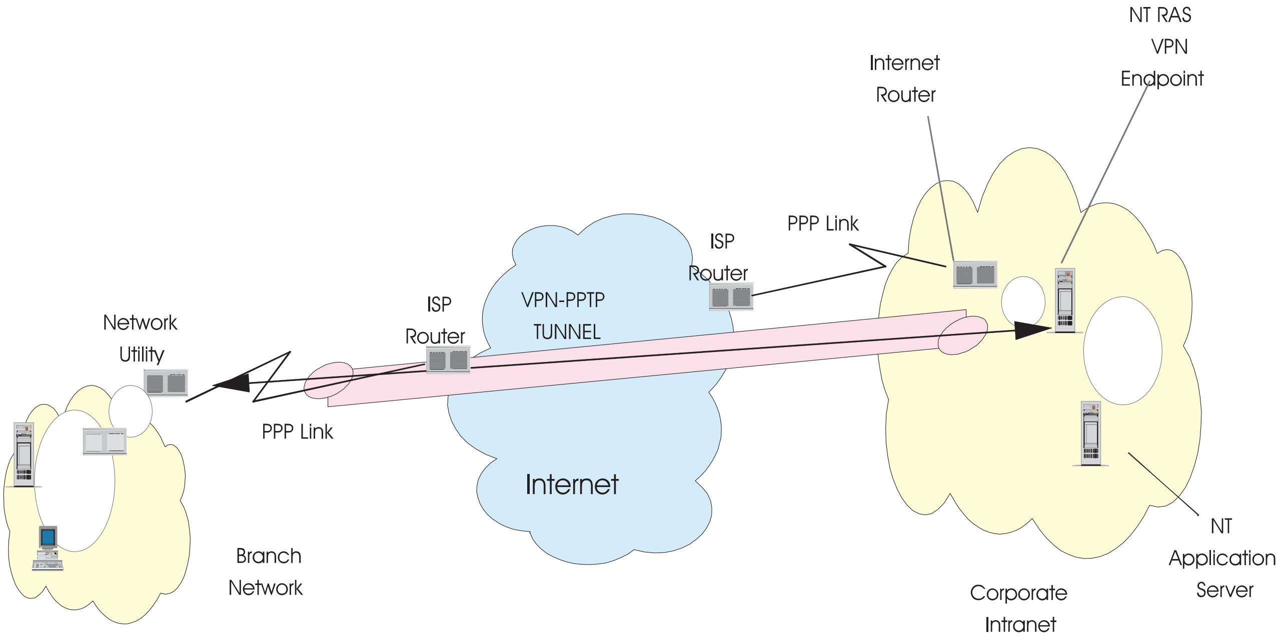


Figure 76. IBM Router Initiated PPTP Tunnel

The lab network used for this example consists of one PPP link and three token-ring segments connected by two IBM 2210 routers and an NT Workstation. We have named the routers VPNRTR1 and VPNRTR2. The routers in the lab are connected with a 56-Kbps PPP link. In a real scenario, the link between the routers could be any private or public WAN.

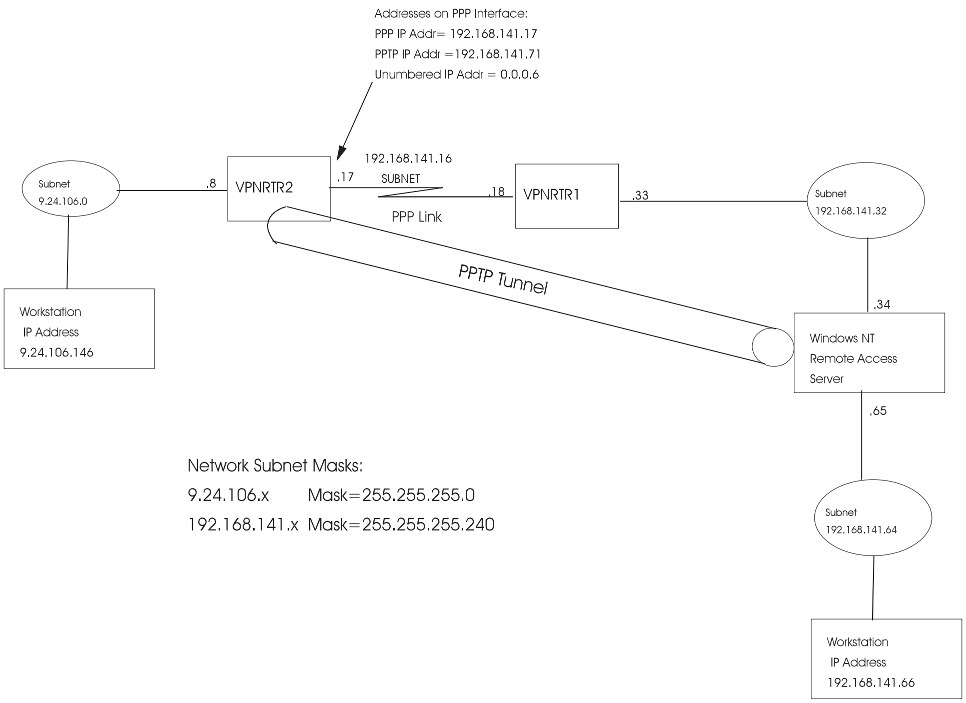


Figure 77. IP Addressing of Lab Network

A PPTP tunnel is established when a device on the 9.24.106.0 branch network has data to send to the corporate LAN IP network 192.168.141.64. The 192.168.141.64 network is not advertised into the IP cloud. Hosts on the corporate network are private addresses and the IP cloud is an ISP's network.

VPNRTR2, which represents the Branch Internet Router as illustrated in Figure 76 on page 333, will be configured with a static route that specifies traffic destined for the 192.168.141.64 network and should be routed via the virtual interface. When data is received on the interface, the router will establish a PPTP tunnel. The router examines the L2Net and tunnel definition to locate the IP address of the PPTP peer, 192.168.141.34, which is illustrated as the NT Remote Access Server and VPN Endpoint. The router will establish a TCP connection to this address via the Internet. After the NT has accepted the PPTP connection, the branch router will negotiate the PPP parameters for its L2Net. The NT server will return an IP address from a pool of addresses configured for that PPTP interface. The IP address has to be in the same subnet as the corporate LAN interface. The L2Net must be configured to receive its IP address via IPCP.

Configure the Branch Router

Here are the basic steps to configure the branch router:

- Enable PPTP
- Add the Tunnel Profile
- Set up the definitions for IP addressing and authentication
- Configure Network Address Translation
- Create Packet Filters

Enable PPTP on the router and add the virtual interface. When traffic is received on this interface, the router will initiate the PPTP tunnel.

Table 125. Add the Layer 2 Networks

```
VPNRTR2 *TALK 6

VPNRTR2 Config>FEATURE Layer-2-Tunneling
VPNRTR2 Layer-2-Tunneling Config>ENABLE PPTP

 Restart system for changes to take effect.
VPNRTR2 Layer-2-Tunneling Config>
Layer-2-Tunneling Config>add l2-nets
Additional L2 nets: [0]? 1
Add unnumbered IP addresses for each L2 net? [Yes]:
Adding device as interface 6        1
Defaulting data-link protocol to PPP
Enable IPX on L2T interfaces?(Yes or [No]):
Enable transparent bridging on L2T interfaces?(Yes or
[No]):
Bridge configuration was not changed.
Restart router for changes to take affect.
Layer-2-Tunneling Config>exit
VPNRTR2 Config>
```

1. Since we specified that the unnumbered IP address be added, 0.0.0.6 is assigned because the L2Net is interface 6. As shown in Table 126, you can use the **list addr** command at the IP Config> command prompt to verify the address. This address will be used as a parameter for the **enable dynamic** command as shown in Table 130 on page 337.

Table 126. List Address to Verify IP Address of PPTP Interface

```
VPNRTR2 Config>PROTOCOL IP
VPNRTR2 IP config>LIST ADDRESSES
IP addresses for each interface:
   intf     0                                    IP disabled on this interface
   intf     1  192.168.141.17   255.255.255.240  Local wire broadcast, fill 1
   intf     2                                    IP disabled on this interface
   intf     3                                    IP disabled on this interface
   intf     4                                    IP disabled on this interface
   intf     5  9.24.106.8       255.255.255.0    Local wire broadcast, fill 1
   intf     6  0.0.0.6          0.0.0.0          Local wire broadcast, fill 1
                                                 DYNAMIC-ADDRESS Enabled
VPNRTR2 Config>EXIT
```

The next step is to define the PPTP tunnel endpoint. The **add tunnel-profile** command is used to define the tunnel. The name you are prompted for is the name of the remote PPTP. This is only for local identification purposes. It is not sent during the PPTP exchanges. You are asked for the tunnel-server endpoint address—this is in the address of the NT server which is reachable via the IP cloud.

Table 127. Add the Tunnel

```
VPNRTR2 Config>ADD TUNNEL-PROFILE
Enter name:  []? NT
Tunneling Protocol?  (PPTP, L2F, L2TP): [L2TP] PPTP
Tunnel-Server endpoint address: [0.0.0.0]? 192.168.141.34

       Tunnel name: NT
          TunnType: PPTP
          Endpoint: 192.168.141.34

Tunnel 'NT' has been added
```

The next step is to tie our virtual interface to the peer called NT. By default, all L2Nets are inbound from any device. This must be changed to outbound. Then you are prompted for the name of the remote device. This means that when traffic is routed to our virtual interface, interface 6, the router will establish a tunnel to a peer called ″NT″. It looks at the ″NT″ tunnel definition and discovers that it is a PPTP tunnel to 192.168.141.34.

The router will look in its routing table to determine how to get to that address, which in this example will be via 192.168.141.17. The router needs to be configured either via static routing or a dynamic routing protocol to know how to get to 192.168.141.34.

Table 128. Configure the Virtual Interface

```
VPNRTR2 Config>NETWORK 6
Session configuration
VPNRTR2 L2T config:   6>SET CONNECTION-DIRECTION OUTBOUND     1
Enter remote tunnel hostname:  []? NT
VPNRTR2 L2T config:   6>
```

When an L2Net is changed from inbound to outbound, the PPP defaults can be configured on that L2Net. You can get to PPP configuration prompt by using the **encapsulator** command. In this example, the router is configured to send the name rtr-1 when prompted. This L2Net is meant to receive its IP address from the NT box. This will be sent during IPCP negotiations, and the router needs to be configured to ask the NT box for its IP address. This can be done using the **set ipcp** command and answering yes to request an IP address.

Table 129. Configure L2net To Send Name and Enable Interface To Receive IP Address Via IPCP

```
VPNRTR2 Config>NETWORK 6
Session configuration
VPNRTR2 L2T config:   6>ENCAPSULATOR
Point-to-Point user configuration
VPNRTR2 PPP 6 Config>SET NAME
Enter Local Name:  []? rtr-1
Password:rtr-1           1
Enter password again:rtr-1
PPP Local Name = rtr-1


VPNRTR2 PPP 6 Config>
VPNRTR2 PPP 6 Config>SET IPCP
IP COMPRESSION [no]:
Request an IP address [no]: yes      2
Interface remote IP address to offer if requested (0.0.0.0 for none) [0.0.0.0]?

VPNRTR2 PPP 6 Config>EXIT
VPNRTR2 L2T config:   6>EXIT
VPNRTR2 Config>
```

1. Password will not appear on screen. This is shown for illustration purposes only. This name and password must match the User and Password set up in the NT Remote Access Server under the User Manager function.
2. Answer yes to have the NT send an IP address for L2Net.

In order for the router L2net to receive an IP address from the NT tunnel endpoint, we must enable the IP address for dynamic IP. This will allow the NT to send an address from a pre-configured address pool via IPCP.

Table 130. Configure IP on the L2Net

```
VPNRTR2 Config>PROTOCOL IP
Internet protocol user configuration
VPNRTR2 IP config>ENABLE DYNAMIC-ADDRESS
Interface address []? 0.0.0.6   1
VPNRTR2 IP config>
```

1. Enter the IP address that was assigned by the **add l2-nets** command as shown in Table 125 on page 335.

A static route to the corporate subnet must be added since no dynamic routing protocols are used on the NT RAS host.

Table 131. Add a Static Route to the Private Network

```
VPNRTR2 IP config>ADD ROUTE
IP destination []? 192.168.141.64       1
Address mask [255.255.255.0]? 255.255.255.240
Via gateway 1 at []? 0.0.0.6
Cost [1]?
Via gateway 2 at []?
VPNRTR2 IP config>EXIT
VPNRTR2 Config>
```

1. This is the address and subnet mask of the network where the NT RAS is located.

Since the router is going to appear as a single user on the corporate network, and the corporate network does not have any knowledge of the branch network, we need to use Network Address and Port Translation (NAPT). NAPT is an enhancement to NAT, which was originally shipped in the IBM routers in V3.1. It is enabled and configured from the NAT feature.

Table 132. Configure NAT

```
VPNRTR2 Config>FEATURE NAT
Network Address Translation (NAT) user configuration
VPNRTR2 NAT config>ENABLE NAT

Complete! NAT set to ENABLED.
VPNRTR2 NAT config>
```

The next step is to define which address we want the packets translated to. This is done using the **reserve** command. When it asks you if the address will be obtained via IPCP, the answer is yes. The interface is 6, the L2Net. The router then asks for a pool name. This is used as a reference when we define which addresses should be translated. The router also tells you that you need to configure IP packet filters to pass the packets to the NAT feature to be translated. This is the first time the router reminds you to configure IP packet filters.

Table 133. Define How the LAN Addresses Should Be Translated

```
VPNRTR2 NAT config>RESERVE
Dynamically allocate address via IPCP? [No]: yes
Network number to get dynamic address. [0]? 6
Reserve Pool name.................... []? dyn-nat

Complete! NAT Reserve Pool defined.

NOTE: The associated TRANSLATE RANGE for this RESERVE POOL
      must still be configured.
      It must have a pool name of: dyn-nat

NOTE: You must have a corresponding INBOUND IP Access Control rule
      applied to your designated NAT interface.
      The rule should include the following information:
          Type=IN (include + NAT)
          DESTINATION_Addr=0.0.0.0
          DESTINATION_Mask=0.0.0.0

VPNRTR2 NAT config>
```

The next step is to define which addresses should be translated. The command is saying, ″Translate all packets with a source address in the 9.24.106.0 network to have an address in the dyn-nat pool.″ In the previous step, we defined that the dyn-nat pool is the address received by IPCP on interface 6. The router reminds you that you need to configure filters get the packets passed to NAT/NAPT for translation.

Table 134. Define Which LAN Addresses Should Be Translated

```
VPNRTR2 NAT config>TRANSLATE
Base (private) IP address to translate [0.0.0.0]? 9.24.106.0
Translate Range mask.................. [255.255.255.0]?
Associated Reserve Pool name.......... [dyn-nat]?

Complete! NAT Translate Range defined.

NOTE: The associated RESERVE POOL for this TRANSLATE RANGE has been found.

NOTE: You must have a corresponding OUTBOUND IP Access Control rule
      applied to your designated NAT interface.
      The rule should include the following information:
          Type=IN (include + NAT)
          SOURCE_Addr=9.24.106.0
          SOURCE_Mask=255.255.255.0
VPNRTR2 NAT config>EXIT
VPNRTR2 Config>
```

We now need to create the IP packet filters. Table 135 on page 339 shows that access control is enabled and then the filters attached to the L2Net are created and named out-6 and in-6.

Table 135. Add Packet Filters

```
VPNRTR2 Config>PROTOCOL IP
Internet protocol user configuration
VPNRTR2 IP config>SET ACCESS-CONTROL ON
VPNRTR2 IP config>

VPNRTR2 IP config>ADD PACKET-FILTER
Packet-filter name []? out-6
Filter incoming or outgoing traffic? [IN]? out
Which interface is this filter for [0]? 6

VPNRTR2 IP config>ADD PACKET-FILTER
Packet-filter name []? in-6
Filter incoming or outgoing traffic? [IN]?
Which interface is this filter for [0]? 6
VPNRTR2 IP config>
```

Once the packet filters are created, use the **update packet-filter** command to define the filters. The purpose of the out-6 filter is to direct all packets that are from subnet 9.24.106.0 and destined for the Internet to the Network Address Translation function.

Table 136. Update the Outbound Packet Filter

```
VPNRTR2 IP config>UPDATE PACKET-FILTER
Packet-filter name []? out-6
VPNRTR2 Packet-filter 'out-6' Config>ADD ACCESS-CONTROL
Access Control type [E]? N      1
Internet source [0.0.0.0]? 9.24.106.0
Source mask [255.255.255.0]?
Internet destination [0.0.0.0]?
Destination mask [0.0.0.0]?
Starting protocol number ([0] for all protocols) [0]?
Starting DESTINATION port number ([0] for all ports) [0]?
Starting SOURCE port number ([0] for all ports) [0]?
Filter on ICMP Type ([-1] for all types) [-1]?
TOS/Precedence filter mask (00-FF - [0] for none) [0]?
TOS/Precedence modification mask (00-FF - [0] for none) [0]?
Enable logging? [No]:
VPNRTR2 Packet-filter 'out-6' Config>exit
```

1. Type N specifies that the datagram should be sent to the NAT function.

The purpose of the in-6 filter is to direct all packets that are from the Internet and destined for subnet 9.24.106.0 to the Network Address Translation function.

Table 137. Update the Inbound Packet Filter

```
VPNRTR2 IP config>UPDATE PACKET-FILTER
Packet-filter name []? in-6
VPNRTR2 Packet-filter 'in-6' Config>ADD ACCESS-CONTROL
Access Control type [E]? N
Internet source [0.0.0.0]?
Source mask [0.0.0.0]?
Internet destination [0.0.0.0]?
Destination mask [0.0.0.0]?
Starting protocol number ([0] for all protocols) [0]?
Starting DESTINATION port number ([0] for all ports) [0]?
Starting SOURCE port number ([0] for all ports) [0]?
Filter on ICMP Type ([-1] for all types) [-1]?
TOS/Precedence filter mask (00-FF - [0] for none) [0]?
TOS/Precedence modification mask (00-FF - [0] for none) [0]?
Enable logging? [No]:
VPNRTR2 Packet-filter 'in-6' Config>exit
VPNRTR2 IP config>EXIT
VPNRTR2 Config>
```

This completes the configuration of the branch router. As always, it is a good idea to save the configuration to either the configuration program or a TFTP server.

Configure NT Remote Access Server

To configure the NT Remote Access Server, follow these steps:

- IP on the Internet accessible token ring = 192.168.141.34 / 255.255.255.240
- IP on ethernet = 192.168.141.65 / 255.255.255.240
- Add PPTP protocol with minimum of 1 VPN interface
- In remote access service, add a RAS device
- Link the VPN interface to the RAS server
- Configure IP pool of 192.168.141.70—192.168.141.73

Add an NT user name **rtr-1** and password of **rtr-1**. This must match the values configured in Table 129 on page 336. Disable the ″change password on first logon″ option, and set the password to never age.

The NT box must be reachable via the IP cloud. To prevent malicious activity, you can enable PPTP filtering on the interface which is connected to the IP cloud. This means that the PPTP server only accepts PPTP packets from authenticated users. The user, (the remote router in our example), is defined using the ″managing users″ function within NT. All non-PPTP packets or PPTP traffic from non-authenticated users will be dropped.

Refer to the following Microsoft Web page for information on setting up the PPTP server:
`http://www.microsoft.com/NTServer/commserv/deployment/planguides/installing_pptp.asp`

Monitoring/Troubleshooting the Configuration

Use ELS to dynamically monitor the PPTP tunnel. Configure ELS to display only subsystem L2 by issuing the **NODISPLAY SUBSYSTEM ALL** command followed by the **DISPLAY SUBSYSTEM L2 ALL ALL** command. Then issue the **TALK 2** command as shown in Table 138 on page 341.

Notice that there will be ″keepalive″ type traffic every 30 seconds even if there is no other traffic.

Table 138. ELS Output for L2 Subsystem

```
VPNRTR2 *TALK 2

40:19:49   L2.024: PPTP PAYLOAD SEND 38 bytes, net=6, callid=55253
40:19:49   L2.041: SND PPTP:F=3081,L=54,Tid=0,Cid=0,NS=121,NR=122,O=0
40:19:49   L2.040: RCV PPTP:F=3081,L=38,Tid=20169,Cid=55253,NS=123,NR=121,O=0
40:19:49   L2.022: PPTP PAYLOAD RCVD 38 bytes, net 6, callid=55253
40:19:59   L2.024: PPTP PAYLOAD SEND 38 bytes, net=6, callid=55253
40:19:59   L2.041: SND PPTP:F=3081,L=54,Tid=0,Cid=0,NS=122,NR=123,O=0
40:19:59   L2.040: RCV PPTP:F=3081,L=38,Tid=20169,Cid=55253,NS=124,NR=122,O=0
40:19:59   L2.022: PPTP PAYLOAD RCVD 38 bytes, net 6, callid=55253
```

For this example, issuing the **INTERFACE** command at the Talk 5 Protocol IP prompt shows that the PPP/4 interface has been assigned an IP address on the subnet at the other end of the tunnel. Remember that we configured the NT RAS to assign IP addresses from a pool starting at 192.168.141.70 and ending at 192.168.141.73.

The NT RAS host took .70 for its tunnel endpoint address and assigned .71 to the Network Utility at the other tunnel endpoint.

Table 139. Display the Interface Information

```
VPNRTR2 *TALK 5

CGW Operator Console
VPNRTR2 + PROTOCOL IP
VPNRTR2 IP>INTERFACE
Interface    MTU   IP Address(es)   Mask(s)         Address-MTU
  PPP/0     2044   192.168.141.17   255.255.255.240 Unspecified
  TKR/0     4082   9.24.106.8       255.255.255.0   Unspecified
  PPP/4     1500   192.168.141.71   255.255.255.255 Unspecified
VPNRTR2 IP>EXIT
```

Use the **call state** and **call statistics** commands at the FEATURE Layer-2-Tunneling prompt as shown in Table 140 to verify tunnel activity.

Table 140. Display Tunnel Status and Statistics

```
VPNRTR2 +FEATURE Layer-2-Tunneling
Layer-2-Tunneling Information
VPNRTR2 Layer-2-Tunneling Console> CALL STATE
CallID | Serial # | Net # |     State     | Time Since Chg | PeerID | TunnelID
 64985 |        0 |     6 | Established   |     0:13:46    |      0 |    19704

VPNRTR2 Layer-2-Tunneling Console> CALL STATISTICS
CallID | Serial # | Tx Pkts | Tx Bytes | Rx Pkts | Rx Bytes |  RTT  |  ATO
 64985 |        0 |      95 |     3440 |      97 |     3415 |     0 |
0
VPNRTR2 Layer-2-Tunneling Console>
```

Note: At the >FEATURE Layer-2-Tunneling, you can use the following sequence of commands: =>T5, =>NET 6, =>LIST ALL.

IBM Network Utility Initiated Voluntary L2TP Tunnel

Follow the steps in IBM Network Utility Initiated Voluntary PPTP Tunnel with the following exceptions:

- Enable L2TP
- Specify L2TP in the tunnel profile

  Note: Prompts are slightly different in this step (See Table 141).

Table 141. Specify L2TP in the Tunnel Profile

```
add tunnel-profile
Enter name: [ ]? L2TP peer
Tunneling Protocol? (PPTP, L2F, L2TP): [L2TP]
Enter local host name: [ ] netU
Set shared secret? (Yes, No): [No] y
Shared secret for tunnel authentication: * will not appear
Enter again to verify:
Tunnel-Server endpoint address: [0.0.0.0] 192.168.141.34

      Tunnel name: L2TP peer
      Tunn Type 3: L2TP
      Endpoint: 192.168.141.34
      Local Hostname: netU
Tunnel 'NT' has been added
```

L2TP Tunnel Terminated at an IBM Network Utility LNS

The L2TP sample scenario will establish the connection between a remote dial-up user in the branch and Network Utility in the corporate using L2TP tunneling. Refer to the sample network diagram in Figure 78 on page 343.

Connecting Dial-in Remote Users

Another application of VPN is to connect remote dial-in users to a central site over a public IP network, such as the Internet. The remote access server can be administrated by an ISP or by the user’s company. This scenario demonstrates how to use the IBM Nways 2210/Network Utility as Remote LAN Access (RLAN) servers using the L2TP and Dial In Access to LANs (DIALs) features of the IBM Nways 2210/Network Utility.

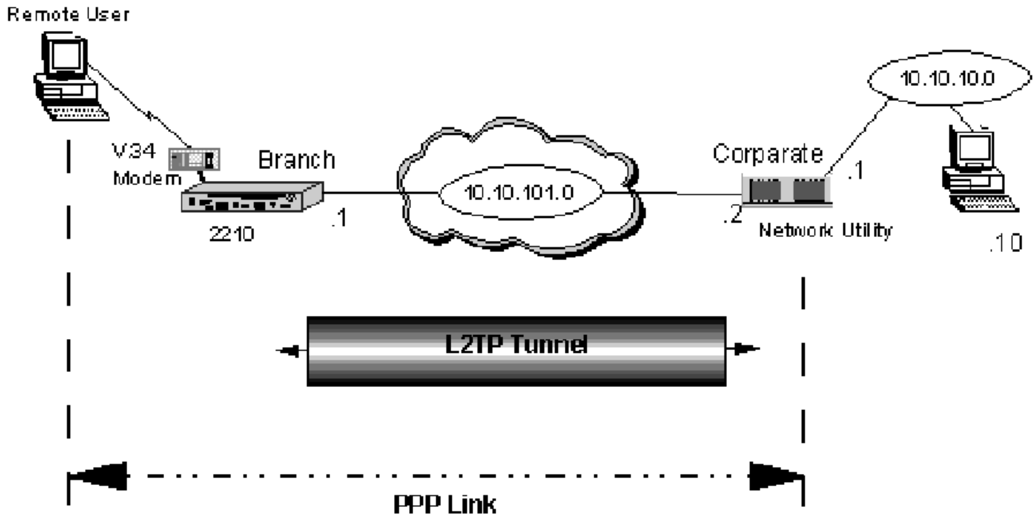


Figure 78. L2TP Sample Configuration

The IBM Nways 2210/Network Utility was used in this example, and the 2210 in the branch provides an RLAN access server for the remote dial-in users. An L2TP tunnel is set up between the branch router and the Network Utility in the data center so that the remote users can use the RLAN function in the Network Utility to access resources on the corporate intranet. Since the L2TP connection is IP-based, this traffic can also be sent through the IPSec tunnel if the IPSec is configured as well. With this alternative, L2TP is inside the IPSec tunnel.

Configuring the Branch Router For DIAL-in Access Server

The branch router 2210 was configured to allow a remote user to access the branch router via a V.34 dial-up modem and then extend the remote user's sessions to the corporate data center location over an IP network, such as the Internet, by using L2TP to tunnel the PPP session from the branch-office 2210 to the central-site IBM Network Utility.

Notes:

1. The use of V.34 for the remote user access is demonstrated in this scenario. However, the 2210 supports V.34, ISDN BRI, and V.25bis. V.34 is supported via external modems connected to WAN ports or via the 4- or 8-port Dial Access Adapters that provide integrated V.34 modems.
2. The IP network could be any IP-based network, such as the Internet or a public frame-relay network. In this scenario, the IP network is represented by a PPP serial WAN link.

The first step in the RLAN configuration is to add a V.34 interface. This is shown in Table 142 on page 344.

Table 142. Adding a V.34 Address and Configuring the V.34 Interface

```
Branch *t 6
Gateway user configuration
Branch Config>add V34-ADDRESS
Assign address name [1-23] chars []? local
Assign network dial address [1-30 digits] []? 9193013461
Branch Config>set data v34
Interface Number [0]? 4
Branch Config>net 4
V.34 Data Link Configuration
Branch V.34 System Net Config 4>set local-address
Local network address name []? local
```

You must map the V.34 port to the V.34 address. You can also set the modem initialization string and speed, but this example uses the default parameters. You can check the parameters you configured with the **’list all’**command as shown in Table 143.

Table 143. Listing the Configuration of the V.34 Port

```
Branch V.34 System Net Config   4>LIST all

         V.34 System Net Configuration:

Local Network Address Name    = local
Local Network Address         = 9193013461

Non-Responding addresses:
Retries                       = 1
Timeout                       = 0 seconds

Mode                          = Switched

Call timeouts:
Command Delay                 = 0 ms
Connect                       = 60 seconds
Disconnect                    = 2 seconds

Modem strings:
Initialization string         =

Speed (bps)                   = 115200
```

The next step is to create the virtual interfaces used for dial-in connections. RLAN users use a special kind of dial circuit called a ’dial-in circuit’. In this scenario, one virtual interface is created for our single RLAN test user. However, you can create many more virtual interfaces. The practical limit is the number of async ports available on the router.

The dial-in interfaces are added from the **talk 6 Config>** prompt as shown in .

Table 144. Creating the Virtual Dial-in Interfaces

```
Branch Config>ADD DEVICE DIAL-IN
Enter the number of PPP Dial-in Circuit interfaces [1]?
Adding device as interface 6
Base net for this circuit [0]? 4
Enable as a Multilink PPP link? [no]
Disabled as a Multilink PPP link.
Defaulting Data-link protocol to PPP
Add more dial circuit interface(s)?(Yes or [No]):
Use "net " command to configure circuit parameters

Branch Config>LIST DEVICES
Ifc 0     Ethernet            CSR  81600, CSR2  80C00, vector 94
Ifc 1     WAN PPP             CSR  81620, CSR2  80D00, vector 93
Ifc 2     WAN PPP             CSR  81640, CSR2  80E00, vector 92
Ifc 3     WAN PPP             CSR 381620, CSR2 380D00, vector 125
Ifc 4     V.34 Base Net       CSR 381640, CSR2 380E00, vector 124
Ifc 5     Token Ring          CSR 6000000, vector 95
Ifc 6     PPP Dial-in Circuit

Branch Config>NETWORK 6
Circuit configuration
Branch Dial-in Circuit config:   6>LIST all

Base net                       = 4
Circuit priority               = 8
```

Note: Only PPP is supported over V.34. However, with DIALs, multiple protocols (IP, IPX, NetBIOS, 802.2, and LLC) can be supported over the PPP connection.

For each dial-in circuit, there are a number of parameters you can configure; however, these can generally be left at their default values.

In order to route IP through the V.34 interface, an IP address must be assigned to the interface. When the client dials in, the router automatically adds a static route to its routing table that says the next hop for the remote user is the IP address of the V.34 virtual interface.

The address must be on a different subnet from the destination LAN segment. You can use a real IP address or use an unnumbered IP. For the unnumbered IP, the format of the address is 0.0.0.n, where n is the interface number. Table 145 shows the dialog for this scenario. Interface 6 is the virtual interface for the test dial-in user.

Table 145. Configuring IP Addresses on the Virtual Interfaces

```
Branch IP config>LIST ADDRESSES
IP addresses for each interface:
   intf     0                               IP disabled on this interface
   intf     1  10.10.101.1 255.255.255.0    Local wire broadcast, fill 1
   intf     2                               IP disabled on this interface
   intf     3                               IP disabled on this interface
   intf     4                               IP disabled on this interface
   intf     5  10.10.1.1   255.255.255.0    Local wire broadcast, fill 1
   intf     6                               IP disabled on this interface

Branch IP config>add address
Which net is this address for [0]? 6
New address []? 0.0.0.6
Address mask [0.0.0.0]? 255.255.255.0
```

ARP-subnet routing must be enabled in order to allow the router to respond to ARPs when the next hop to the destination is over a different interface from the interface that is receiving the ARP request. This is the case with RLAN where the client IP address is on the same subnet as the router’ s LAN interface, but the next hop (the V.34 interface) is on a different subnet. ARP-subnet routing is enabled as shown in Table 146.

Table 146. Enabling ARP-Subnet Routing

```
Branch IP config>ENABLE ARP-SUBNET-ROUTING

Branch IP config>LIST ADDRESSES
IP addresses for each interface:
   intf     0                               IP disabled on this interface
   intf     1  10.10.101.1 255.255.255.0    Local wire broadcast, fill 1
   intf     2                               IP disabled on this interface
   intf     3                               IP disabled on this interface
   intf     4                               IP disabled on this interface
   intf     5  10.10.1.1   255.255.255.0    Local wire broadcast, fill 1
   intf     6  0.0.0.6     255.255.255.0    Local wire broadcast, fill 1
```

This completes the configuration of the branch router for the basic DIALs function. The router is now restarted to activate the changes as shown in Table 147.

Table 147. Restarting the Router

```
Branch config>
Branch *res
Are you sure you want to restart the gateway? (Yes or [No]): yes
```

Configuring L2TP in the Branch Router

For this example, the dial-in user’ s PPP connection can be extended by setting up an L2TP tunnel between the 2210 in the branch location and Network Utility in the data center. The end user should be able to use the RLAN function in the Network Utility to connect to resources in the data center.

L2TP is a mechanism that involves a tunnel between an L2TP Access Concentrator (LAC) and an L2TP Network Server (LNS). In this scenario, the 2210 in the branch will be configured as the LAC and the Network Utility will be configured as the LNS. The first step is to enable L2TP in the LAC. See xTable 148.

Table 148. Enabling L2TP in the LAC (Branch Router)

```
Branch Config> FEATURE Layer-2-Tunneling
Branch Layer-2-Tunneling Config>ENABLE L2TP

 Restart system for changes to take effect.
Branch Layer-2-Tunneling Config>EXIT
```

Next, a tunnel is created in the LAC. This is shown in Table 149 on page 347.

Table 149. Creating L2TP Tunnel in the LAC (Branch Router)

```
Branch Config>ADD TUNNEL-PROFILE
Enter name:  []? lns.org
Tunneling Protocol?  (PPTP, L2F, L2TP): [L2TP]
Enter local hostname: []? lac.org
set shared secret? (Yes, No): [No] y
Shared secret for tunnel authentication:
Enter again to verify:
Passwords do not match.
...try again
Shared secret for tunnel authentication:
Enter again to verify:
Tunnel-Server endpoint address: [0.0.0.0]? 10.10.101.2

       Tunnel name: lns.org
          TunnType: L2TP
          Endpoint: 10.10.101.2
    Local Hostname: lac.org

Tunnel 'lns.org' has been added

Branch Config>LIST TUNNEL-PROFILES
TunnType  Endpoint         Tunnel name              Hostname

L2TP      10.10.101.2      lns.org                  lac.org

1 TUNNEL record displayed.
```

The following notes pertain to the LAC tunnel configuration:

Tunnel name
: This name should match the hostname which is configured on the LNS (Network Utility).

Hostname
: This is the hostname of the LAC.

Tunnel-Server endpoint
: The IP address of the endpoint of the tunnel. This address has to be reachable from the LAC. It can be any interface address or an internal IP address on the Network Utility. Here the address of the interface which is the endpoint of the tunnel is used.

Shared secret
: This parameter must be set if authentication is to be used on the tunnel and the value here must match the value configured in the LNS. L2TP tunnel authentication is enabled by default.

In order to activate these changes, the router must be restarted.

Configuring L2TP in the Network Utility

The 2216 has been configured as an L2TP Network Server (LNS). First, L2TP was enabled in the LNS. This is shown in Table 150.

Table 150. Enabling L2TP in the LNS

```
Corp Config>FEATURE Layer-2-Tunneling
Corp Layer-2-Tunneling Config>ENABLE L2TP

 Restart system for changes to take effect.
Corp Layer-2-Tunneling Config>EXIT
```

Then the tunnel is created in the LNS, pointing to the IP address and the name of the LAC. This is shown in Table 151.

Table 151. Creating L2TP Tunnel in the LNS (Corp Network Utility)

```
Corp Config>ADD TUNNEL-PROFILE
Enter name:  []? lac.org
Tunneling Protocol?  (PPTP, L2F, L2TP): [L2TP]
Enter local hostname: []? lns.org
set shared secret? (Yes, No): [No] Y
Shared secret for tunnel authentication:
Enter again to verify:
Tunnel-Server endpoint address: [0.0.0.0]? 10.10.101.1

       Tunnel name: lac.org
          TunnType: L2TP
          Endpoint: 10.10.101.1
    Local Hostname: lns.org
Tunnel 'lac.org' has been added

Corp Config>LIST TUNNEL-PROFILES
TunnType  Endpoint         Tunnel name            Hostname

L2TP      10.10.101.1      lac.org                lns.org

1 TUNNEL record displayed.
```

Note: If you are using shared secrets, the key must match the one configured in the LAC.

You can modify the PPP parameters for the L2TP tunnel. However, these parameters will be negotiated between the LAC and the LNS. The LAC acts as a proxy for the client PC in the PPP negotiation. An authentication protocol must be enabled for the L2TP tunnel. The default PPP parameters on the LNS were used for this scenario.

Next, the virtual interfaces over which the PPP connections will be terminated were added. These are analogous to the dial-in interface that was added in the branch router when it was configured for the DIALs function. However, in this case, the users are coming in through an L2TP tunnel instead of a V.34 interface.

In the LNS, the virtual interfaces are added from the L2TP feature configuration prompt. (In the LAC, they were added from the talk 6 main prompt.) This is shown in Table 152.

Table 152. Adding the Virtual Interface

```
Corp Config>FEATURE Layer-2-Tunneling
Corp Layer-2-Tunneling Config>ADD L2-NETS
Additional L2 nets:  [0]? 2
Add unnumbered IP addresses for each L2 net? [Yes]:
Adding device as interface 6
Defaulting Data-link protocol to PPP
Adding device as interface 7
Defaulting Data-link protocol to PPP
Enable IPX on L2T interfaces?(Yes or [No]):
Enable transparent bridging on L2T interfaces?(Yes or [No]):
Bridge configuration was not changed.

Restart router for changes to take affect.
Corp Layer-2-Tunneling Config>EXIT
```

In order to route IP through the L2 nets, an IP address must be assigned to the interface. When the client establishes the PPP connection through the L2TP tunnel, the router automatically adds a static route to its routing table that says that the next hop for the remote user is the IP address of the L2TP virtual interface. The address must be on a different subnet from the destination LAN segment.

The IP addresses for these interfaces are added when you create the interfaces. By default, they are unnumbered IP addresses. The format of the address is 0.0.0.n where n is the interface number (for example, for interface 7, the unnumbered IP address would be 0.0.0.7).

Note: If you need to change the default IP address associated with an L2TP net, you can do so via the IP config prompt in talk 6. However, unnumbered IP addressing works very well for RLAN because users connect to an L2TP net arbitrarily and the particular IP address associated with an L2TP net is not very critical.

ARP-subnet routing must be enabled in order to allow the router to respond to ARPs when the next hop to the destination is over a different interface from the interface that is receiving the ARP request. This is the case with RLAN where the client IP address is on the same subnet as the router’ s LAN interface but the next hop (the L2TP virtual interface) is on a different subnet. ARP-subnet routing is enabled as shown in Table 153.

Table 153. Enabling ARP-Subnet Routing

```
Corp config>protocol ip
Corp IP config>ENABLE ARP-SUBNET-ROUTING
Corp IP config>EXIT
```

Next, the method for clients to obtain an IP address is defined. The DIALs server in the Network Utility needs to be configured as if the users were dialing in via ISDN or V.34 rather than tunneling in through an L2TP tunnel. DIALs users need to be assigned an IP address that is on the same subnet as the LAN interface to which they wish to connect. There are five methods available:

Client The IP address is configured on the client.

User ID
The IP address is configured as part of the User ID definition on the router and sent to the client when it is authenticated. In this case, the IP address is associated with a specific user.

Interface
The IP address is configured in the interface and sent to the client. Here, the IP address is associated with the interface instead of the User ID.

DHCP Proxy
The IP address will be provided by a DHCP server and the router acts as a DHCP proxy for the client.

IP Pool
The IP pooling allows you to set up a block of IP addresses that are stored in a pool. When a client connects and requests an IP address, the router retrieves an address from the pool.

The methods for the clients to obtain an IP address are configured from the global DIALs menu. The client, user ID, interface and IP Pool methods are enabled. The router will attempt to use the first method that is enabled (in the order listed). You can also define primary and secondary domain name servers whose addresses are passed to the client during the IPCP negotiations. This is shown in Table 154.

Table 154. Listing Methods to Obtain IP Adresses

```
Corp Config>FEATURE DIALs
Dial-in Access to LANs global configuration
Corp Config>FEATURE DIALs
Dial-in Access to LANs global configuration
Corp DIALs config>LIST IP-ADDRESS-ASSIGNMENT
DIALs client IP address assignment:
Client     :  Enabled
UserID     :  Enabled
Interface  :  Enabled
Pool       :  Enabled
DHCP Proxy :  Disabled
```

In this scenario, the IP address for DIALs users from an IP pool is allocated. This is shown in Table 155.

Table 155. Adding IP Pool for DIALs Users

```
Corp Config>FEATURE DIALs
Dial-in Access to LANs global configuration
Corp DIALs config>ADD IP-POOL
Base address []? 10.10.10.11
Number of addresses [1]? 20
Corp DIALs config>LIST IP-POOLS
Configured IP address pools:
     Base Address      Last Address        Number
     ------------      ------------        ------
      10.10.10.11       10.10.10.30            20
```

At this point, the tunnel is configured in both the LNS and LAC, and the DIALs feature is configured in the LNS. Now, the PPP users that will tunnel to the LNS need to be configured. There are two ways to configure the PPP users to be tunneled:

- **Rhelm-Based Tunneling:** Using this method, the user only needs to be defined at the LNS. The format, username@domain must be used, where domain is the hostname of the LNS. When the client dials into the LAC using the username@domain format (for example, Sharif@lns.org), the LAC will create a tunnel to the specified domain (lns.org), and the PPP connection will be tunneled to the desired destination. With this method, all users with the same domain name are tunneled to the same destination.
- **User-Based Tunneling:** With this method, the user’s profile needs to be configured at both the LAC and the LNS and does not use the username@domain format. In the LAC, you specify the end destination in the user’s profile. In the LNS, you configure a normal dial-up user.

Table 156 on page 351 shows the definition of a Rhelm-based user on the Network Utility in the data center.

Table 156. Adding a Rhelm-Based L2TP User

```
Corp Config>ADD PPP-USER
Enter name:  []? sharif@lns.org
Password:
Enter again to verify:
Allow inbound access for user? (Yes, No): [Yes]
Will user be tunneled? (Yes, No): [No]
Is this a 'DIALs' user? (Yes, No): [Yes]
Type of route?  (hostroute, netroute): [hostroute]
Number of days before account expires [0-1000] [0]?
Number of grace logins allowed after an expiration [0-100] [0]?
IP address: [0.0.0.0]?
Enter hostname: []?
Allow virtual connections? (Yes, No): [No]
Give user default time allotted ? (Yes, No): [Yes]
Enable callback for user? (Yes, No): [No]
Will user be able to dial-out ? (Yes, No): [No]
Set ECP encryption key for this user? (Yes, No): [No]
Disable user ? (Yes, No): [No]


     PPP user name: sharif@lns.org
   User IP address: Interface Default
     Netroute Mask: 255.255.255.255
          Hostname:       Virtual Conn: disabled
      Time alotted: Box Default
     Callback type: disabled
          Dial-out: disabled
        Encryption: disabled
            Status: enabled
    Login Attempts: 0
    Login Failures: 0
  Lockout Attempts: 0
    Account Expiry:    Password Expiry:
Is information correct? (Yes, No, Quit): [Yes]

User 'sharif@lns.org' has been added
```

For user-based tunneling, the ID is defined in both the LAC and LNS. Table 157 shows the definition of a user-based ID on the 2210 in the branch office. This user is set to be tunneled, and the router is notified to set up the L2TP tunnel when the user dials in. The destination IP address of the other tunnel endpoint is specified along with the hostname of the 2210 to use when creating the tunnel.

Table 157. Adding a User-Based Tunneling User in the 2210 (LAC)

```
Branch Config>ADD PPP-USER
Enter name:  []? shoma
Password:
Enter again to verify:
Allow inbound access for user? (Yes, No): [Yes]
Will user be tunneled? (Yes, No): [No] y
Tunneling Protocol?  (PPTP, L2F, L2TP): [L2TP]
Enter local hostname: []? lac.org
Tunnel-Server endpoint address: [0.0.0.0]? 10.10.101.2

     PPP user name: shoma
          TunnType: L2TP
          Endpoint: 10.10.101.2
    Local Hostname: lac.org

Is information correct? (Yes, No, Quit): [Yes] y

User 'shoma' has been added
```

Note: As soon as you specify that the user will be tunneled, the router knows to not ask you whether you want the DIALs function enabled for this user, what the IP address of the client should be, or any of the other parameters that you are prompted for when defining a DIALs user. This is because the DIALs function for this user is being provided by the Network Utility. The 2210 is merely providing a gateway service to the Network Utility.

Table 158 shows the definition of the same user-based ID on the 2216 in the data center. A normal DIALs user is defined here. This user is not a tunneled user since he is authenticated by the DIALs function by the time he is authenticated, the L2TP headers have all been stripped off and the packets are just normal PPP packets. This completes the configuration of the LNS.

Table 158. Adding a User-Based Tunneling User in the Network Utility (LNS)

```
Corp Config>ADD PPP-USER
Enter name:  []? shoma
Password:
Enter again to verify:
Allow inbound access for user? (Yes, No): [Yes]
Will user be tunneled? (Yes, No): [No]
Is this a 'DIALs' user? (Yes, No): [Yes]
Type of route?  (hostroute, netroute): [hostroute]
Number of days before account expires [0-1000] [0]?
Number of grace logins allowed after an expiration [0-100] [0]?
IP address: [0.0.0.0]?
Enter hostname: []?
Allow virtual connections? (Yes, No): [No]
Give user default time allotted ? (Yes, No): [Yes]
Enable callback for user? (Yes, No): [No]
Will user be able to dial-out ? (Yes, No): [No]
Set ECP encryption key for this user? (Yes, No): [No]
Disable user ? (Yes, No): [No]


     PPP user name: shoma
   User IP address: Interface Default
     Netroute Mask: 255.255.255.255
          Hostname:       Virtual Conn: disabled
      Time alotted: Box Default
     Callback type: disabled
          Dial-out: disabled
        Encryption: disabled
            Status: enabled
    Login Attempts: 0
    Login Failures: 0
  Lockout Attempts: 0
    Account Expiry:    Password Expiry:
Is information correct? (Yes, No, Quit): [Yes] y

User 'shoma' has been added
```

The Network Utility must be restarted in order to activate these changes.

Monitoring L2TP

Now that the configuration is in place, the L2TP and RLAN configurations can be tested. The L2TP can be tested by dialing in from the remote PC, first with the Rhelm-based user ID and then with the User-based ID.

IP connectivity can be tested using PING from the PC client to the Network Utility. L2TP can be monitored from ELS using disp sub l2 all. A sample talk 2 session from the Network Utility LNS is shown in Table 159.

Table 159. Monitoring L2TP from ELS

```
Corp *TALK 2
00:04:27   L2.052: Tunnel 7042/0 has 15 seconds to establish itself
00:04:27   L2.050: EVENT Rx-SCCRQ,tid=7042/0,state=Idle
00:04:27   L2.048: RCV l2tpGetHostname, tid=7042/0
00:04:27   L2.058: Peer TunnelID = 48802
00:04:27   L2.060: Peer Hostname = lac.org
00:04:27   L2.047: Tunnel 7042/48802 State Changed Idle -> Authorizing
00:04:27   L2.074: Upcall from AAA subsystem, request SUCCESS
00:04:27   L2.050: EVENT Continue-SCCRQ,tid=7042/48802,state=Authorizing
00:04:27   L2.048: RCV SCCRQ, tid=7042/48802
00:04:27   L2.058: Peer TunnelID = 48802
00:04:27   L2.060: Peer Hostname = lac.org
00:04:27   L2.058: Peer Rcv Window = 4
00:04:27   L2.058: Peer Challenge = 0
00:04:27   L2.049: SEND SCCRP, tid=7042/48802
00:04:27   L2.035: Tunnel Auth Create Challenge, Tid=7042/48802, Len=16
00:04:27   L2.035: Tunnel Auth Create Challenge Response, Tid=7042/48802,
Len=16
00:04:27   L2.044: Allocating UDP port 1026 for tunnelid=7042
00:04:27   L2.041: SND L2TP:F=C802,L=121,Tid=48802,Cid=0,NS=0,NR=1,O=0
00:04:27   L2.047: Tunnel 7042/48802 State Changed Authorizing -> Wait-ctl-cnn
00:04:27   L2.040: RCV L2TP:F=C800,L=42,Tid=7042,Cid=0,NS=1,NR=1,O=0
00:04:27   L2.050: EVENT Rx-SCCCN,tid=7042/48802,state=Wait-ctl-cnn
00:04:27   L2.048: RCV SCCCN, tid=7042/48802
00:04:27   L2.057: Processing Challenge Response from Peer 4.7.3.3
00:04:27   L2.039: NOTE:SCCCN: Tunnel Authenticated
00:04:27   L2.047: Tunnel 7042/48802 State Changed Wait-ctl-cnn -> Established
00:04:27   L2.040: RCV L2TP:F=C800,L=48,Tid=7042,Cid=0,NS=2,NR=1,O=0
00:04:27   L2.007: LNS Allocated L2 net 8
00:04:27   L2.020: RCV Inbound-Call-Request, callid=25642, net=8
00:04:27   L2.021: SEND Inbound-Call-Reply, callid=25642, net=8
00:04:27   L2.041: SND L2TP:F=C802,L=44,Tid=48802,Cid=1156,NS=1,NR=3,O=0
00:04:27   L2.013: L2TP Call 25642 State Changed Idle -> Wait Connect
00:04:27   L2.030: LNS Forcing LCP option ACFC
00:04:27   L2.039: NOTE:Proxy-LCP Callback received
00:04:27   L2.009: Call Rcv Proxy-Auth-Type AVP,attr=29,val=4,len=8,flag=8008
00:04:27   L2.009: Call Rcv SEQUENCING_REQUIRED AVP,attr=39,val=0,len=6,flag=800
00:04:27   L2.013: L2TP Call 25642 State Changed Wait Connect -> Established
00:04:27   L2.015: Call Established-LNS,net=8,speed=115200,flags=4802
00:04:27   L2.017: Using Proxy-LCP AUTH on net 8
00:04:27   L2.021: SEND Set-Link-Info, callid=25642, net=8
00:04:27   L2.041: SND L2TP:F=C802,L=36,Tid=48802,Cid=1156,NS=2,NR=4,O=0
00:04:27   L2.040: RCV L2TP:F=C800,L=12,Tid=7042,Cid=0,NS=4,NR=3,O=0
00:04:32   L2.022: L2TP PAYLOAD RCVD 53 bytes, net 8, callid=25642
00:04:32   L2.024: L2TP PAYLOAD SEND 6 bytes, net=8, callid=25642
00:04:32   L2.041: SND L2TP:F=6902,L=18,Tid=48802,Cid=1156,NS=1,NR=2,O=0
00:04:32   L2.024: L2TP PAYLOAD SEND 8 bytes, net=8, callid=25642
```

The L2TP tunnel state can be checked from talk 5 as shown in Table 160 on page 354.

Table 160. Monitoring L2TP from Talk 5

```
Branch *TALK 5

Branch +FEATURE Layer-2-Tunneling
Layer-2-Tunneling Information
Branch Layer-2-Tunneling Console> TUNNEL STATE
Tunnel ID | Type | Peer ID |    State    | Time Since Chg | # Calls | Flags
    35589 | L2TP |   58774 | Established |     0: 1:24    |       1 |  TL
 F
Branch Layer-2-Tunneling Console> TUNNEL STATISTICS
Tunnel ID | Type | Tx Pkts | Tx Bytes | Rx Pkts | Rx Bytes |  RTT  |  ATO
    35589 | L2TP |     108 |     7883 |     104 |     5388 |     5 |     5


Corp *TALK 5
Corp +FEATURE Layer-2-Tunneling
Layer-2-Tunneling Information
Corp Layer-2-Tunneling Console> TUNNEL STATE
Tunnel ID | Type | Peer ID |    State    | Time Since Chg | # Calls | Flags
    58774 | L2TP |   35589 | Established |     0: 2: 9    |       1 |  TL
 F
Corp Layer-2-Tunneling Console> TUNNEL STATISTICS
Tunnel ID | Type | Tx Pkts | Tx Bytes | Rx Pkts | Rx Bytes |  RTT  |  ATO
    58774 | L2TP |     108 |     5540 |     112 |     8035 |     5 |     5
```

This completes the configuration and monitoring of L2TP for Remote LAN Access using IBM 2210 and Network Utility.

Part 4. Appendixes

Appendix A. Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these 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 IBM's product, program, or service may be used. Subject to IBM's valid intellectual property, or other legally protectable rights, any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights 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 user's responsibility.

IBM may have patents or pending patent applications covering subject matter 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
North Castle Drive
Armonk, NY 10504-1785 U.S.A

Notice to Users of Online Versions of This Book

For online versions of this book, we authorize you 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.

Electronic Emission Notices

Federal Communications Commission (FCC) Class A Statement

Note: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user will be required to correct the interference at his own expense.

Properly shielded and grounded cables and connectors must be used in order to meet FCC emission limits. IBM is not responsible for any radio or television interference caused by using other than recommended cables and connectors or by unauthorized changes or modifications to this equipment. Unauthorized changes or modifications could void the user's authority to operate the equipment.

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

Industry Canada Class A Emission Compliance Statement

This Class A digital apparatus complies with Canadian ICES-003.

Avis de conformité aux normes d'Industrie Canada

Cet appareil numérique de la classe A est conform à la norme NMB-003 du Canada.

Japanese Voluntary Control Council for Interference (VCCI) Statement

This product is a Class A Information Technology Equipment and conforms to the standards set by the Voluntary Control Council for Interference by Technology Equipment (VCCI). In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

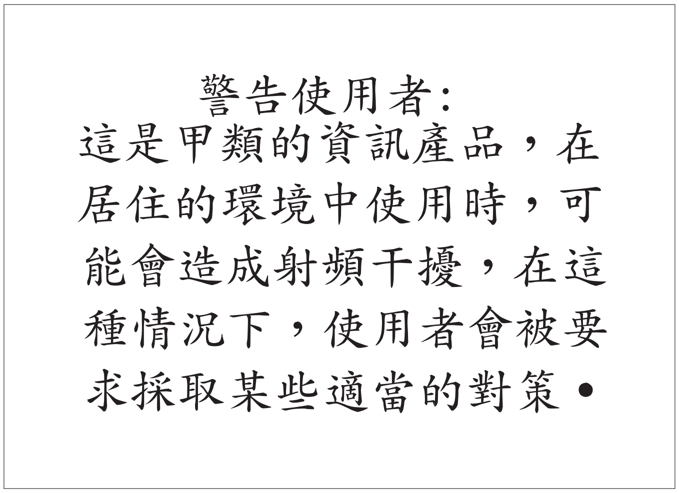
この装置は、情報処理装置等電波障害自主規制協議会（ＶＣＣＩ）の基準に基づくクラスＡ情報技術装置です。この装置を家庭環境で使用すると電波妨害を引き起こすことがあります。この場合には使用者が適切な対策を講ずるよう要求されることがあります。

CISPR22 Compliance Statement

This product has been tested and found to comply with the limits for Class A Information Technology Equipment according to CISPR 22/European Standard EN 55022. The limits for Class A equipment were derived for commercial and industrial environments to provide reasonable protection against interference with licensed communication equipment.

Warning: This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

Taiwanese Class A Warning Statement



警告使用者:
這是甲類的資訊產品，在
居住的環境中使用時，可
能會造成射頻干擾，在這
種情況下，使用者會被要
求採取某些適當的對策•

EMC Directive 89/336/EEC Statement

This product is in conformity with the protection requirements of EC Council Directive 89/336/EEC on the approximation of the laws of the Member States relating to electromagnetic compatibility. IBM cannot accept responsibility for any failure to satisfy the protection requirements resulting from a non-recommended modification of the product, including the fitting of non-IBM option cards.

The product (2216 Model 400) bears the Telecom CE mark (CE 168 X) for ISDN Basic Rate complying with I-CTR3 (Bridging measures) as per the European directive 91/263/EEC (TTE directive). The product bears the Telecom CE mark (CE 168 X) for: V.24/V.28,V36 and X.21 electrical interfaces complying with NET 1 and with NET 2 physical level. ISDN Basic Rate complying with I-CTR3 (Bridging measures) as per the European directive 91/263/EEC (TTE directive).

Warning: This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

Zulassungsbescheinigung laut dem Deutschen Gesetz über die elektromagnetische Verträglichkeit von Geräten (EMVG) vom 30. August 1995 (bzw. der EMC EG Richlinie 89/336)

Dieses Gerät ist berechtigt in Übereinstimmung mit dem Deutschen EMVG das EG-Konformitätszeichen - CE - zu führen.

Verantwortlich für die Konformitätserklärung nach Paragraph 5 des EMVG ist die IBM Deutschland Informationssysteme GmbH, 70548 Stuttgart.

Informationen in Hinsicht EMVG Paragraph 3 Abs. (2) 2:

Das Gerät erfüllt die Schutzanforderungen nach EN 50082-1 und EN 55022 Klasse A.

EN 55022 Klasse A Geräte müssen mit folgendem Warnhinweis versehen werden: “Warnung: dies ist eine Einrichtung der Klasse A. Diese Einrichtung kann im

Wohnbereich Funkstörungen verursachen; in diesem Fall kann vom Betreiber verlangt werden, angemessene Maßnahmen durchzuführen und dafür aufzukommen.”

EN 50082-1 Hinweis: “Wird dieses Gerät in einer industriellen Umgebung betrieben (wie in EN 50082-2 festgelegt), dann kann es dabei eventuell gestört werden. In solch einem Fall ist der Abstand bzw. die Abschirmung zu der industriellen Störquelle zu vergrößern.”

Anmerkung: Um die Einhaltung des EMVG sicherzustellen sind die Geräte, wie in den IBM Handbüchern angegeben, zu installieren und zu betreiben.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

AIX
eNetwork
ESCON
IBM
Microsoft
Nways
NetView
OS/2
Parallel Sysplex
Presentation Manger
VM/ESA

Tivoli is a trademark of Tivoli Systems Inc. United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and/or other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

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

Appendix B. Safety Information

Danger: Before you begin to install this product, read the safety information in *Caution: Safety Information—Read This First*, SD21-0030. This booklet describes safe procedures for cabling and plugging in electrical equipment.

Gevaar: Voordat u begint met de installatie van dit produkt, moet u eerst de veiligheidsinstructies lezen in de brochure *PAS OP! Veiligheidsinstructies—Lees dit eerst,* SD21-0030. Hierin wordt beschreven hoe u electrische apparatuur op een veilige manier moet bekabelen en aansluiten.

Danger: Avant de procéder à l'installation de ce produit, lisez d'abord les consignes de sécurité dans la brochure *ATTENTION: Consignes de sécurité—A lire au préalable,* SD21-0030. Cette brochure décrit les procédures pour câbler et connecter les appareils électriques en toute sécurité.

Perigo: Antes de começar a instalar este produto, leia as informações de segurança contidas em *Cuidado: Informações Sobre Segurança—Leia Isto Primeiro,* SD21-0030. Esse folheto descreve procedimentos de segurança para a instalação de cabos e conexões em equipamentos elétricos.

危險：安裝本產品之前， 請先閱讀 "Caution: Safety Information--Read This First" SD21-0030 手冊中所提供的安全注意事項。 這本手冊將會說明使用電器設備的纜線及電源的安全程序。

Opasnost: Prije nego sto pǒcnete sa instalacijom produkta, pročitajte naputak o pravilima o sigurnom rukovanju u Upozorenje: Pravila o sigurnom rukovanju - Prvo pročitaj ovo, SD21-0030. Ovaj privitak opisuje sigurnosne postupke za priključrivanje kabela i priključivanje na električno napajanje.

Upozornění: než zahájíte instalaci tohoto produktu, přečtěte si nejprve bezpečnostní informace v pokynech „Bezpečnostní informace" č. 21-0030. Tato brožurka popisuje bezpečnostní opatření pro kabeláž a zapojení elektrického zařízení.

Fare! Før du installerer dette produkt, skal du læse sikkerhedsforskrifterne i *NB: Sikkerhedsforskrifter—Læs dette først* SD21-0030. Vejledningen beskriver den fremgangsmåde, du skal bruge ved tilslutning af kabler og udstyr.

Gevaar Voordat u begint met het installeren van dit produkt, dient u eerst de veiligheidsrichtlijnen te lezen die zijn vermeld in de publikatie *Caution: Safety Information - Read This First* , SD21-0030. In dit boekje vindt u veilige procedures voor het aansluiten van elektrische appratuur.

VAARA: Ennen kuin aloitat tämän tuotteen asennuksen, lue julkaisussa *Varoitus: Turvaohjeet—Lue tämä ensin* , SD21-0030, olevat turvaohjeet. Tässä kirjasessa on ohjeet siitä, miten sähkölaitteet kaapeloidaan ja kytketään turvallisesti.

Danger : Avant d'installer le présent produit, consultez le livret *Attention : Informations pour la sécurité — Lisez-moi d'abord* SD21-0030, qui décrit les procédures à respecter pour effectuer les opérations de câblage et brancher les équipements électriques en toute sécurité.

Vorsicht: Bevor mit der Installation des Produktes begonnen wird, die Sicherheitshinweise in *Achtung: Sicherheitsinformationen—Bitte zuerst lesen,* IBM Form SD21-0030. Diese Veröffentlichung beschreibt die Sicherheitsvorkehrungen für das Verkabeln und Anschließen elektrischer Geräte.

Vigyázat: Mielôtt megkezdi a berendezés üzembe helyezését, olvassa el a *Caution: Safety Information— Read This First,* SD21-0030 könyvecskében leírt biztonsági információkat. Ez a könyv leírja, milyen biztonsági intézkedéseket kell megtenni az elektromos berendezés huzalozásakor illetve csatlakoztatásakor.

Pericolo: prima di iniziare l'installazione di questo prodotto, leggere le informazioni relative alla sicurezza riportate nell'opuscolo *Attenzione: Informazioni di sicurezza — Prime informazioni da leggere* in cui sono descritte le procedure per il cablaggio ed il collegamento di apparecchiature elettriche.

危険：　導入作業を開始する前に、安全に関する
小冊子SD21-0030　の「最初にお読みください」
(Read This First)の項をお読みください。
この小冊子は、電気機器の安全な配線と接続の
手順について説明しています。

위험: 이 제품을 설치하기 전에 반드시
"주의: 안전 정보-시작하기 전에"
(SD21-0030)   에 있는 안전 정보를
읽으십시오.

Fare: Før du begynner å installere dette produktet, må du lese sikkerhetsinformasjonen i *Advarsel: Sikkerhetsinformasjon — Les dette først* , SD21-0030 som beskriver sikkerhetsrutinene for kabling og tilkobling av elektrisk utstyr.

Uwaga:
Przed rozpoczęciem instalacji produktu należy zapoznać się z instrukcją:
"Caution: Safety Information - Read This First", SD21-0030.
Zawiera ona warunki bezpieczeństwa przy podłączaniu do sieci elektrycznej
i eksploatacji.

Perigo: Antes de iniciar a instalação deste produto, leia as informações de segurança *Cuidado: Informações de Segurança — Leia Primeiro* , SD21-0030. Este documento descreve como efectuar, de um modo seguro, as ligações eléctricas dos equipamentos.

ОСТОРОЖНО: Прежде чем инсталлировать этот продукт, прочтите Инструкцию по технике безопасности в документе "Внимание: Инструкция по технике безопасности -- Прочесть в первую очередь", SD21-0030. В этой брошюре описаны безопасные способы каблирования и подключения электрического оборудования.

Nebezpečenstvo: Pred inštaláciou výrobku si prečítajte
bezpečnosté predpisy v
Výstraha: Bezpeč osté predpisy - Prečítaj ako prvé,
SD21 0030. V tejto brožúrke sú opísané bezpečnosté
postupy pre pripojenie elektrických zariadení.

Pozor: Preden zaènete z instalacijo tega produkta preberite poglavje: ´Opozorilo: Informacije o varnem rokovanju-preberi pred uporabo," SD21-0030. To poglavje opisuje pravilne postopke za kabliranje,

Peligro: Antes de empezar a instalar este producto, lea la información de seguridad en *Atención: Información de Seguridad — Lea Esto Primero,* SD21-0030. Este documento describe los procedimientos de seguridad para cablear y enchufar equipos eléctricos.

Varning — livsfara: Innan du börjar installera den här produkten bör du läsa säkerhetsinformationen i dokumentet *Varning: Säkerhetsföreskrifter— Läs detta först,* SD21-0030. Där beskrivs hur du på ett säkert sätt ansluter elektrisk utrustning.

危險：

開始安裝此產品之前，請先閱讀安全資訊。

注意：

請先閱讀 - 安全資訊 SD21-0030

此冊子說明插接電器設備之電纜線的安全程序。

Index

Readers' Comments — We'd Like to Hear from You

Network Utility
Installation,
Getting Started,
and User's Guide

Publication No. GA27-4167-02

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
GA27-4167-02

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

Part Number: 31L3207

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

GA27-4167-02

31L3207