

TECHNICAL NOTE

MD8470A

Signalling Tester

Application Test

ANRITSU CORPORATION

Copyright © 2005 by ANRITSU CORPORATION

The contents of this manual shall not be disclosed in any way or reproduced in any media without the express written permission of Anritsu Corporation.

Detailed Explanations for Each Application

Discover What's Possible™
MD8470A-E-E-1

Slide 1

Anritsu

Contents

- Part 1: Packet Communication
(HTML and WAP)
- Part 2: Video Streaming
- Part 3: Video Telephony
- Part 4: SMS
- Part 5: MMS
- Part 6: DRM

Discover What's Possible™
MD8470A-E-E-1

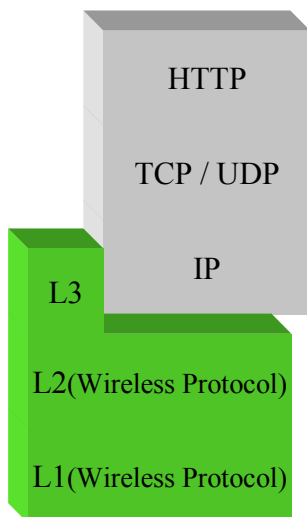
Slide 2

Anritsu

Part 1: Packet Communication (HTTP and WAP)

Packet Communication (HTTP)

Protocol Stack



HTTP : Hypertext Transfer Protocol

TCP : Transmission Control Protocol

UDP : User Datagram Protocol

IP : Internet Protocol

L2 (W-CDMA) : MAC / RLC

MAC : Media Access Control

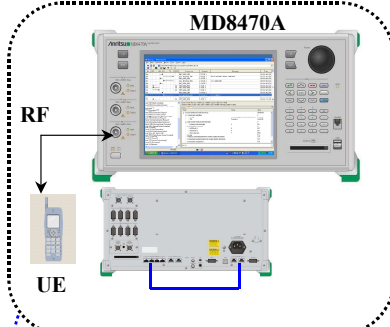
RLC : Radio Link Control

L2 (GSM/GPRS) : DL /

DL : Data Link

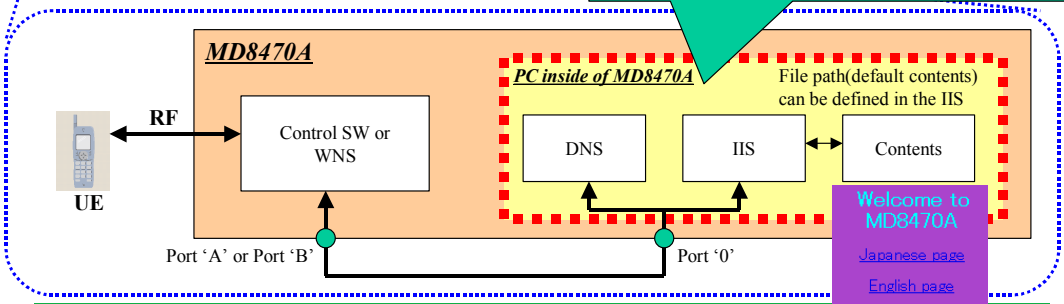
L1 (W / G) : PHY(Physical Layer)

Packet Communication (HTTP)

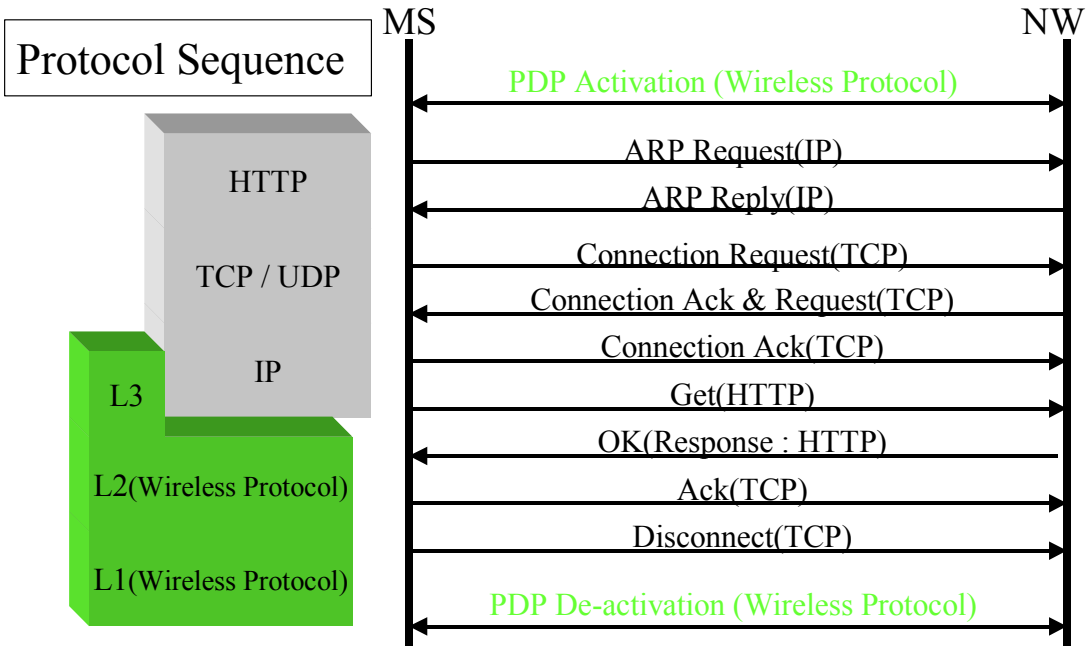


- ✓ **DNS (Domain Name System)**
DNS supports the conversion between a domain name and an IP address.
(Ex : vodafone.com <--> 192.168.1.2)
- ✓ **IIS (Internet Information Server)**
IIS supports the Web server function that sends the contents and the default contents (default web site).

All these parts are available in an external PC

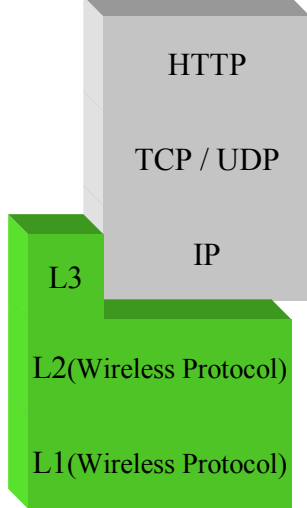


Packet Communication (HTTP)



Packet Communication (HTTP)

Protocol Sequence (ARP Request)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 1 | 0.000000 | 192.168.1.1 | Broadcast | ARP | who has 192.168.1.2? Tell 192.168.1.1 |
| 2 | 0.000025 | 192.168.1.2 | 192.168.1.1 | ARP | 192.168.1.2 is at 00:10:71:00:56:b9 |
| 3 | 0.009941 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [SYN] seq=0 Ack=0 win=3920 |
| 4 | 0.010002 | 192.168.1.2 | 192.168.1.1 | TCP | http > 6616 [SYN, ACK] seq=0 Ack=1 win= |
| 5 | 0.259874 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [ACK] seq=1 Ack=1 win=3920 |
| 6 | 0.320626 | 192.168.1.1 | 192.168.1.2 | HTTP | GET / HTTP/1.1 |
| 7 | 0.320296 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (text/html) |
| 8 | 0.959824 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [ACK] seq=518 Ack=924 win= |
| 9 | 5.959447 | 192.168.1.1 | 192.168.1.2 | TCP | [TCP Dup ACK #61] 6616 > http [ACK] se |
| 10 | 16.438704 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [RST] seq=518 Ack=40341723 |


```

Frame 1 (60 bytes on wire, 60 bytes captured)
Ethernet II, Src: 00:00:91:03:19:15, Dst: ff:ff:ff:ff:ff:ff
Address Resolution Protocol (request)
Hardware type: Ethernet (0x0001)
Protocol type: IP (0x0800)
Hardware size: 6
Protocol size: 4
opcode: request (0x0001)
Sender MAC address: 00:00:91:03:19:15 (192.168.1.1)
Sender IP address: 192.168.1.1 (192.168.1.1)
Target MAC address: 00:00:00:00:00:00 (00:00:00_00:00:00)
Target IP address: 192.168.1.2 (192.168.1.2)
    
```

*1 : MS(IP : 192.168.1.1) searches the MAC address of the Server(IP : 192.168.1.2)

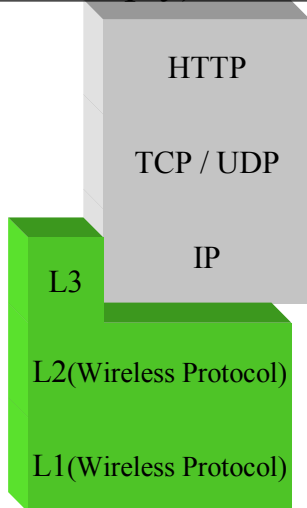
Discover What's Possible™
MD8470A-E-E-1

Slide 7

Anritsu

Packet Communication (HTTP)

Protocol Sequence (ARP Reply)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 1 | 0.000000 | 192.168.1.1 | Broadcast | ARP | who has 192.168.1.2? Tell 192.168.1.1 |
| 2 | 0.000025 | 192.168.1.2 | 192.168.1.1 | ARP | 192.168.1.2 is at 00:10:71:00:56:b9 |
| 3 | 0.009941 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [SYN] seq=0 Ack=0 win=3920 |
| 4 | 0.010002 | 192.168.1.2 | 192.168.1.1 | TCP | http > 6616 [SYN, ACK] seq=0 Ack=1 win= |
| 5 | 0.259874 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [ACK] seq=1 Ack=1 win=3920 |
| 6 | 0.320626 | 192.168.1.1 | 192.168.1.2 | HTTP | GET / HTTP/1.1 |
| 7 | 0.320296 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (text/html) |
| 8 | 0.959824 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [ACK] seq=518 Ack=924 win= |
| 9 | 5.959447 | 192.168.1.1 | 192.168.1.2 | TCP | [TCP Dup ACK #61] 6616 > http [ACK] se |
| 10 | 16.438704 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [RST] seq=518 Ack=40341723 |


```

Frame 2 (42 bytes on wire, 42 bytes captured)
Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:91:03:19:15
Address Resolution Protocol (reply)
Hardware type: Ethernet (0x0001)
Protocol type: IP (0x0800)
Hardware size: 6
Protocol size: 4
opcode: reply (0x0002)
Sender MAC address: 00:10:71:00:56:b9 (192.168.1.2)
Sender IP address: 192.168.1.2 (192.168.1.2)
Target MAC address: 00:00:91:03:19:15 (192.168.1.1)
Target IP address: 192.168.1.1 (192.168.1.1)
    
```

*1 : Server (IP : 192.168.1.2) replies the MAC address to MS(IP : 192.168.1.1)

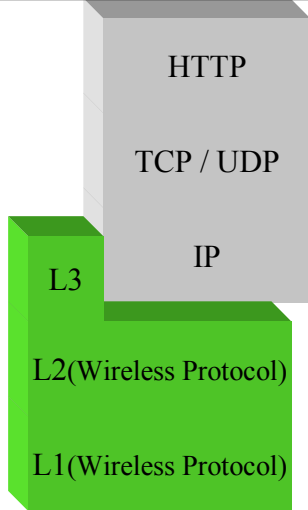
Discover What's Possible™
MD8470A-E-E-1

Slide 8

Anritsu

Packet Communication (HTTP)

Protocol Sequence (Connection Req.)



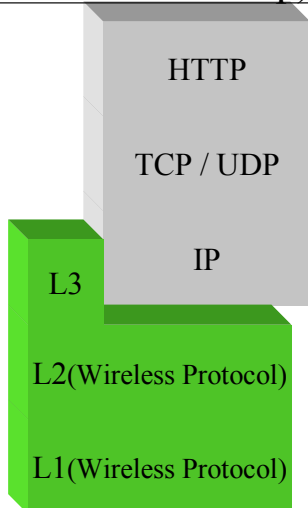
| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 1 | 0.000000 | 192.168.1.1 | Broadcast | ARP | who has 192.168.1.2? Tell 192.168.1.1 |
| 2 | 0.000025 | 192.168.1.2 | 192.168.1.1 | ARP | 192.168.1.2 is at 00:10:71:00:56:b9 |
| 3 | 0.009941 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [SYN] Seq=0 Win=3920 |
| 4 | 0.010002 | 192.168.1.2 | 192.168.1.1 | TCP | http > 6616 [ACK] Seq=1 Ack=1 Win=0 Len=0 *1,2 |
| 5 | 0.259874 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [ACK] Seq=1 Ack=1 Win=3920 |
| 6 | 0.320626 | 192.168.1.1 | 192.168.1.2 | HTTP | GET / HTTP/1.1 |
| 7 | 0.322096 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (text/html) |
| 8 | 0.959824 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [ACK] Seq=518 Ack=924 Win=0 Len=0 |
| 9 | 5.959447 | 192.168.1.1 | 192.168.1.2 | TCP | [TCP Dup ACK #6] 6616 > http [ACK] Seq=518 Ack=40341723 |
| 10 | 16.438704 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [RST] Seq=518 Ack=40341723 |

*1 : data from '6616(port number of MS)' to '80(http port number of the Server)'

*2 : connection request from MS to server

Packet Communication (HTTP)

Protocol Sequence (Conn.Ack & Req.)



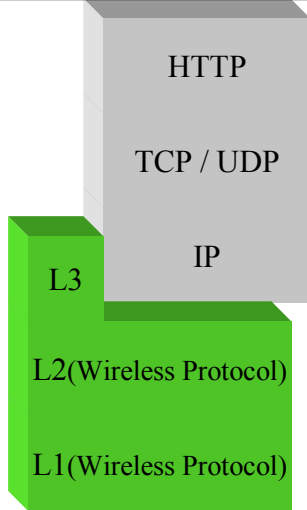
| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 1 | 0.000000 | 192.168.1.1 | Broadcast | ARP | who has 192.168.1.2? Tell 192.168.1.1 |
| 2 | 0.000025 | 192.168.1.2 | 192.168.1.1 | ARP | 192.168.1.2 is at 00:10:71:00:56:b9 |
| 3 | 0.009941 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [SYN] Seq=0 Ack=0 Win=3920 |
| 4 | 0.010002 | 192.168.1.2 | 192.168.1.1 | TCP | http > 6616 [SYN, ACK] Seq=0 Ack=1 Win=0 Len=0 *1,2 |
| 5 | 0.259874 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [ACK] Seq=1 Ack=1 Win=3920 |
| 6 | 0.320626 | 192.168.1.1 | 192.168.1.2 | HTTP | GET / HTTP/1.1 |
| 7 | 0.322096 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (text/html) *1,2 |
| 8 | 0.959824 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [ACK] Seq=518 Ack=924 Win=0 Len=0 |
| 9 | 5.959447 | 192.168.1.1 | 192.168.1.2 | TCP | [TCP Dup ACK #6] 6616 > http [ACK] Seq=518 Ack=40341723 |
| 10 | 16.438704 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [RST] Seq=518 Ack=40341723 |

*1 : data from '80(http port number of the Server)' to '6616(port number of MS)'

*2 : Ack of 'connection request' and connection request from server to MS.

Packet Communication (HTTP)

Protocol Sequence (Connection Ack)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 1 | 0.000000 | 192.168.1.1 | Broadcast | ARP | who has 192.168.1.2? Tell 192.168.1.1 |
| 2 | 0.000025 | 192.168.1.2 | 192.168.1.1 | ARP | 192.168.1.2 is at 00:10:71:00:56:b9 |
| 3 | 0.009941 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [SYN] Seq=0 Ack=0 Win=3920 |
| 4 | 0.010002 | 192.168.1.2 | 192.168.1.1 | TCP | http > 6616 [SYN, ACK] Seq=0 Ack=1 Win=3920 |
| 5 | 0.320926 | 192.168.1.1 | 192.168.1.2 | HTTP | GET / HTTP/1.1 *1 |
| 7 | 0.322096 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (text/html) |
| 8 | 0.959824 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [ACK] Seq=518 Ack=921 Win= |
| 9 | 5.959447 | 192.168.1.1 | 192.168.1.2 | TCP | [TCP dup ACK 981] 6616 > http [ACK] Seq= |
| 10 | 16.438704 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [RST] Seq=518 Ack=40341723 |


```

Frame 5 (60 bytes on wire (60 bytes captured)
Ethernet II, Src: 00:00:91:03:19:15, Dst: 00:10:71:00:56:b9
Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
Transmission Control Protocol, Src Port: 6616 (6616), Dst Port: http (80), Seq: 1, Ack: 1, Len: 0
  source port: 6616 (6616)
  destination port: http (80)
  sequence number: 1 (relative sequence number)
  acknowledgement number: 1 (relative ack number)
  header length: 20 bytes
  flags: 0x0010 (ACK) *2
  window size: 39200
  checksum: 0x04e3 (correct)
  [SEQ/ACK analysis]
    
```

*1 : data from '6616(port number of MS)' to '80(http port number of the Server)'

*2 : Ack of 'connection request'

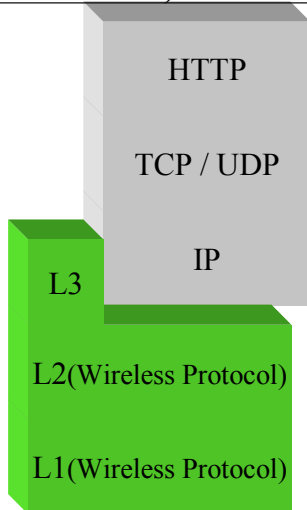
Discover What's Possible™
MD8470A-E-E-1

Slide 11

Anritsu

Packet Communication (HTTP)

Protocol Sequence (HTTP Get)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 1 | 0.000000 | 192.168.1.1 | Broadcast | ARP | who has 192.168.1.2? Tell 192.168.1.1 |
| 2 | 0.000025 | 192.168.1.2 | 192.168.1.1 | ARP | 192.168.1.2 is at 00:10:71:00:56:b9 |
| 3 | 0.009941 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [SYN] Seq=0 Ack=0 Win=3920 |
| 4 | 0.010002 | 192.168.1.2 | 192.168.1.1 | TCP | http > 6616 [SYN, ACK] Seq=0 Ack=1 Win=3920 |
| 5 | 0.259874 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [ACK] Seq=1 Ack=1 Win=3920 |
| 6 | 0.320926 | 192.168.1.1 | 192.168.1.2 | HTTP | GET / HTTP/1.1 *1 |
| 7 | 0.322096 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (text/html) *1 |
| 8 | 0.959824 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [ACK] Seq=518 Ack=921 Win= |
| 9 | 5.959447 | 192.168.1.1 | 192.168.1.2 | TCP | [TCP dup ACK 981] 6616 > http [ACK] Seq= |
| 10 | 16.438704 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [RST] Seq=518 Ack=40341723 |


```

Transmission Control Protocol, Src Port: 6616 (6616), Dst Port: http (80), Seq: 1, Ack: 1, Len: 51
Hypertext Transfer Protocol
  GET / HTTP/1.1\r\n *1
  Request Method: GET
  Request URI: /
  Request Version: HTTP/1.1
  Connection: keep-alive\r\n
  Host: 192.168.1.2\r\n *2
  User-Agent: Mozilla/5.0(SMB3;Z105)/Samsung\r\n
  Accept: application/vnd.wap.wmlc, application/vnd.wap.wmlscriptcc, image/vnd.wap.wbmp, image/png,
  Accept-Language: en\r\n
  Accept-Charset: utf-8\r\n
  Profile: http://wap.samsungmobile.com/uaprof/z105UAPProf.rdf\r\n
  X-VODAFONE-3GPPContext: yes\r\n
  \r\n
    
```

*1 : request the html contents. ' / ' means the relative path. Only ' / ' case is root contents of the server.

*2 :
Host : Server IP address
Application : supported MIME type
Language : en (English)
Charaset : utf-8(one of the common unicode)

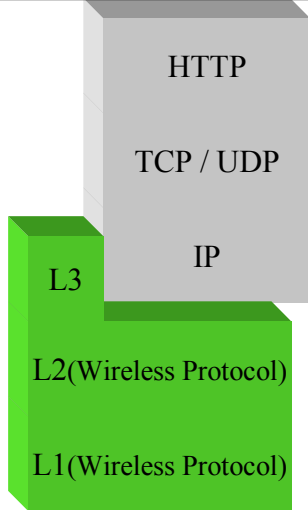
Discover What's Possible™
MD8470A-E-E-1

Slide 12

Anritsu

Packet Communication (HTTP)

Protocol Sequence (HTTP Response)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 1 | 0.000000 | 192.168.1.1 | Broadcast | ARP | who has 192.168.1.2? Tell 192.168.1.1 |
| 2 | 0.000025 | 192.168.1.2 | 192.168.1.1 | ARP | 192.168.1.2 is at 00:10:71:00:56:b9 |
| 3 | 0.009941 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [SYN] Seq=0 Ack=0 Win=3920 |
| 4 | 0.010002 | 192.168.1.2 | 192.168.1.1 | TCP | http > 6616 [SYN, ACK] Seq=0 Ack=1 Win= |
| 5 | 0.259874 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [ACK] Seq=1 Ack=1 Win=3920 |
| 6 | 0.320626 | 192.168.1.1 | 192.168.1.2 | HTTP | GET / HTTP/1.1 |
| 7 | 0.322096 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (text/html) |
| 8 | 0.959824 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [ACK] Seq=518 Ack=924 Win= |
| 9 | 5.959447 | 192.168.1.1 | 192.168.1.2 | TCP | [TCP Dup ACK 8#1] 6616 > http [ACK] Se |
| 10 | 16.438704 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [RST] Seq=518 Ack=40341 [R |


```

Frame 7 (977 bytes on wire, 977 bytes captured)
Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:91:03:19:15
Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
Transmission Control Protocol, Src Port: http (80), Dst Port: 6616 (6616), Seq: 1, Ack: 518, Len:
Hypertext Transfer Protocol
  HTTP/1.1 200 OK\r\n
  Server: Microsoft-IIS/5.1\r\n
  X-Powered-By: ASP.NET\r\n
  Content-Location: http://192.168.1.2/default.htm\r\n
  Date: Tue, 24 May 2005 03:20:48 GMT\r\n
  Content-Type: text/html\r\n
  Accept-Ranges: bytes\r\n
  Last-Modified: Fri, 08 Apr 2005 09:03:40 GMT\r\n
  ETag: "60f5acdb193cc51:acd"\r\n
  Content-Length: 624\r\n
  \r\n
  Line-based text data: text/html
  <html>
  <head>
  
```

- *1 : 'Http/1.1 200 OK' means 'success of downloading'
- 'Content-Location' : path of the content
- 'Content-Type' : MIME type of the content
- 'Content-Length' : Length of the content
- *2 : HTML content data itself

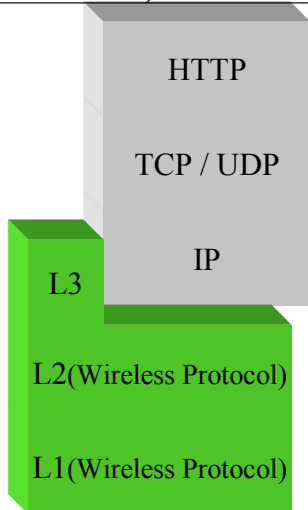
Discover What's Possible™
MD8470A-E-E-1

Slide 13

Anritsu

Packet Communication (HTTP)

Protocol Sequence (TCP Ack)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 1 | 0.000000 | 192.168.1.1 | Broadcast | ARP | who has 192.168.1.2? Tell 192.168.1.1 |
| 2 | 0.000025 | 192.168.1.2 | 192.168.1.1 | ARP | 192.168.1.2 is at 00:10:71:00:56:b9 |
| 3 | 0.009941 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [SYN] Seq=0 Ack=0 Win=3920 |
| 4 | 0.010002 | 192.168.1.2 | 192.168.1.1 | TCP | http > 6616 [SYN, ACK] Seq=0 Ack=1 Win= |
| 5 | 0.259874 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [ACK] Seq=1 Ack=1 Win=3920 |
| 6 | 0.320626 | 192.168.1.1 | 192.168.1.2 | HTTP | GET / HTTP/1.1 |
| 7 | 0.322096 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (text/html) |
| 8 | 0.959824 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [ACK] Seq=518 Ack=924 Win= |
| 9 | 5.959447 | 192.168.1.1 | 192.168.1.2 | TCP | [TCP Dup ACK 8#1] 6616 > http [ACK] Se |
| 10 | 16.438704 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [RST] Seq=518 Ack=40341 [R |


```

Frame 8 (60 bytes on wire, 60 bytes captured)
Ethernet II, Src: 00:00:91:03:19:15, Dst: 00:10:71:00:56:b9
Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
Transmission Control Protocol, Src Port: 6616 (6616), Dst Port: http (80), Seq: 518, Ack: 924, Len:
  Source port: 6616 (6616)
  Destination port: http (80)
  Sequence number: 518 (relative sequence number)
  Acknowledgement number: 924 (relative ack number)
  Header length: 20 bytes
  Flags: 0x0010 (ACK)
  Window size: 39200
  Checksum: 0xFF42 (correct)
  [SEQ/ACK analysis]
  [This is an ACK to the segment in frame: 7]
  [The RTT to ACK the segment was: 0.637728000 seconds]
  
```

- *1 : TCP level acknowledge of previous HTTP content access

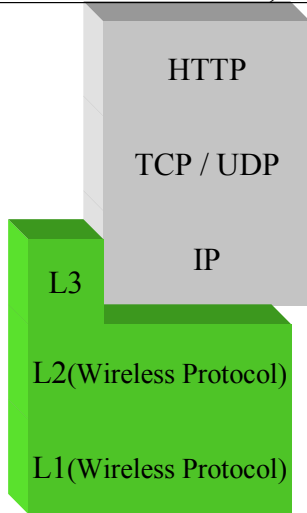
Discover What's Possible™
MD8470A-E-E-1

Slide 14

Anritsu

Packet Communication (HTTP)

Protocol Sequence (TCP Disconnect)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 1 | 0.000000 | 192.168.1.1 | Broadcast | ARP | who has 192.168.1.2? Tell 192.168.1.1 |
| 2 | 0.000025 | 192.168.1.2 | 192.168.1.1 | ARP | 192.168.1.2 is at 00:10:71:00:56:b9 |
| 3 | 0.000941 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [SYN] seq=0 Ack=0 win=3920 |
| 4 | 0.010002 | 192.168.1.2 | 192.168.1.1 | TCP | http > 6616 [SYN, ACK] seq=0 Ack=1 win= |
| 5 | 0.298874 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [ACK] seq=1 Ack=1 win=3920 |
| 6 | 0.320626 | 192.168.1.1 | 192.168.1.2 | HTTP | GET / HTTP/1.1 |
| 7 | 0.322096 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (text/html) |
| 8 | 0.959824 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [ACK] seq=518 Ack=924 win= |
| 9 | 5.959447 | 192.168.1.1 | 192.168.1.2 | TCP | [TCP Dup ACK 8#1] 6616 > http [ACK] se |
| 10 | 16.438704 | 192.168.1.1 | 192.168.1.2 | TCP | 6616 > http [RST] seq=518 Ack=40341728 |

*1 : Disconnect request

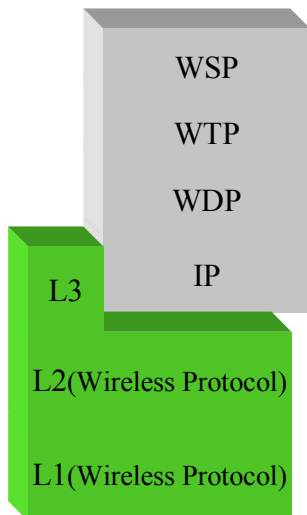
Discover What's Possible™
MD8470A-E-E-1

Slide 15

Anritsu

Packet Communication (WAP)

Protocol Stack



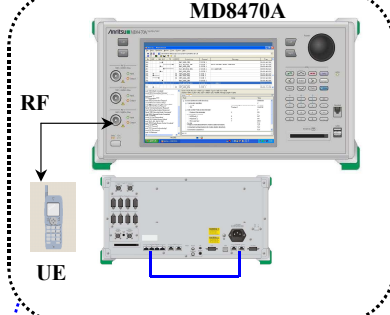
- WSP : Wireless Session Protocol
- WTP : Wireless Transaction Protocol
- WDP : Wireless Datagram Protocol
- IP : Internet Protocol
- L2(W-CDMA) : MAC / RLC
 - MAC : Media Access Control
 - RLC : Radio Link Control
- L2(GSM/GPRS) : DL /
 - DL : Data Link
- L1(W / G) : PHY(Physical Layer)

Discover What's Possible™
MD8470A-E-E-1

Slide 16

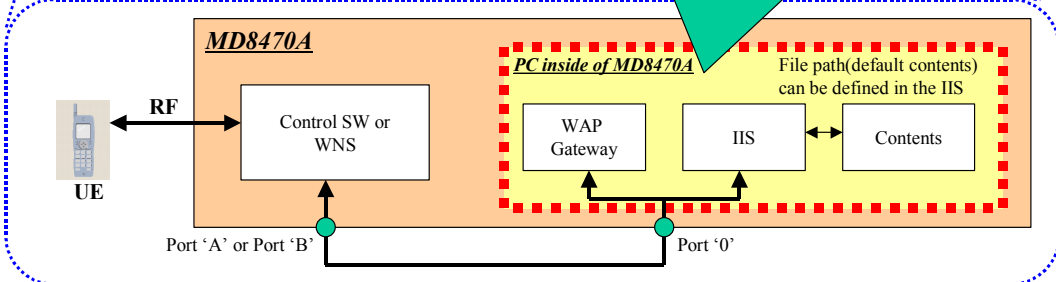
Anritsu

Packet Communication (WAP)



- ✓ **WAP Gateway**
- Converting to the format supported by the UE and downsizing the contents
- MD8470A uses 'Kannel' free software V1.4

All these parts are available in an external PC

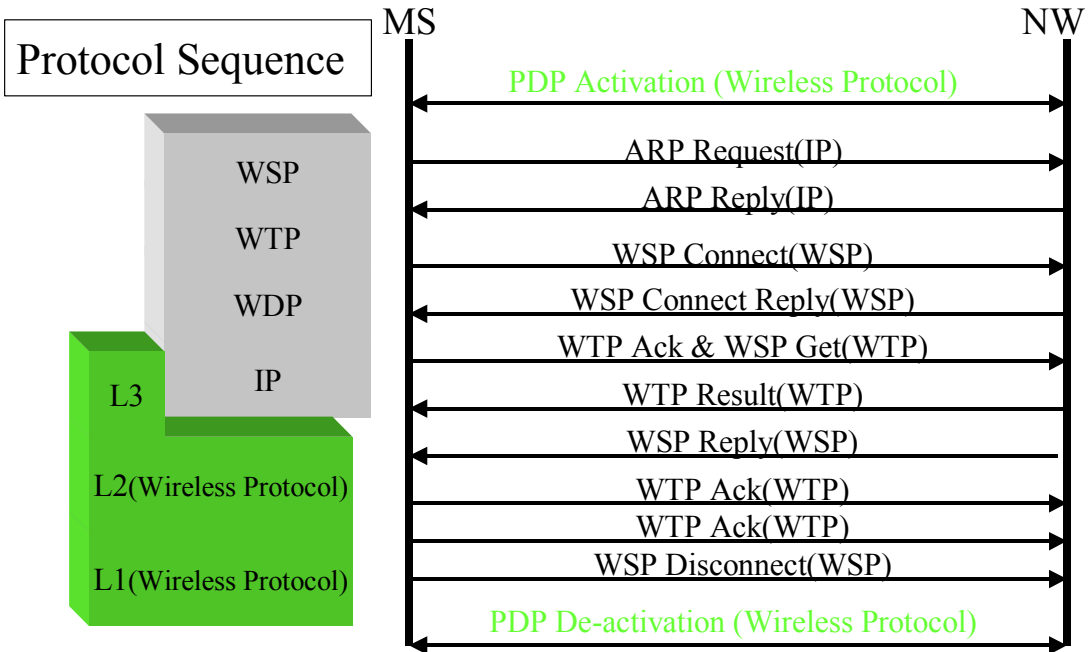


Discover What's Possible™
MD8470A-E-E-1

Slide 17



Packet Communication (WAP-HTML)



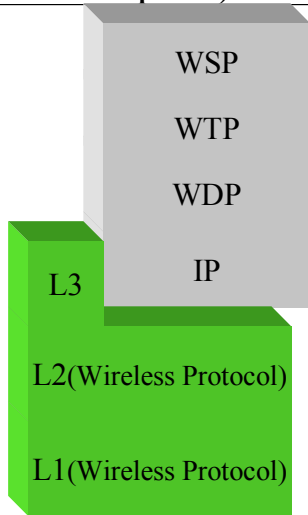
Discover What's Possible™
MD8470A-E-E-1

Slide 18



Packet Communication (WAP-HTML)

Protocol Sequence (ARP Request)



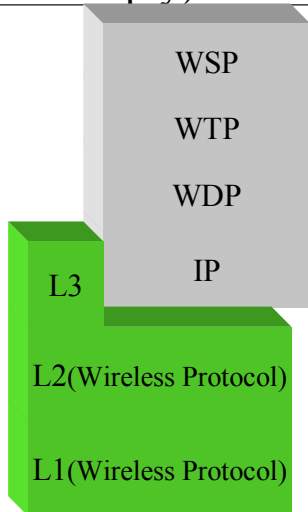
| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 1 | 0.000000 | 192.168.1.1 | Broadcast | ARP | who has 192.168.1.2? Tell 192.168.1.1 |
| 2 | 0.000027 | 192.168.1.2 | 192.168.1.1 | ARP | 192.168.1.2 is at 00:10:71:00:56:b9 |
| 3 | 0.010279 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WSP Connect (0x01) |
| 4 | 0.067025 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WSP ConnectReply (0x02) |
| 5 | 0.400141 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack, WSP Get (0x40) http://192.168 |
| 6 | 0.519352 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WTP Result (Unreassembled Fragment 0) |
| 7 | 0.529427 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WSP Reply (0x04): 200 OK (0x20) (text/ |
| 8 | 0.819858 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack |
| 9 | 4.859544 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack |
| 10 | 14.278843 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WSP Disconnect (0x05) |

Frame 1 (60 bytes on wire, 60 bytes captured)
 Ethernet II, Src: 00:00:91:03:19:15, Dst: ff:ff:ff:ff:ff:ff
 Address Resolution Protocol (request)
 Hardware type: Ethernet (0x0001)
 Protocol type: IP (0x0800)
 Hardware size: 6
 Protocol size: 4
 Opcode: request (0x0001)
 Sender MAC address: 00:00:91:03:19:15 (192.168.1.1)
 Sender IP address: 192.168.1.1 (192.168.1.1)
 Target MAC address: 00:00:00:00:00:00 (00:00:00:00:00:00)
 Target IP address: 192.168.1.2 (192.168.1.2) *1

*1 : MS(IP : 192.168.1.1) searches the MAC address of the Server(IP : 192.168.1.2)

Packet Communication (WAP-HTML)

Protocol Sequence (ARP Reply)



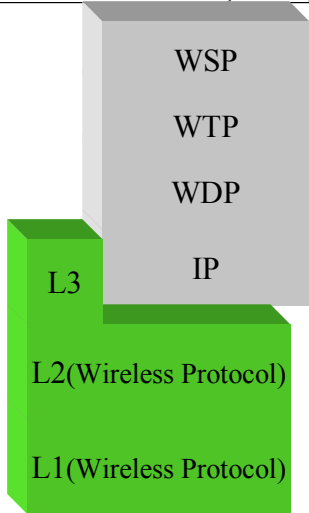
| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 1 | 0.000000 | 192.168.1.1 | Broadcast | ARP | who has 192.168.1.2? Tell 192.168.1.1 |
| 2 | 0.000027 | 192.168.1.2 | 192.168.1.1 | ARP | 192.168.1.2 is at 00:10:71:00:56:b9 |
| 3 | 0.010279 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WSP Connect (0x01) |
| 4 | 0.067025 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WSP ConnectReply (0x02) |
| 5 | 0.400141 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack, WSP Get (0x40) http://192.168 |
| 6 | 0.519352 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WTP Result (Unreassembled Fragment 0) |
| 7 | 0.529427 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WSP Reply (0x04): 200 OK (0x20) (text/ |
| 8 | 0.819858 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack |
| 9 | 4.859544 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack |
| 10 | 14.278843 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WSP Disconnect (0x05) |

Frame 2 (42 bytes on wire, 42 bytes captured)
 Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:91:03:19:15
 Address Resolution Protocol (reply)
 Hardware type: Ethernet (0x0001)
 Protocol type: IP (0x0800)
 Hardware size: 6
 Protocol size: 4
 Opcode: reply (0x0002)
 Sender MAC address: 00:10:71:00:56:b9 (192.168.1.2)
 Sender IP address: 192.168.1.2 (192.168.1.2)
 Target MAC address: 00:00:91:03:19:15 (192.168.1.1)
 Target IP address: 192.168.1.1 (192.168.1.1) *1

*1 : Server (IP : 192.168.1.2) replies the MAC address to MS(IP : 192.168.1.1)

Packet Communication (WAP-HTML)

Protocol Sequence (WSP Connect)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|-------------------------------------|
| 2 | 0.000027 | 192.168.1.2 | 192.168.1.1 | ARP | 192.168.1.2 is at 00:10:71:00:56:b9 |
| 3 | 0.067025 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WSP Connect (0x01) |
| 4 | 0.067025 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WSP ConnectReply (0x02) *2 |

```

Wireless Transaction Protocol, PDU: Invoke (1), Transaction Class: Reliable Invoke with Reliable R
1... .. = Continue Flag: TP3 Present *1
.000 1... = PDU Type: Invoke (0x01)
.... .01 = Trailer Flags: Last packet of message (0x01)
.... .10 = Re-transmission Indicator: First transmission
0... .. = TID Response: Original
.000 0000 0000 0001 = Transaction ID: 0x0001 *1
00... .. = Version: Current (0x00)
...1 .... = TIDNew: TID is valid
.... 00.. = U/P Flag: User Acknowledgement required
.... 00.. = Reserved: 0x00
... .10 = Transaction Class: Reliable Invoke with Reliable Result (0x02)
TP1: option (0x02)
Wireless Session Protocol, Method: Connect (0x01), version: 1.0
PDU Type: Connect (0x01)
0001 .... = version (Major): 1 *2
.... 0000 = version (Minor): 0
Capabilities Length: 19
Headers Length: 203
Capabilities
Method MOR: 4
Push MOR: 1
Server SDU size: 356352 *3
Client SDU size: 344250
    
```

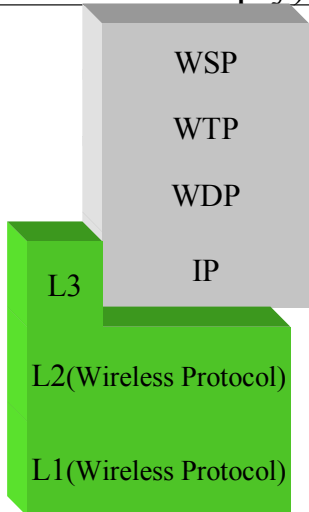
*1 : WTP 'PDU Type : Invoke' : request to WSP (Transaction ID = 1)

*2 : WSP 'PDU Type : Connect' : WSP connect WSP version is 1.0

*3 : Size of sending in 1 time
 Server SDU Size : 356352
 Client SDU Size : 344250

Packet Communication (WAP-HTML)

Protocol Sequence (WSP Conn. Reply)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|--|
| 3 | 0.010279 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WSP Connect (0x01) |
| 4 | 0.067025 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WSP ConnectReply (0x02) *2 |
| 5 | 0.400141 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack, WSP Get (0x40) http://192.168.1.1 |

```

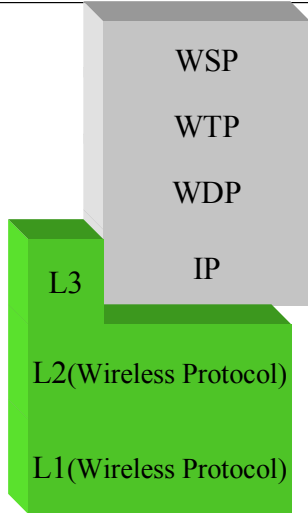
Frame 4 (89 bytes on wire, 89 bytes captured)
Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:91:103:19:15
Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
User Datagram Protocol, Src Port: 9201 (9201), Dst Port: 49152 (49152)
Wireless Transaction Protocol, PDU: Result (2)
0... .. = Continue Flag: No TPI
.001 0... = PDU Type: Result (0x02) *1
.... .01 = Trailer Flags: Last packet of message (0x01)
.... .10 = Re-transmission Indicator: First transmission
1... .. = TID Response: Response
.000 0000 0000 0001 = Transaction ID: 0x0001 *1
Wireless Session Protocol, Method: connectReply (0x02), session ID: 0
PDU Type: connectReply (0x02) *2
Server session ID: 0
Capabilities Length: 19
Headers Length: 21
Capabilities
Client SDU size: 344250
Server SDU size: 356352
Protocol options:
Method MOR: 4
[Malformed Packet: WSP]
    
```

*1 : WTP 'PDU Type : Result' : reply from WSP (Transaction ID = 1)

*2 : WSP 'PDU Type : ConnectReply' : WSP Connect Reply

Packet Communication (WAP-HTML)

Protocol Sequence (WTP Ack & WSP Get)



```

No.  Time      Source      Destination  Protocol  Info
4 0.067025 192.168.1.2 192.168.1.1  WTP+WS  WSP ConnectReply (0x02)
5 0.519352 192.168.1.2 192.168.1.1  WTP+WS  WTP ACK (0x03) http://192.168.1.2
6 0.519352 192.168.1.2 192.168.1.1  WTP+WS  WTP Result (Unreassembled Fragment 0)

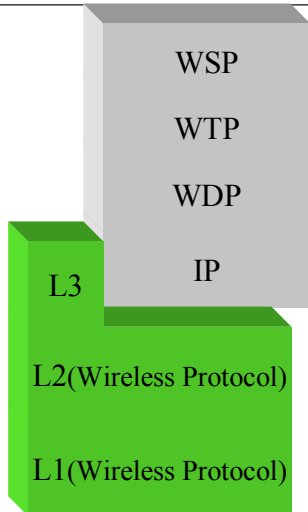
Wireless Transaction Protocol, PDU concatenation, PDU count: 3
Sub PDU size: 3
Wireless Transaction Protocol, PDU: ACK (3)
0... .. = Continue Flag: No TPI
.001 1... = PDU Type: Ack (0x03) *1
... ..0. = Tve/Tok Flag: False
... ..0. = Re-transmission Indicator: First transmission
0... ..0. = TID Response: Original
.000 0000 0000 0001 = Transaction ID: 0x0001 *1
Sub PDU size: 179
Wireless Transaction Protocol, PDU: Invoke (1), Transaction Class: Reliable Invoke with Reliable
1... ..0. = Continue Flag: TPI Present
.000 1... = PDU Type: Invoke (0x01) *2
... ..01. = Trailer Flags: Last packet of message (0x01)
... ..0. = Re-transmission Indicator: First transmission
0... ..0. = TID Response: Original
.000 0000 0000 0010 = Transaction ID: 0x0002 *2
00... ..0. = Version: Current (0x00)
..0. .... = TIDNew: TID is valid
...1 .... = U/P Flag: User Acknowledgement required
... ..00. = Reserved: 0x00
... ..10 = Transaction Class: Reliable Invoke with Reliable Result (0x02)
TPI: Option (0x02)
Wireless Session Protocol, Method: Get (0x40), URI: http://192.168.1.2
    
```

*1 : First WTP 'PDU Type : Ack' : Ack of previous Reply (Transaction ID = 1)

*2 : Second WTP 'PDU Type : Invoke' : request to WSP (request of 'WSP Get', Transaction ID = 2)

Packet Communication (WAP-HTML)

Protocol Sequence (WTP Ack & WSP Get)



```

No.  Time      Source      Destination  Protocol  Info
4 0.067025 192.168.1.2 192.168.1.1  WTP+WS  WSP ConnectReply (0x02)
5 0.519352 192.168.1.2 192.168.1.1  WTP+WS  WTP ACK (0x03) http://192.168.1.2
6 0.519352 192.168.1.2 192.168.1.1  WTP+WS  WTP Result (Unreassembled Fragment 0)

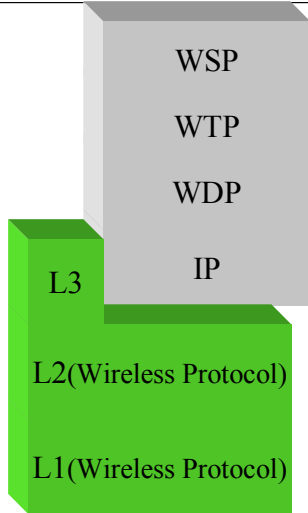
Wireless Session Protocol, Method: Get (0x40), URI: http://192.168.1.2
PDU Type: Get (0x40) *1
URI Length: 18
URI: http://192.168.1.2
Headers
User-Agent: Mozilla/SMB3(2105)/Samsung
Accept: application/vnd.wap.wmlc
Accept: application/vnd.wap.wmlscriptc
Accept: text/vnd.wap.wml
Accept: image/vnd.wap.wbmp
Accept: image/png
Accept: image/jpeg
Accept: image/gif
Accept: application/vnd.oma.ddi+xml
Accept: text/vnd.sun.j2me.app-descriptor
Accept: image/bmp
Accept: application/vnd.wap.multipart.mixed
Accept: */*
Accept-Language: English (en)
Accept-Charset: utf-8
X-VODAFONE-3GPPContext: yes
    
```

*1 : WSP 'PDU Type : Get' : request the content
URI : path of content

*2 : WSP > Headers >
Accept : supported MIME type
Accept-Language : English
Accept-Charaset : utf-8(common unicode)

Packet Communication (WAP-HTML)

Protocol Sequence (WTP Result)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---|
| 6 | 0.819352 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WTP Result (Unreassembled Fragment 0) |
| 7 | 0.829427 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WSP Reply (0x04): 200 OK (0x20) (text/html) |
| 8 | 0.819858 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack |

```

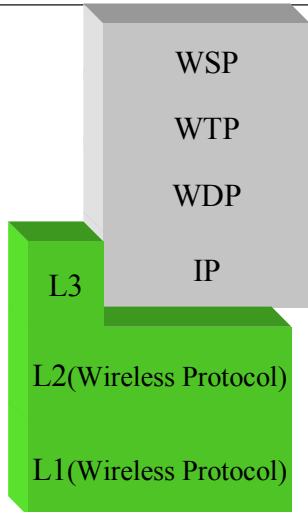
Frame 6 (621 bytes on wire, 621 bytes captured)
  Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:91:03:19:15
  Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
  User Datagram Protocol, Src Port: 9201 (9201), Dst Port: 49152 (49152)
  Wireless Transaction Protocol, PDU: Result (2)
    0... .. = Continue Flag: No TPI
    .001 0... = PDU Type: Result (0x02) *1
    .... 00. = Trailer Flags: Not last packet (0x00) *2
    .... ..0 = Re-transmission Indicator: First transmission
    1... .. = TID Response: Response
    .000 0000 0000 0010 = Transaction ID: 0x0002 *1
    Reassembled in: 7
    Payload
    
```

*1 : WTP 'PDU Type : Result' : reply from WSP (Transaction ID = 2)

*2 : 'Trailer Flags : Not Last Packet' means the segmentation of content data had happened
Downloading content is still continuing

Packet Communication (WAP-HTML)

Protocol Sequence (WSP Reply)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---|
| 6 | 0.819352 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WTP Result (Unreassembled Fragment 0) |
| 7 | 0.829427 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WSP Reply (0x04): 200 OK (0x20) (text/html) |
| 8 | 0.819858 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack |

```

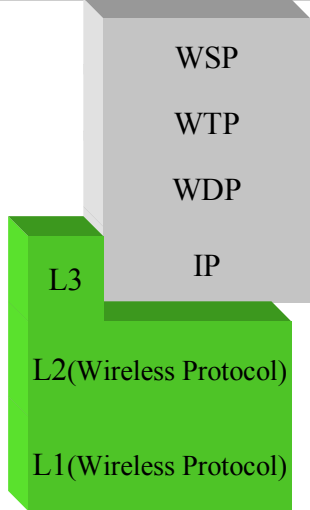
Frame 7 (211 bytes on wire, 211 bytes captured)
  Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:91:03:19:15
  Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
  User Datagram Protocol, Src Port: 9201 (9201), Dst Port: 49152 (49152)
  Wireless Transaction Protocol, PDU: Segmented Result (6), Packet Sequence Number: 1
    0... .. = Continue Flag: No TPI
    .011 0... = PDU Type: Segmented Result (0x06) *1
    .... 01. = Trailer Flags: Last packet of message (0x01) *2
    .... ..0 = Re-transmission Indicator: First transmission
    1... .. = TID Response: Response
    .000 0000 0000 0010 = Transaction ID: 0x0002 *1
    Packet Sequence Number: 1
    [WTP Fragments]
    [Frame: 6, payload: 0-575 (576 bytes)]
    [Frame: 7, payload: 576-710 (135 bytes)]
  Wireless session Protocol, Method: Reply (0x04), Status: 200 OK (0x20), Content-Type: text/html
    PDU Type: Reply (0x04)
    Status: 200 OK (0x20)
    Headers Length: 114
    Content-Type: text/html
    Headers
      Server: Microsoft-IIS/5.1
      X-Powered-By: ASP.NET
      Content-Location: http://192.168.1.2/default.htm
    
```

*1 : WTP 'PDU Type : Segmented Result' : reply from WSP (Transaction ID = 2) for the second data block

*2 : 'Trailer Flags : Last Packet of message' means the last part of the content data

Packet Communication (WAP-HTML)

Protocol Sequence (WSP Reply)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|------------|---|
| 6 | 0.519352 | 192.168.1.2 | 192.168.1.1 | WTP+WS WTP | Result (Unreassembled fragment 0) |
| 7 | 0.819858 | 192.168.1.1 | 192.168.1.2 | WTP+WS WSP | Reply (0x04): 200 OK (0x20) (text/html) |
| 8 | 0.819858 | 192.168.1.1 | 192.168.1.2 | WTP+WS WTP | Ack |


```

.000 0000 0000 0010 = Transaction ID: 0x0002
Packet Sequence Number: 1
  [WTP Fragments]
    [Frame: 6, payload: 0-575 (576 bytes)]
    [Frame: 7, payload: 576-740 (165 bytes)]
  [Wireless Session Protocol, Method: Reply (0x04), Status: 200 OK (0x20), Content-Type: text/html]
  PDU Type: Reply (0x04) *1
  Status: 200 OK (0x20)
  Headers Length: 114
  Content-Type: text/html
  [Headers]
    Server: Microsoft-IIS/5.1
    X-Powered-By: ASP.NET
    Content-Location: http://192.168.1.2/default.htm
    Date: May 24, 2005 11:55:30.000000000
    Accept-Ranges: bytes
    Last-Modified: Apr 8, 2005 18:03:40.000000000
    ETag: "60f5acdb193cc51:acd"
    Content-Length: 624
  [Line-based text data: text/html]
    <html>
    <head>
    <title>192.168.1.2 Direct</title>
    </head>
  
```

- *1 : WSP 'PDU Type : Reply' : reply the content access
'Status : 200 OK' means success of download
- *2 : content location and length
- *3 : html data itself

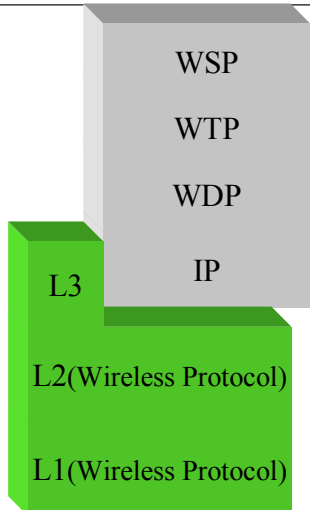
Discover What's Possible™
MD8470A-E-E-1

Slide 27

Anritsu

Packet Communication (WAP-HTML)

Protocol Sequence (WTP Ack)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|------------|---|
| 7 | 0.529427 | 192.168.1.2 | 192.168.1.1 | WTP+WS WSP | Reply (0x04): 200 OK (0x20) (text/html) |
| 8 | 0.859544 | 192.168.1.2 | 192.168.1.1 | WTP+WS WTP | Ack |
| 9 | 4.859544 | 192.168.1.1 | 192.168.1.2 | WTP+WS WTP | Ack |


```

Frame 8 (60 bytes on wire, 60 bytes captured)
  [Ethernet II, Src: 00:00:91:03:19:15, Dst: 00:10:71:00:56:b9]
  [Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)]
  [User Datagram Protocol, Src Port: 49152 (49152), Dst Port: 9201 (9201)]
  [Wireless Transaction Protocol, PDU: Ack (3)]
    1... .. = Continue Flag: TPI Present
    .001 1... = PDU Type: Ack (0x03) *1
    ....0. = Tve/Tok Flag: False
    ....0.0 = Re-transmission Indicator: First transmission
    0... .. = TID Response: original
    .000 0000 0000 0010 = Transaction ID: 0x0002 *1
  [TPI: Option (0x02)]
  [TPI: Packet sequence number (0x03)]
  
```

- *1 : WSP 'PDU Type : Ack' : Ack for the WTP result of first content data block (Transaction ID = 2)

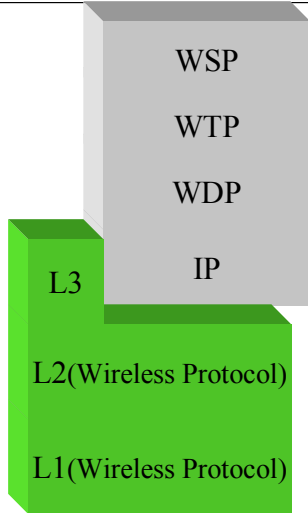
Discover What's Possible™
MD8470A-E-E-1

Slide 28

Anritsu

Packet Communication (WAP-HTML)

Protocol Sequence (WTP Ack)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|-----------------------|
| 8 | 0.819858 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack |
| 9 | 4.839844 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack |
| 10 | 14.279843 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WSP Disconnect (0x05) |

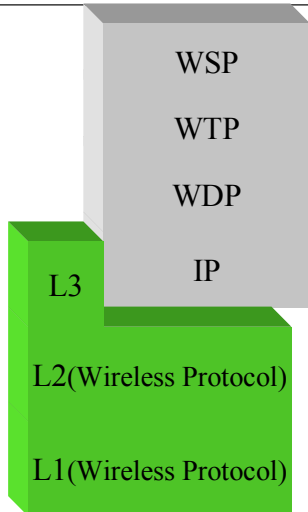
```

Frame 9 (60 bytes on wire, 60 bytes captured)
  Ethernet II, Src: 00:00:91:03:19:15, Dst: 00:10:71:00:56:b9
  Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
  User Datagram Protocol, Src Port: 49152 (49152), Dst Port: 9201 (9201)
  Wireless Transaction Protocol, PDU concatenation, PDU count: 2
    Sub PDU size: 3
    Wireless Transaction Protocol, PDU: ACK (3)
      0... .. = Continue Flag: No TPI
      .001 1... = PDU Type: Ack (0x03) *1
      ....0. = TUE/TOK Flag: False
      ....0.0 = Re-transmission Indicator: First transmission
      0... .. = TID Response: Original
      .000 0000 0000 0010 = Transaction ID: 0x0002 *1
    
```

*1 : WSP 'PDU Type : Ack' : Ack for the WTP result of second(last) content data block (Transaction ID = 2)

Packet Communication (WAP-HTML)

Protocol Sequence (WSP Disconnect)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|-----------------------|
| 8 | 0.819858 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack |
| 9 | 4.839844 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack |
| 10 | 14.279843 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WSP Disconnect (0x05) |

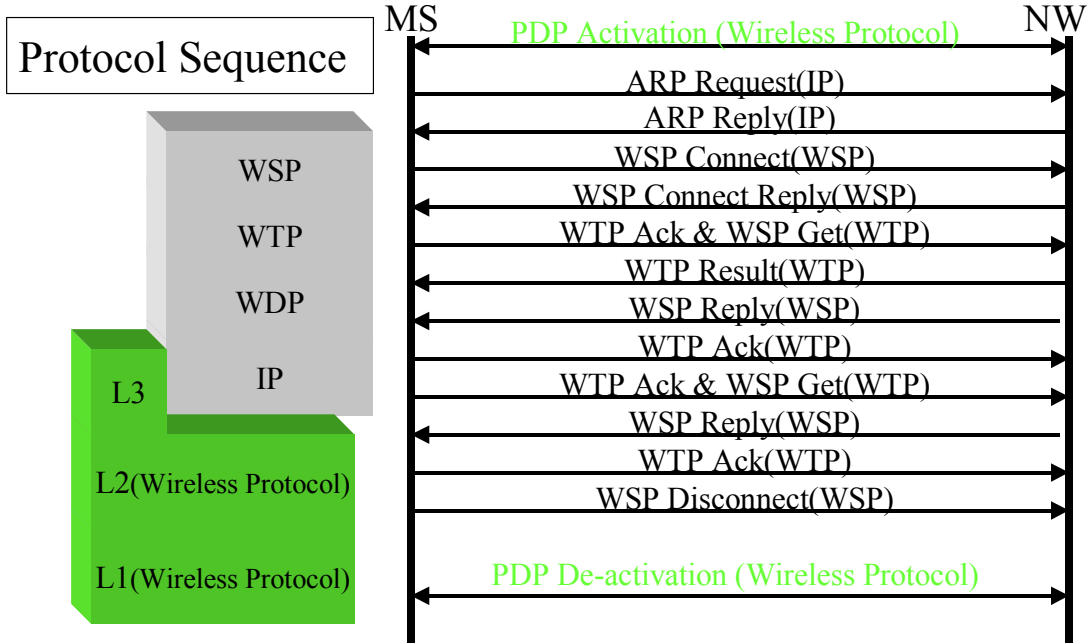
```

Frame 10 (60 bytes on wire, 60 bytes captured)
  Ethernet II, Src: 00:00:91:03:19:15, Dst: 00:10:71:00:56:b9
  Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
  User Datagram Protocol, Src Port: 49152 (49152), Dst Port: 9201 (9201)
  Wireless Transaction Protocol, PDU: Invoke (1), Transaction Class: Unreliable Invoke without Result
    0... .. = Continue Flag: No TPI
    .000 1... = PDU Type: Invoke (0x01) *1
    ....11. = Trailer Flags: Re-assembly not supported (0x03)
    ....0.0 = Re-transmission Indicator: First transmission
    0... .. = TID Response: Original
    .000 0000 0000 0011 = Transaction ID: 0x0003 *1
    .00... .. = Version: Current (0x00)
    ...0... = TIDNew: TID is valid
    ...0... = U/P Flag: User Acknowledgement optional
    ...00.. = Reserved: 0x00
    ....00 = Transaction Class: Unreliable Invoke without Result (0x00)
  Wireless Session Protocol, Method: Disconnect (0x05), Session ID: 0
    PDU Type: Disconnect (0x05) *2
    Server Session ID: 0
    
```

*1 : WTP 'PDU Type : Invoke' : request to WSP (request of 'WSP Disconnect', Transaction ID = 3)

*2 : WSP 'PDU Type : Disconnect' : disconnection

Packet Communication (WAP-WML)



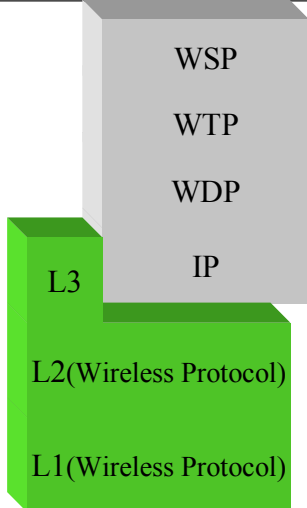
Discover What's Possible™
MD8470A-E-E-1

Slide 31

Anritsu

Packet Communication (WAP-WML)

Protocol Sequence (ARP Request)



| No. | Time | Source | Destination | Protocol Info |
|-----|----------|-------------|-------------|---|
| 1 | 0.000000 | 192.168.1.1 | 192.168.1.2 | WTP+WS WSP Connect (0x01) |
| 2 | 0.026588 | 192.168.1.2 | 192.168.1.1 | ARP who has 192.168.1.1? Tell 192.168.1.1 |
| 3 | 0.029539 | 192.168.1.1 | 192.168.1.2 | ARP 192.168.1.1 is at 00:00:91:03:19:15 |
| 4 | 0.029551 | 192.168.1.2 | 192.168.1.1 | WTP+WS WSP ConnectReply (0x02) |
| 5 | 0.339914 | 192.168.1.1 | 192.168.1.2 | WTP+WS WTP Ack, WSP get (0x40) http://192.168.1.1 |
| 6 | 0.444060 | 192.168.1.2 | 192.168.1.1 | WTP+WS WTP Result (Unreassembled Fragment 0) |
| 7 | 0.444899 | 192.168.1.2 | 192.168.1.1 | WTP+WS WSP Reply (0x04): 200 OK (0x20) |
| 8 | 0.719643 | 192.168.1.1 | 192.168.1.2 | WTP+WS WTP Ack |
| 9 | 1.219860 | 192.168.1.1 | 192.168.1.2 | WTP+WS WTP Ack, WSP get (0x40) http://192.168.1.1 |
| 10 | 1.336886 | 192.168.1.2 | 192.168.1.1 | WTP+WS WSP Reply (0x04): 200 OK (0x20) (image) |

| | | | |
|--|--|--|--|
| Frame 2 (42 bytes on wire, 42 bytes captured) | | | |
| Ethernet II, Src: 00:10:71:00:56:b9, Dst: ff:ff:ff:ff:ff:ff | | | |
| Address Resolution Protocol (request) | | | |
| Hardware type: Ethernet (0x0001) | | | |
| Protocol type: IP (0x0800) | | | |
| Hardware size: 6 | | | |
| Protocol size: 4 | | | |
| Opcode: request (0x0001) | | | |
| Sender MAC address: 00:10:71:00:56:b9 (192.168.1.2) | | | |
| Sender IP address: 192.168.1.2 (192.168.1.2) | | | |
| Target MAC address: 00:00:00:00:00:00 (00:00:00:00:00:00) *1 | | | |
| Target IP address: 192.168.1.1 (192.168.1.1) | | | |

*1 : MS(IP : 192.168.1.1) searches the MAC address of the Server(IP : 192.168.1.2)

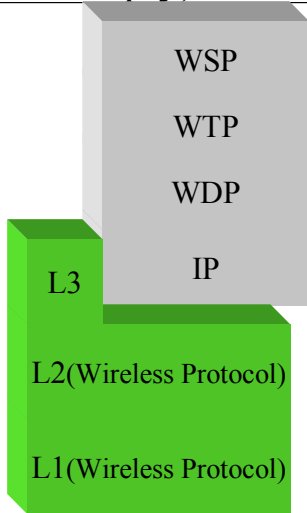
Discover What's Possible™
MD8470A-E-E-1

Slide 32

Anritsu

Packet Communication (WAP-WML)

Protocol Sequence (ARP Reply)

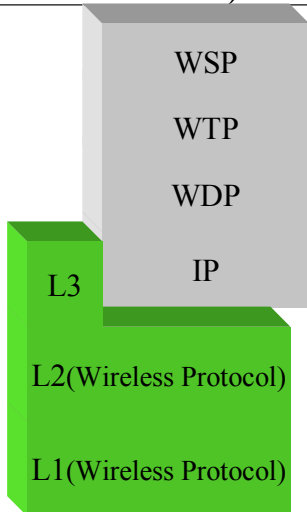


| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|--|
| 1 | 0.000000 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WSP Connect (0x01) |
| 2 | 0.026388 | 192.168.1.2 | Broadcast | ARP | who has 192.168.1.1? Tell 192.168.1.2 |
| 3 | 0.029339 | 192.168.1.1 | 192.168.1.2 | ARP | 192.168.1.1 is at 00:10:71:00:56:b9 |
| 4 | 0.029551 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WSP ConnectReply (0x02) |
| 5 | 0.359914 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack, WSP Get (0x04) http://192.168.1.1 |
| 6 | 0.444060 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WTP Result (Unreassembled frame) |
| 7 | 0.444899 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WSP Reply (0x04): 200 OK (0x20) |
| 8 | 0.719643 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack |
| 9 | 1.219860 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack, WSP Get (0x04) http://192.168.1.1 |
| 10 | 1.336886 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WSP Reply (0x04): 200 OK (0x20) (image) |

*1 : Server (IP : 192.168.1.2) replies the MAC address to MS(IP : 192.168.1.1)

Packet Communication (WAP-WML)

Protocol Sequence (WSP Connect)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---------------------------------------|
| 1 | 0.000000 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WSP Connect (0x01) |
| 2 | 0.026388 | 192.168.1.2 | Broadcast | ARP | who has 192.168.1.1? Tell 192.168.1.2 |
| 3 | 0.029339 | 192.168.1.1 | 192.168.1.2 | ARP | 192.168.1.1 is at 00:10:71:00:56:b9 |

```

Wireless Transaction Protocol, PDU: Invoke (1), transaction class: Reliable Invoke with Reliable R
1..... = Continue Flag: TPI Present
.000 1... = PDU Type: Invoke (0x01) *1
.... .01. = Trailer Flags: Last packet of message (0x01)
.... .0. = Re-transmission Indicator: First transmission
0..... = TID Response: Original
.000 0000 0000 0101 = Transaction ID: 0x0005 *1
00. .... = Version: Current (0x00)
.0. .... = TIDNew: TID is valid
...1 .... = U/P Flag: User Acknowledgement required
.... 00. = Reserved: 0x00
.... .10 = Transaction Class: Reliable Invoke with Reliable Result (0x02)
Wireless Session Protocol, Method: Connect (0x01), version: 1.0
PDU Type: Connect (0x01) *2
0001 .... = Version (Major): 1
.... 0000 = version (Minor): 0
Capabilities Length: 19
Headers Length: 203
Capabilities
Method MOR: 4
Push MOR: 1
Server SDU size: 356352 *3
Client SDU size: 344250
    
```

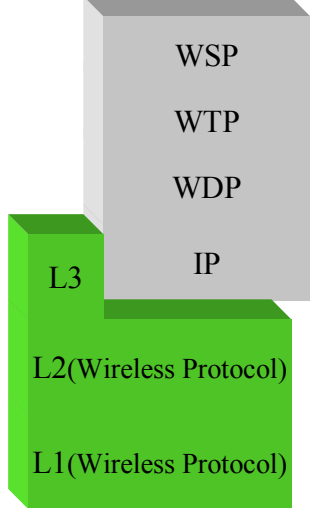
*1 : WTP 'PDU Type : Invoke' : request to WSP (Transaction ID = 5)

*2 : WSP 'PDU Type : Connect' : WSP connect WSP version is 1.0

*3 : Size of sending in 1 time
Server SDU Size : 356352
Client SDU Size : 344250

Packet Communication (WAP-WML)

Protocol Sequence (WSP Conn. Reply)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|--------------------------------|-------------------------------------|
| 3 | 0.029539 | 192.168.1.1 | 192.168.1.2 | ARP | 192.168.1.1 is at 00:00:91:03:19:15 |
| 4 | 0.030945 | 192.168.1.2 | 192.168.1.1 | WTP+WS WSP ConnectReply (0x02) | |
| 5 | 0.359914 | 192.168.1.1 | 192.168.1.2 | WTP+WS WTP Ack, WSP Get (0x40) | http://192.168... |


```

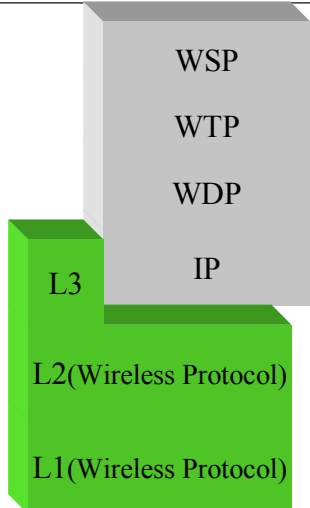
Frame 4 (89 bytes on wire, 89 bytes captured)
Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:91:03:19:15
Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
User Datagram Protocol, Src Port: 9201 (9201), Dst Port: 49152 (49152)
Wireless Transaction Protocol, PDU: Result (2)
0..... = Continue Flag: No TPI
.001 0... = PDU Type: Result (0x02) *1
.... 01 = Trailer Flags: Last packet of message (0x01)
.... ..0 = Re-transmission Indicator: First transmission
1..... = TID Response: Response
[000 0000 0000 0101 = Transaction ID: 0x0005] *1
Wireless Session Protocol, Method: connectReply (0x42), Session ID: 3
PDU Type: connectReply (0x02) *2
Server Session ID: 3
Capabilities Length: 19
Headers Length: 21
Capabilities
Client SDU size: 344250
Server SDU size: 356352
Protocol Options:
Method MCR: 4
[Malformed Packet: WSP]
    
```

*1 : WTP 'PDU Type : Result' : reply from WSP (Transaction ID = 5)

*2 : WSP 'PDU Type : ConnectReply' : WSP Connect Reply

Packet Communication (WAP-WML)

Protocol Sequence (WTP Ack & WSP Get)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|--|-------------------|
| 4 | 0.029551 | 192.168.1.2 | 192.168.1.1 | WTP+WS WSP ConnectReply (0x02) | |
| 5 | 0.030945 | 192.168.1.2 | 192.168.1.1 | WTP+WS WTP Ack (0x40) | http://192.168... |
| 6 | 0.444060 | 192.168.1.2 | 192.168.1.1 | WTP+WS WTP Result (Unreassembled Fragment 0) | |


```

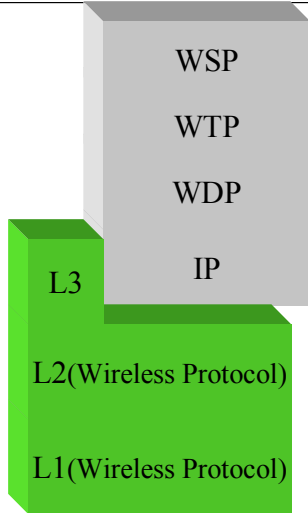
Wireless Transaction Protocol, PDU concatenation, PDU count: 3
Sub PDU size: 3
Wireless Transaction Protocol, PDU: ACK (3)
0..... = Continue Flag: No TPI
.001 1... = PDU Type: Ack (0x03) *1
.... 01 = Trailer Flags: Last packet of message (0x01)
.... ..0 = Re-transmission Indicator: First transmission
1..... = TID Response: Original
[000 0000 0000 0101 = Transaction ID: 0x0005] *1
Sub PDU size: 192
Wireless Transaction Protocol, PDU: Invoke (1), Transaction Class: Reliable Invoke with Reliable
1..... = Continue Flag: TPI Present
[000 1... = PDU Type: Invoke (0x01)] *2
.... 01 = Trailer Flags: Last packet of message (0x01)
.... ..0 = Re-transmission Indicator: First transmission
0..... = TID Response: Original
[000 0000 0000 0110 = Transaction ID: 0x0006] *2
00..... = Version: Current (0x00)
..0..... = TIDNew: TID is valid
...1.... = U/P Flag: User Acknowledgement required
.... 00.. = Reserved: 0x00
.... 10 = Transaction Class: Reliable Invoke with Reliable Result (0x02)
TPI: Option (0x02)
Wireless Session Protocol, Method: Get (0x40), URI: http://192.168.1.2/wml/main.wml
    
```

*1 : First WTP 'PDU Type : Ack' : Ack of previous Reply (Transaction ID = 5)

*2 : Second WTP 'PDU Type : Invoke' : request to WSP (request of 'WSP Get', Transaction ID = 6)

Packet Communication (WAP-WML)

Protocol Sequence (WTP Ack & WSP Get)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---|
| 4 | 0.029551 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WSP ConnectReply (0x02) |
| 5 | 0.359914 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack, WSP Get (0x04) http://192.168.1.2/wml/main.wml |
| 6 | 0.444060 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WTP Result (Unreassembled Fragment: 0) |


```

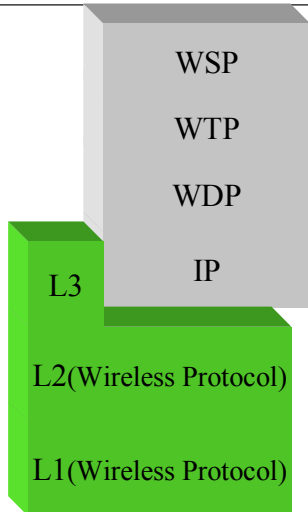
... 00.. = Reserved: 0x00
... ..10 = Transaction Class: Reliable Invoke with Reliable Result (0x02)
TPF: Option (0x02)
Wireless Session Protocol, Method: Get (0x40), URI: http://192.168.1.2/wml/main.wml
PDU Type: Get (0x40)
URI Length: 31
URI: http://192.168.1.2/wml/main.wml *1
Headers
User-Agent: Mozilla/SMB3(Z105)/Samsung
Accept: application/vnd.wap.wmlc
Accept: application/vnd.wap.wmlscriptc
Accept: text/vnd.wap.wml
Accept: image/vnd.wap.wbmp
Accept: image/png
Accept: image/jpeg
Accept: image/gif
Accept: application/vnd.oma.dd+xml
Accept: text/vnd.sun.j2me.app-descriptor
Accept: image/bmp
Accept: application/vnd.wap.multipart.mixed
Accept: */*
Accept-Language: English (en)
Accept-Charset: utf-8
X-VODAFONE-3GPPDPCContext: yes
    
```

*1 : WSP 'PDU Type : Get' : request the content
URI : path of content

*2 : WSP > Headers >
Accept : supported MIME type
Accept-Language : English
Accept-Charaset : utf-8(common unicode)

Packet Communication (WAP-WML)

Protocol Sequence (WTP Result)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---|
| 5 | 0.359914 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack, WSP Get (0x04) http://192.168.1.2/wml/main.wml |
| 6 | 0.444060 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WTP Result (Unreassembled Fragment: 0) |
| 7 | 0.444999 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WSP Reply (0x04): 200 OK (0x20) *1 |


```

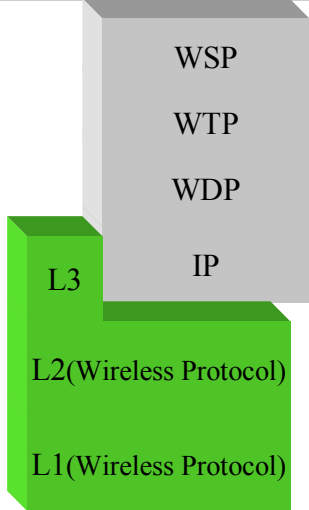
Frame 6 (621 bytes on wire, 621 bytes captured)
Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:91:03:19:15
Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
User Datagram Protocol, Src Port: 9201 (9201), Dst Port: 49152 (49152)
Wireless Transaction Protocol, PDU: Result (2)
0... .. = Continue Flag: No TPF *1
1001 0... = PDU type: Result (0x02) *1
.... 00.. = Trailer Flags: Not last packet (0x00) *2
.... ..0 = Re-transmission Indicator: First transmission
1... .. = TID Response: Response
0000 0000 0000 0010 = Transaction ID: 0x0006 *1
Reassembled in: 7
Payload
    
```

*1 : WTP 'PDU Type : Result' : reply from WSP
(Transaction ID = 6)

*2 : 'Trailer Flags : Not Last Packet' means the
segmentation of content data had happened
Downloading content is still continuing

Packet Communication (WAP-WML)

Protocol Sequence (WSP Reply)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|-------------------|----------------------------|
| 6 | 0.444060 | 192.168.1.2 | 192.168.1.1 | WTP+WS WTP Result | (Unreassembled Fragment 0) |
| 7 | 0.444899 | 192.168.1.2 | 192.168.1.1 | WTP+WS WSP Reply | (0x04): 200 OK (0x20) |
| 8 | 0.719643 | 192.168.1.1 | 192.168.1.2 | WTP+WS WTP Ack | |


```

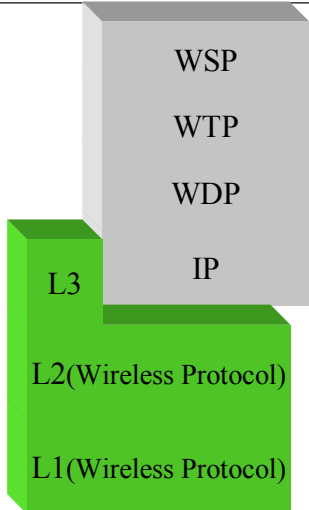
Frame 7 (123 bytes on wire (123 bytes captured)
# Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:91:03:19:15
# Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
# User Datagram Protocol, Src Port: 9201 (9201), Dst Port: 49152 (49152)
# Wireless Transaction Protocol, PDU: Segmented Result (6), Packet Sequence Number: 1
0...0... = continue Flag: No TPI
011 0... = PDU Type: Segmented Result (0x06) *1
....01 = Trailer Flags: Last packet of message (0x01) *2
...0 = Re-transmission Indicator: First transmission
1... = TID Response: Response
000 0000 0000 0110 = Transaction ID: 0x0006 *1
Packet Sequence Number: 1
[WTP Fragments]
# Wireless Session Protocol, Method: Reply (0x04), status: 200 OK (0x20), Content-Type: text/vnd.wap
PDU Type: Reply (0x04)
Status: 200 OK (0x20)
Headers Length: 81
Content-Type: text/vnd.wap.wml
# Headers
Server: Microsoft-IIS/5.1
X-Powered-By: ASP.NET
Date: May 24, 2005 12:06:02.000000000
Accept-Ranges: bytes
Last-Modified: Jun 23, 2004 17:26:04.000000000
    
```

*1 : WTP 'PDU Type : Segmented Result' : reply from WSP (Transaction ID = 6) for the second data block

*2 : 'Trailer Flags : Last Packet of message' means the last part of the content data

Packet Communication (WAP-WML)

Protocol Sequence (WSP Reply)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|-------------------|----------------------------|
| 6 | 0.444060 | 192.168.1.2 | 192.168.1.1 | WTP+WS WTP Result | (Unreassembled Fragment 0) |
| 7 | 0.444899 | 192.168.1.2 | 192.168.1.1 | WTP+WS WSP Reply | (0x04): 200 OK (0x20) |
| 8 | 0.719643 | 192.168.1.1 | 192.168.1.2 | WTP+WS WTP Ack | |


```

# Wireless Session Protocol, Method: Reply (0x04), Status: 200 OK (0x20), Content-Type: text/vnd.wap
PDU Type: Reply (0x04)
Status: 200 OK (0x20)
Headers Length: 81
Content-Type: text/vnd.wap.wml *1
# Headers
Server: Microsoft-IIS/5.1
X-Powered-By: ASP.NET
Date: May 24, 2005 12:06:02.000000000
Accept-Ranges: bytes
Last-Modified: Jun 23, 2004 17:26:04.000000000
ETag: "0362db9fb58c41:acd" *2
Content-Length: 569
# Extensible Markup Language
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN" "http://www.wapforum.org/DTD/wml1_1_1.xml"
# <wml>
# <card id="Main" title="Main Page">
# <p>
New signalling Tester
<br/>
MD8470A Debut!
<br/>
 *3
    
```

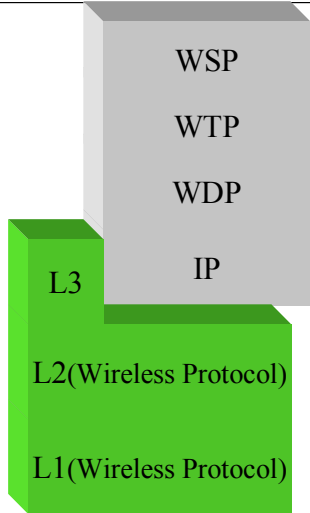
*1 : WSP 'PDU Type : Reply' : reply the content access 'Status : 200 OK' means success of download

*2 : content length

*3 : wml data itself

Packet Communication (WAP-WML)

Protocol Sequence (WTP Ack)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|-----------------------------------|
| 7 | 0.444899 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WSP Reply (0x04): 200 OK (0x20) |
| 8 | 0.719643 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack |
| 9 | 1.219960 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WSP Get (0x40) http://192.168.1.2 |

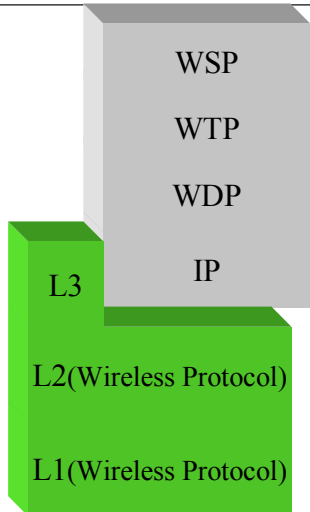
```

Frame 8 (60 bytes on wire, 60 bytes captured)
Ethernet II, Src: 00:00:51:03:19:15, Dst: 00:10:71:00:56:b9
Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
User Datagram Protocol, Src Port: 49152 (49152), Dst Port: 9201 (9201)
Wireless Transaction Protocol, PDU: ACK (3)
1. .... = Continue Flag: TPI Present
.001 1... = PDU Type: Ack (0x03) *1
....0. = Tve/Tok Flag: False
....0. = Re-transmission Indicator: First transmission
0. .... = TID Response: Original
.000 0000 0000 0110 = Transaction ID: 0x0006 *1
TPI: Option (0x02)
TPI: Packet sequence number (0x03)
    
```

*1 : WSP 'PDU Type : Ack' : Ack for the WTP result of first content data block (Transaction ID = 6)

Packet Communication (WAP-WML)

Protocol Sequence (WTP Ack & WSP Get)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|-------------------------------------|
| 8 | 0.719643 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WTP Ack |
| 9 | 1.219960 | 192.168.1.1 | 192.168.1.2 | WTP+WS | WSP Get (0x40) http://192.168.1.2 |
| 10 | 1.336886 | 192.168.1.2 | 192.168.1.1 | WTP+WS | WSP Reply (0x04): 200 OK *1,2 image |

```

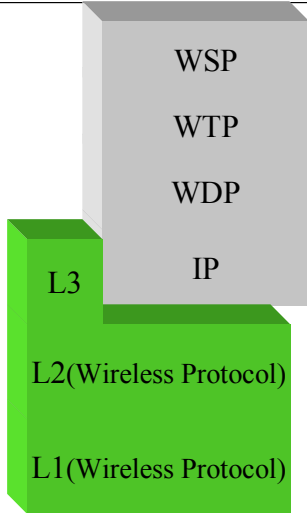
Wireless Transaction Protocol, PDU concatenation, PDU count: 3
Sub PDU size: 3
Wireless Transaction Protocol, PDU: ACK (3)
0. .... = Continue Flag: No TPI
.001 1... = PDU Type: Ack (0x03) *1
....0. = Tve/Tok Flag: False
....0. = Re-transmission Indicator: First transmission
0. .... = TID Response: Original
.000 0000 0000 0110 = Transaction ID: 0x0006 *1
Sub PDU size: 196
Wireless Transaction Protocol, PDU: Invoke (1), Transaction Class: Reliable Invoke with Reliable
1. .... = Continue Flag: TPI Present
.000 1... = PDU Type: Invoke (0x01) *2
....01 = Trailer Flags: Last packet of message (0x01)
....0. = Re-transmission Indicator: First transmission
0. .... = TID Response: Original
.000 0000 0000 0111 = Transaction ID: 0x0007 *2
00. .... = Version: Current (0x00)
.0. .... = TIDNew: TID is valid
...1 .... = U/P Flags: User Acknowledgement required
....00. = Reserved: 0x00
....10 = Transaction Class: Reliable Invoke with Reliable result (0x02)
TPI: Option (0x02)
Wireless Session Protocol, Method: Get (0x40), URI: http://192.168.1.2/wml/md8470a.wbmp
    
```

*1 : WSP 'PDU Type : Ack' : Ack for the WTP result of second(last) content data block (Transaction ID = 6)

*2 : Second WTP 'PDU Type : Invoke' : request to WSP (request of 'WSP Get', Transaction ID = 7)

Packet Communication (WAP-WML)

Protocol Sequence (WTP Ack & WSP Get)



```

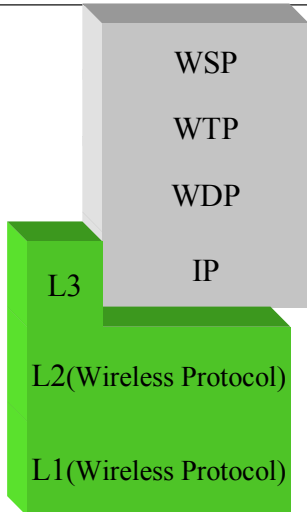
No.  Time      Source      Destination  Protocol  Info
8  0.719643  192.168.1.1  192.168.1.2  WTP+WS  WTP Ack
9  1.336886  192.168.1.2  192.168.1.1  WTP+WS  WTP Get (0x40) http://192.168.1.2/...
10 1.336886  192.168.1.2  192.168.1.1  WTP+WS  WSP Reply (0x04): 200 OK (0x20) image/vnd.wap.wbmp

... 00.. = Reserved: 0x00
... ..10 = Transaction Class: Reliable Invoke with Reliable Result (0x02)
... ..TPI: option (0x02)
Wireless Session Protocol, Method: Get (0x40), URI: http://192.168.1.2/wml/md8470a.wbmp
PDU Type: Get (0x40) *1
URI Length: 35
URI: http://192.168.1.2/wml/md8470a.wbmp
Headers
User-Agent: Mozilla/SMB3(z105)/Samsung
Accept: application/vnd.wap.wmlc
Accept: application/vnd.wap.wmlscriptc
Accept: text/vnd.wap.wml
Accept: image/vnd.wap.wbmp
Accept: image/png
Accept: image/jpeg
Accept: image/gif
Accept: application/vnd.oma.dd+xml
Accept: text/vnd.sun.j2me.app-descriptor
Accept: image/bmp
Accept: application/vnd.wap.multipart.mixed
Accept: */*
Accept-Language: English (en)
Accept-Charset: utf-8
X-VODAFONE-3GPPContext: yes
    
```

- *1 : WSP 'PDU Type : Get' : request the content
 URI : path of content (md8470a.wbmp)
 UE uses 'Get' for each contents file(this logic is same as HTTP)
- *2 : WSP > Headers >
 Accept : supported MIME type
 Accept-Language : English
 Accept-Charset : utf-8(common unicode)

Packet Communication (WAP-WML)

Protocol Sequence (WSP Reply)



```

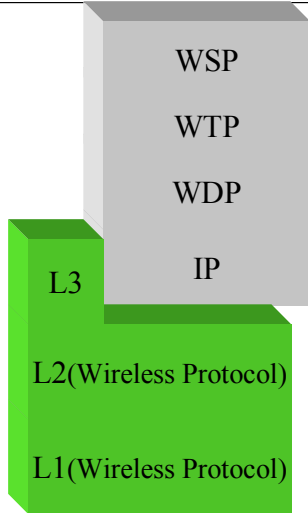
No.  Time      Source      Destination  Protocol  Info
9  1.219860  192.168.1.1  192.168.1.2  WTP+WS  WTP Ack, WSP Get (0x40) http://192.168.1.2/...
10 1.336886  192.168.1.2  192.168.1.1  WTP+WS  WSP Reply (0x04): 200 OK (0x20) image/vnd.wap.wbmp
11 5.619264  192.168.1.1  192.168.1.2  WTP+WS  WTP Ack

Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:01:03:19:13
Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
User Datagram Protocol, Src Port: 9201 (9201), Dst Port: 49152 (49152)
Wireless Transaction Protocol, PDU: Result (2)
0... .. = continue Flag: No TPI
.001 0... = PDU Type: Result (0x02)
.....01 = Trailer Flags: Last packet of message (0x01) *1
.....0 = Re-transmission Indicator: First transmission
1... .. = TID Response: Response
.000 0000 0000 0111 = Transaction ID: 0x0007 *1
Wireless Session Protocol, Method: Reply (0x04), Status: 200 OK (0x20), Content-Type: image/vnd.wap.wbmp
PDU Type: Reply (0x04)
Status: 200 OK (0x20)
Headers Length: 80
Content-Type: image/vnd.wap.wbmp *2
Headers
Server: Microsoft-IIS/5.1
X-Powered-By: ASP.NET
Date: May 24, 2005 12:06:02.000000000
Accept-Ranges: bytes
Last-Modified: Jun 22, 2004 17:30:40.000000000
ETag: "0b9453398c41:acd"
Content-Length: 228 *3
Media Type: image/vnd.wap.wbmp (228 bytes)
    
```

- *1 : WTP 'PDU Type : Result' : reply from WSP (Transaction ID = 7)
- *2 : WSP 'PDU Type : Reply' : reply the content access
 'Status : 200 OK' means success of download content type : .wbmp (Picture data)
- *3 : Content-length : the length of the content

Packet Communication (WAP-WML)

Protocol Sequence (WTP Ack)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|--|------|
| 10 | 1.336886 | 192.168.1.2 | 192.168.1.1 | WTP4WS WSP Reply (0x04): 200 OK (0x20) (image) | |
| 11 | 5.619264 | 192.168.1.1 | 192.168.1.2 | WTP4WS WTP Ack | |
| 12 | 14.358591 | 192.168.1.1 | 192.168.1.2 | WTP4WS WSP Disconnect (0x05) | |

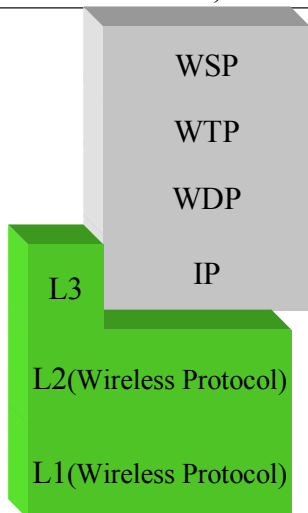

```

Frame 11 (60 bytes on wire, 60 bytes captured)
Ethernet II, Src: 00:00:91:03:19:15, Dst: 00:10:71:00:56:b9
Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
User Datagram Protocol, Src Port: 49152 (49152), Dst Port: 9201 (9201)
Wireless Transaction Protocol, PDU concatenation, PDU count: 2
  Sub PDU size: 3
  Wireless Transaction Protocol, PDU: ACK (3)
    0... .. = Continue Flag: No TPI
    [0001...] = PDU Type: Ack (0x03) *1
    ....0. = TVB/TK Flag: False
    ....00 = Re-transmission Indicator: First transmission
    0... .. = TID Response: Original
    [0000000000000111] = Transaction ID: 0x0007 *1
    
```

*1 : WSP 'PDU Type : Ack' : Ack for the WTP result of the content data(Picture data) (Transaction ID = 7)

Packet Communication (WAP-WML)

Protocol Sequence (WSP Disconnect)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|--|------|
| 10 | 1.336886 | 192.168.1.2 | 192.168.1.1 | WTP4WS WSP Reply (0x04): 200 OK (0x20) (image) | |
| 11 | 5.619264 | 192.168.1.1 | 192.168.1.2 | WTP4WS WTP Ack | |
| 12 | 14.358591 | 192.168.1.1 | 192.168.1.2 | WTP4WS WSP Disconnect (0x05) | |


```

Frame 12 (60 bytes on wire, 60 bytes captured)
Ethernet II, Src: 00:00:91:03:19:15, Dst: 00:10:71:00:56:b9
Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
User Datagram Protocol, Src Port: 49152 (49152), Dst Port: 9201 (9201)
Wireless Transaction Protocol, PDU: Invoke (1), Transaction Class: Unreliable Invoke without Result
  0... .. = Continue Flag: No TPI
  [0001...] = PDU Type: Invoke (0x01) *1
  ....11 = Trailer Flags: Re-assembly not supported (0x03)
  ....00 = Re-transmission Indicator: First transmission
  0... .. = TID Response: Original
  [0000000000001000] = Transaction ID: 0x0008 *1
  00... .. = Version: Current (0x00)
  ..0... = TIDNew: TID is valid
  ...0... = U/P Flag: User Acknowledgement optional
  ....00... = Reserved: 0x00
  ....00 = Transaction Class: Unreliable Invoke without Result (0x00)
  Wireless Session Protocol, Method: Disconnect (0x05), Session ID: 3
  [PDU Type: Disconnect (0x05)] *2
  Server session ID: 3
    
```

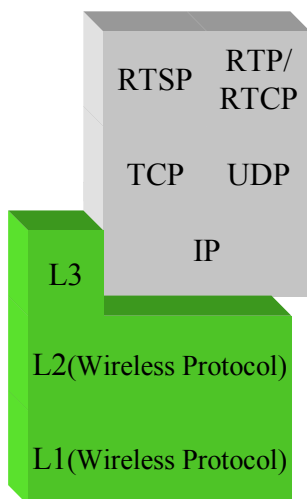
*1 : WTP 'PDU Type : Invoke' : request to WSP (request of 'WSP Disconnect', Transaction ID = 3)

*2 : WSP 'PDU Type : Disconnect' : disconnection

Part 2: Video Streaming

Video Streaming (RTSP)

Protocol Stack



SDP : Session Description Protocol

RTP : Real-time Transfer Protorol

RTCP : Real-time Transfer Control Protocol

RTSP : Real Time Streaming Protocol

UDP : User Datagram Protocol

TCP : Transmission Control Protocol

IP : Internet Protocol

L2(W-CDMA) : MAC / RLC

MAC : Media Access Control

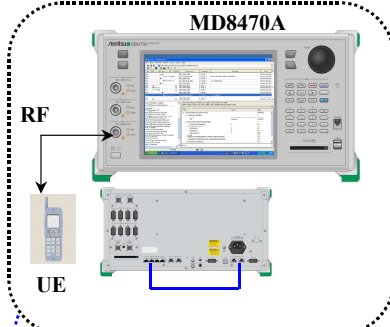
RLC : Radio Link Control

L2(GSM/GPRS) : DL /

DL : Data Link

L1(W / G) : PHY(Physical Layer)

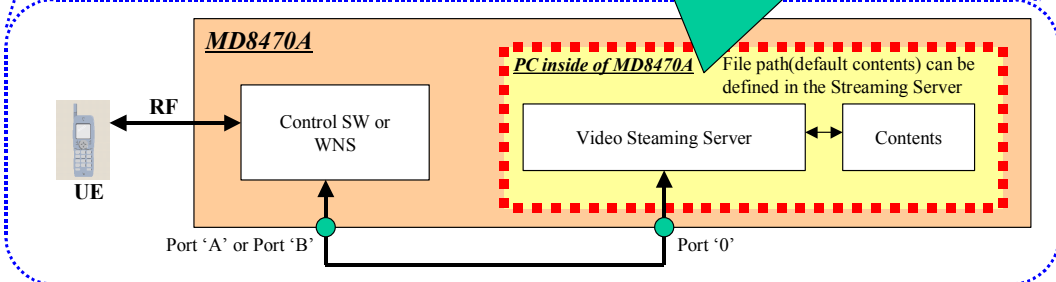
Video Streaming (RTSP)



✓ Video Steaming Server

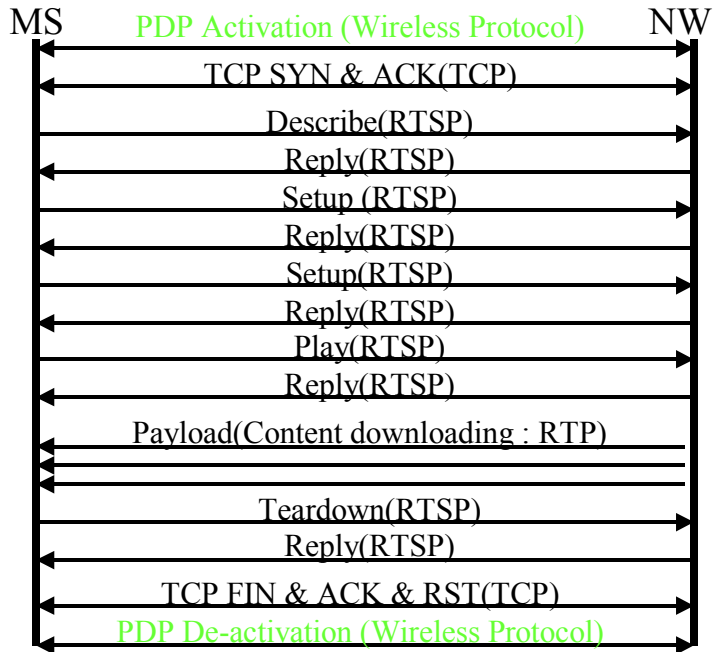
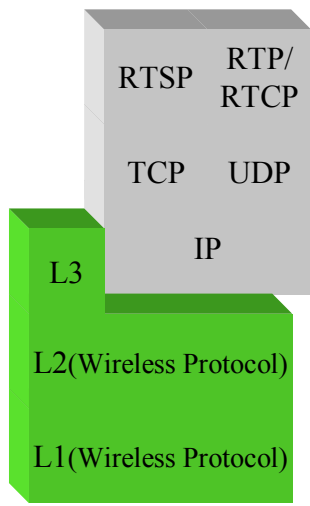
- Supports the video steaming service.
- MD8470A already provides a 'Darwin Video Streaming Server'.

All these parts are available in an external PC



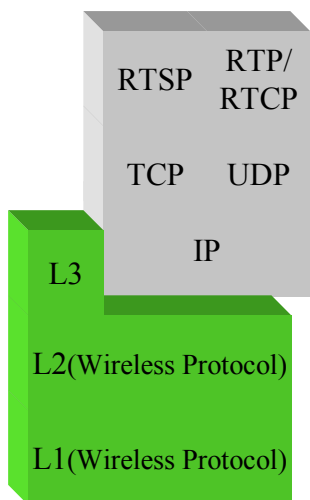
Video Streaming (RTSP)

Protocol Sequence



Video Streaming (RTSP)

Setup

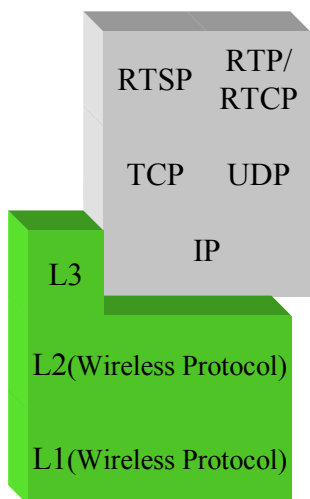


How to install the video streaming Server and how to operate : Please refer to 'DarwinStreamingServerIntoMD8470A.doc'

How to operate video streaming demo on the MD8470A : Please refer to the demonstration note (section1-2)

Video Streaming (RTSP)

Protocol Sequence (TCP SYN & Ack)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 1 | 0.000000 | 192.168.1.1 | 192.168.1.2 | TCP | 6053 > http [SYN] Seq=0 Ack=0 Win=3920 |
| 2 | 0.000053 | 192.168.1.2 | 192.168.1.1 | TCP | http > 6053 [SYN, ACK] Seq=0 Ack=1 Win= |
| 3 | 0.259956 | 192.168.1.1 | 192.168.1.2 | TCP | 6053 > http [ACK] Seq=1 Ack=1 Win=3920 |
| 4 | 0.320715 | 192.168.1.1 | 192.168.1.2 | HTTP | GET / HTTP/1.1 |
| 5 | 0.321380 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (text/html) |
| 6 | 0.939905 | 192.168.1.1 | 192.168.1.2 | TCP | 6053 > http [ACK] Seq=518 Ack=924 Win= |
| 7 | 5.939577 | 192.168.1.1 | 192.168.1.2 | TCP | [TCP Dup ACK 6#1] 6053 > http [ACK] Se |
| 8 | 10.339245 | 192.168.1.1 | 192.168.1.2 | TCP | 6053 > http [RST] Seq=518 Ack=924 Win=0 |
| 9 | 17.378668 | 192.168.1.1 | 192.168.1.2 | TCP | 6479 > 554 [SYN] Seq=0 Ack=0 Win=39200 |
| 10 | 17.378745 | 192.168.1.2 | 192.168.1.1 | TCP | 554 > 6479 [SYN, ACK] Seq=0 Ack=1 Win= |
| 11 | 17.638649 | 192.168.1.1 | 192.168.1.2 | TCP | 6479 > 554 [ACK] Seq=1 Ack=1 Win=39200 |
| 12 | 17.638882 | 192.168.1.2 | 192.168.1.1 | TCP | [TCP Dup ACK 10#1] 554 > 6479 [ACK] Se |
| 13 | 17.699105 | 192.168.1.1 | 192.168.1.2 | RTSP | DESCRIBE rtsp://192.168.1.2/4.sdp RTSP |

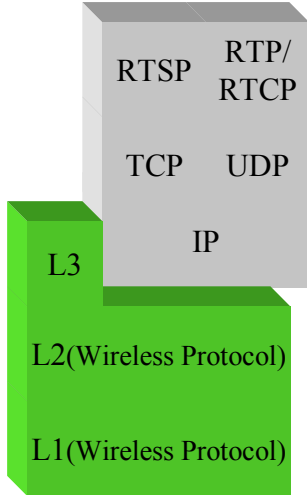
*1 : HTTP content download (Please refer the part 1 'HTTP' in detail), at first UE accesses the default.htm content, then links to the streaming content in this example. (No.1 to No.5)

*2 : When UE accesses the streaming content link, at first UE disconnects TCP and wireless connection (No.8)

*3 : then activates TCP and wireless session again. (No.9 to No.12)

Video Streaming (RTSP)

Protocol Sequence (RTSP DESCRIBE)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 12 | 17.638982 | 192.168.1.2 | 192.168.1.1 | TCP | [TCP Dup ACK 10#] 554 > 6479 [ACK] Seq=1414 |
| 13 | 17.699105 | 192.168.1.1 | 192.168.1.2 | RTSP | DESCRIBE rtsp://192.168.1.2/v.sdp RTSP/1.0 |
| 14 | 17.701257 | 192.168.1.2 | 192.168.1.1 | RTSP/S | Reply: RTSP/1.0 200 OK, with session id |


```

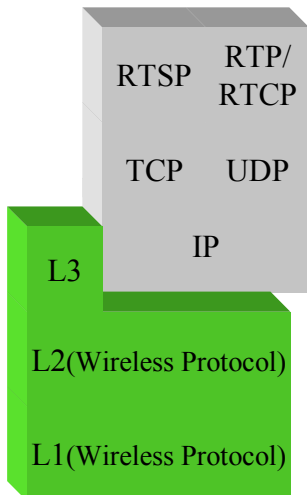
Frame 13 (338 bytes on wire (338 bytes captured)
  Ethernet II, Src: 00:00:91:03:19:15, Dst: 00:10:71:00:56:b9
  Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
  Transmission Control Protocol, Src Port: 6479 (6479), Dst Port: 554 (554), Seq: 1, Ack: 1, Len: 284
  Real Time Streaming Protocol
    DESCRIBE rtsp://192.168.1.2/v.sdp RTSP/1.0\r\n
      Method: DESCRIBE
      URL: rtsp://192.168.1.2/v.sdp *1
      CSeq: 0\r\n *2
      User-Agent: PVPlayer2.4\r\n
      Accept: application/sdp\r\n
      x-wap-profile: http://wap.samsungmobile.com/uaprof/Z105UAProf.rdf\r\n
      User-Network: UNKNOWN\r\n
      DeviceInfo: MANUF=SAMSUNG;PROC=ARM;MEM=16MB;OS=PSOS;DISPLAY=\r\n
      Timestamp: 253897.343750\r\n
      \r\n
  
```

*1 : 'URL', we can confirm the URL (same as streaming Server setting), 'DESCRIBE' means requesting the detail content information. Server will reply to this request with SDP.

*2 : 'C Seq' is the RTSP's sequence number. ('C Seq = 0')

Video Streaming (RTSP)

Protocol Sequence (RTSP Reply)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 13 | 17.699105 | 192.168.1.1 | 192.168.1.2 | RTSP | DESCRIBE rtsp://192.168.1.2/v.sdp RTSP/1.0 |
| 14 | 17.701257 | 192.168.1.2 | 192.168.1.1 | RTSP/S | Reply: RTSP/1.0 200 OK, with session id |
| 15 | 18.218946 | 192.168.1.1 | 192.168.1.2 | RTSP | SETUP rtsp://192.168.1.2/v.sdp/554 ID |


```

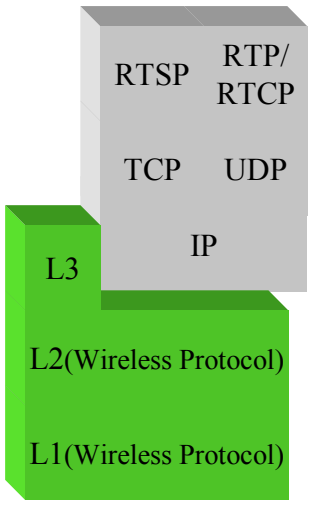
Frame 14 (1414 bytes on wire (1414 bytes captured)
  Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:91:03:19:15
  Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
  Transmission Control Protocol, Src Port: 554 (554), Dst Port: 6479 (6479), Seq: 1, Ack: 285, Len: 284
  Real Time Streaming Protocol
    RTSP/1.0 200 OK\r\n *1
      Status: 200
      Server: BSS/5.0.3.2 (Build/452.22.3; Platform/win32; Release/Panther; update/3GPP); \r\n
      CSeq: 0\r\n
      Cache-Control: no-cache\r\n
      Content-Length: 988\r\n
      Date: Tue, 24 May 2005 06:55:23 GMT\r\n *2
      Expires: Tue, 24 May 2005 06:55:23 GMT\r\n
      Content-type: application/sdp\r\n
      x-Accept-retransmit: our-retransmit\r\n
      x-Accept-dynamic-rate: 1\r\n
      Content-Base: rtsp://192.168.1.2/v.sdp\r\n
      \r\n
    Session Description Protocol
      Session Description Protocol Version (v): 0
      Owner/Creator, session id (o): QTSS_Play_List 216638092 216656518 IN IP4 192.168.1.2
      Session Name (s): C:\Program Files\Darwin Streaming Server\Playlists\VodafoneVoda0
      Connection Information (c): IN IP4 0.0.0.0
      Bandwidth information (b): AS64
      Time Description, active time (t): 0 0
  
```

*1 : '200 OK' means RTSP reply

*2 : 'C Seq' is the RTSP's sequence number. (C Seq = 0)
we can confirm Streaming Server version
application – type : sdp

Video Streaming (RTSP)

Protocol Sequence (RTSP Reply)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 13 | 17.699105 | 192.168.1.1 | 192.168.1.2 | RTSP | DESCRIBE rtsp://192.168.1.2/v.sdp RTSP |
| 14 | 17.701257 | 192.168.1.2 | 192.168.1.1 | RTSP | S/Reply: RTSP/1.0 200 OK, with session d |
| 15 | 18.218946 | 192.168.1.1 | 192.168.1.2 | RTSP | SETUP rtsp://192.168.1.2/v.sdp/trackID |
| 16 | 18.219796 | 192.168.1.2 | 192.168.1.1 | RTSP | Reply: RTSP/1.0 200 OK |


```

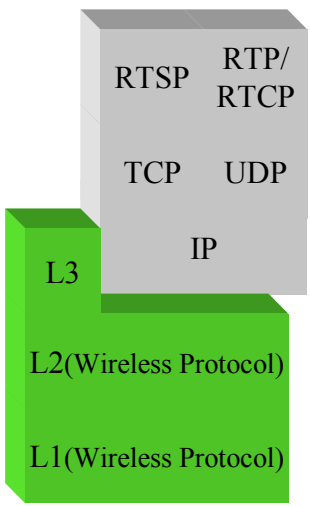
Content-Base: rtsp://192.168.1.2/v.sdp/\r\n
\r\n
Session Description Protocol
Session Description Protocol version (v): 0
Owner/Creator, Session id (o): QTSS_Play_List 216638092 216656518 IN IP4 192.168.1.2
Session Name (s): C:\Program Files\Darwin Streaming Server\Playlists\VodafoneVoda8
Connection Information (c): IN IP4 0.0.0.0
Bandwidth Information (b): AS:64
Time Description, active time (t): 0 0
Session Attribute (a): X-broadcastcontrol:RTSP
Session Attribute (a): mpeg4-fod:"data:application/mpeg4-fod;base64,Ao1bAE///8I/WOBKGABQIXkYXrh
Session Attribute (a): isma-compliance:1,1,0,1
Session Attribute (a): control:*
Media Description, name and address (m): video 0 RTP/AVP 96
Bandwidth Information (b): AS:51
Media Attribute (a): rtpmap:96 MP4V-ES/90000
Media Attribute (a): control:trackID=1
Media Attribute (a): cliprect:0,0,144,176
Media Attribute (a): fmtp:96 profile-level-1-id=1; config=000001B008000001B50EE040C0CF0000010000000
Media Attribute (a): mpeg4-esid:201
Media Description, name and address (m): audio 0 RTP/AVP 97
Bandwidth Information (b): AS:13
Media Attribute (a): rtpmap:97 AMR/8000/1
Media Attribute (a): control:trackID=2
Media Attribute (a): fmtp:97 octet-align
    
```

*1 : There are 2 sessions. One is for video; the other is for voice. 'Video' is MP4 and 'Track ID = 1'.

*2 : This is for voice. 'Voice' is 'AMR', and 'Track ID = 2'

Video Streaming (RTSP)

Protocol Sequence (RTSP SETUP)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|---------------|--|
| 14 | 17.701257 | 192.168.1.2 | 192.168.1.1 | RTSP/S Reply: | RTSP/1.0 200 OK, with session d |
| 15 | 18.218946 | 192.168.1.1 | 192.168.1.2 | RTSP | SETUP rtsp://192.168.1.2/v.sdp/trackID |
| 16 | 18.219796 | 192.168.1.2 | 192.168.1.1 | RTSP | Reply: RTSP/1.0 200 OK |


```

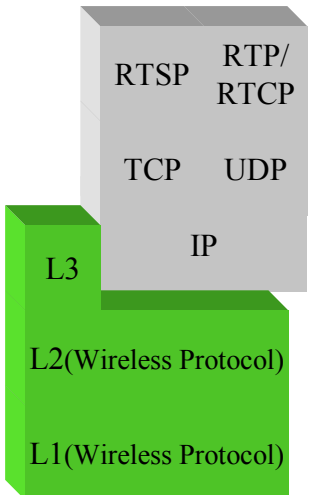
Frame 15 (294 bytes on wire, 294 bytes captured)
Ethernet II, Src: 00:00:91:103:19:15, Dst: 00:10:71:100:56:b9
Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
Transmission Control Protocol, Src Port: 6479 (6479), Dst Port: 554 (554), Seq: 285, Ack: 1361, Len:
Real Time Streaming Protocol
SETUP rtsp://192.168.1.2/v.sdp/trackID=2 RTP/1.0\r\n
Method: SETUP
URL: rtsp://192.168.1.2/v.sdp/trackID=2
CSeq: 1\r\n
User-Agent: PVPlayer 3.4\r\n
Transport: RTP/AVP;unicast;client_port=7000-7001\r\n
Blocksize: 1400\r\n
User-Network: UNKNOWN\r\n
DeviceInfo: MANUF=SAMSUNG;PROC=ARM;MEM=16MB;OS=PSOS;DISPLAY=\r\n
\r\n
    
```

*1 : This is the 'SETUP' for the 'Track ID = 2(voice)'.

*2 : In the 'SETUP', Client informs the 'available port number'. And 'C Seq = 1'. And requests the Server side port number.

Video Streaming (RTSP)

Protocol Sequence (RTSP Reply)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 15 | 18.218946 | 192.168.1.1 | 192.168.1.2 | RTSP | SETUP rtsp://192.168.1.2/v.sdp/trackID=2 |
| 16 | 18.219756 | 192.168.1.2 | 192.168.1.1 | RTSP | Reply: RTSP/1.0 200 OK |
| 17 | 18.578996 | 192.168.1.1 | 192.168.1.2 | RTSP | SETUP rtsp://192.168.1.2/v.sdp/trackID=1 |

```

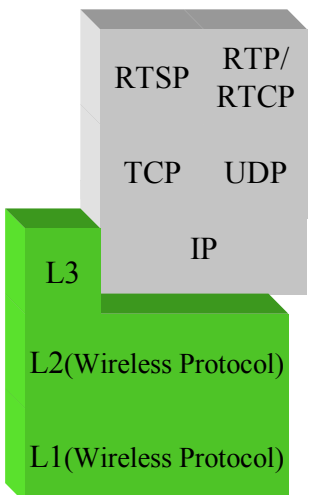
# Frame 16 (386 bytes on wire, 386 bytes captured)
# Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:91:03:19:15
# Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
# Transmission Control Protocol, Src Port: 554 (554), Dst Port: 6479 (6479), Seq: 1361, Ack: 525, Len: 386
# Real Time Streaming Protocol
# RTSP/1.0 200 OK
# Status: 200
# Server: DSS/5.0.3.2 (Build/452.22.3; Platform/win32; Release/Panther; Update/3GPP; )\r\n
# CSeq: 1\r\n
# Cache-Control: no-cache\r\n
# Session: 89361089572884\r\n
# Session: 89361089572884\r\n
# Date: Tue, 24 May 2005 06:55:23 GMT\r\n
# Expires: Tue, 24 May 2005 06:55:23 GMT\r\n
# Transport: RTP/AVP;unicast;source=192.168.1.2;client_port=7000-7001;server_port=6970-6971\r\n
\r\n
    
```

*1 : This is the 'Reply' of 'SETUP(Track ID = 2 : voice)'. 'RTSP/1.0 200 OK' means 'Reply'

*2 : 'C Seq = 1', and Server informs the session number for the 'Track ID = 2', and also informs the port number (Server side : 6970-6971, Client 7000-7001).

Video Streaming (RTSP)

Protocol Sequence (RTSP SETUP)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 16 | 18.219756 | 192.168.1.2 | 192.168.1.1 | RTSP | Reply: RTSP/1.0 200 OK |
| 17 | 18.578996 | 192.168.1.1 | 192.168.1.2 | RTSP | SETUP rtsp://192.168.1.2/v.sdp/trackID=1 |
| 18 | 18.579832 | 192.168.1.2 | 192.168.1.1 | RTSP | Reply: RTSP/1.0 200 OK |

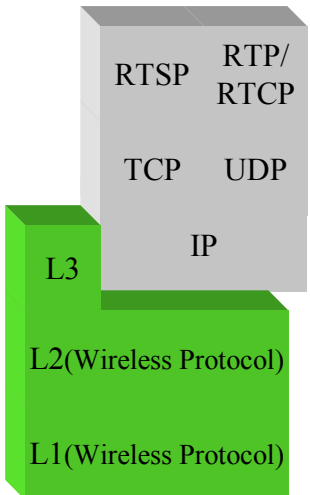
```

# Frame 17 (319 bytes on wire, 319 bytes captured)
# Ethernet II, Src: 00:00:91:03:19:15, Dst: 00:10:71:00:56:b9
# Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
# Transmission Control Protocol, Src Port: 6479 (6479), Dst Port: 554 (554), Seq: 525, Ack: 1693, Len: 319
# Real Time Streaming Protocol
# SETUP rtsp://192.168.1.2/v.sdp/trackID=1 RTSP/1.0\r\n
# Method: SETUP
# URL: rtsp://192.168.1.2/v.sdp/trackID=1
# CSeq: 2\r\n
# Session: 89361089572884\r\n
# Session: 89361089572884\r\n
# User-Agent: PVPlayer 3.4\r\n
# Transport: RTP/AVP;unicast;client_port=7002-7003\r\n
# BlocksSize: 1400\r\n
# User-Agent: UNKNOWN\r\n
# DeviceInfo: MANUF=SAMSUNG;PROC=ARM;MEM=16MB;OS=PSOS;DISPLAY=\r\n
\r\n
    
```

*1 : This is the 'SETUP' for the 'Track ID = 1(video)'.
*2 : In the 'SETUP', Client informs the 'available port number'. And 'C Seq = 2'. And requests the Server side port number.

Video Streaming (RTSP)

Protocol Sequence (RTSP Reply)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 17 | 18.578966 | 192.168.1.1 | 192.168.1.2 | RTSP | SETUP rtsp://192.168.1.2/v.sdp/trackID=1 RT |
| 18 | 18.578992 | 192.168.1.2 | 192.168.1.1 | RTSP | Reply: RTSP/1.0 200 OK |
| 19 | 19.178528 | 192.168.1.1 | 192.168.1.2 | TCP | 6479 > 554 [ACK] Seq=790 Ack=2025 win=39200 |

```

Frame 18 (386 bytes on wire, 386 bytes captured)
  Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:01:03:19:15
  Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
  Transmission Control Protocol, Src Port: 554 (554), Dst Port: 6479 (6479), Seq: 1693, Ack: 790, Len: 332
  Real Time Streaming Protocol
    RTSP/1.0 200 OK\r\n
    Status: 200 *1
    Server: SSS/5.0.3.2 (Build/452.22.3; Platform/Win32; Release/Panther; Update/3GPP1 )\r\n
    Cseq: 2\r\n
    Session: 89361089572884\r\n
    Session: 89361089572884
    Cache-Control: no-cache\r\n
    Date: Tue, 24 May 2005 06:55:23 GMT\r\n
    Expires: Tue, 24 May 2005 06:55:23 GMT\r\n
    Transport: RTP/AVP;unicast;source=192.168.1.2;client_port=7002-7003;server_port=6970-6971\r\n
    \r\n
  
```

- *1 : This is the 'Reply' of 'SETUP(Track ID = 1 : video)'. 'RTSP/1.0 200 OK' means 'Reply'
- *2 : 'C Seq = 2', and Server informs the session number for the 'Track ID = 2', and also informs the port number (Server side : 6970-6971, Client 7002-7003).

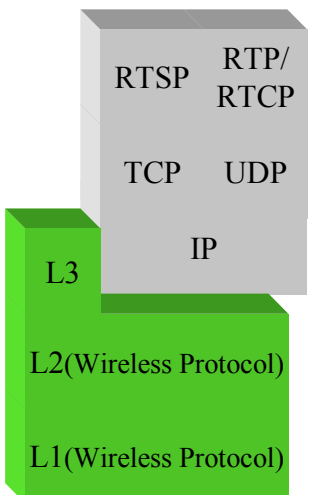
Discover What's Possible™
MD8470A-E-E-1

Slide 59



Video Streaming (RTSP)

Protocol Sequence (RTSP PLAY)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 20 | 19.238721 | 192.168.1.1 | 192.168.1.2 | RTSP | PLAY rtsp://192.168.1.2/v.sdp RTSP/1.0 |
| 21 | 19.239450 | 192.168.1.2 | 192.168.1.1 | RTSP | Reply: RTSP/1.0 200 OK |

```

Frame 20 (152 bytes on wire, 152 bytes captured)
  Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:10:71:00:56:b9
  Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
  Transmission Control Protocol, Src Port: 6479 (6479), Dst Port: 554 (554), Seq: 790, Ack: 2025, Len: 98
  Real Time Streaming Protocol
    PLAY rtsp://192.168.1.2/v.sdp RTSP/1.0\r\n
    Method: PLAY *1
    URL: RTSP://192.168.1.2/v.sdp
    CSeq: 3\r\n
    Session: 89361089572884\r\n
    Session: 89361089572884
    Range: npr=0:00000-\r\n
    \r\n
  
```

- *1 : 'PLAY' means 'start the video streaming content'.
- *2 : 'C Seq = 3'. And 'PLAY' with added session number

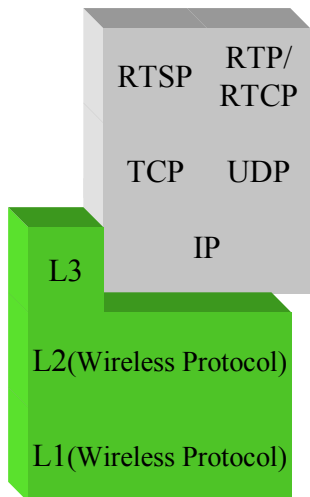
Discover What's Possible™
MD8470A-E-E-1

Slide 60



Video Streaming (RTSP)

Protocol Sequence (RTSP Reply)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 20 | 19.238721 | 192.168.1.1 | 192.168.1.2 | RTSP | PLAY rtsp://192.168.1.2/v.sdp RTSP/1.0 |
| 21 | 19.239450 | 192.168.1.2 | 192.168.1.1 | RTSP | Reply: RTSP/1.0 200 OK |
| 22 | 19.239967 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=unknown (96), SSRC=216656599, |

```

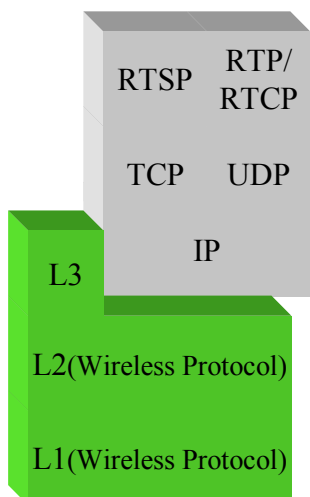
Frame 21 (299 bytes on wire, 299 bytes captured)
Ethernet II, Src: 00:10:72:100:56:b9, Dst: 00:00:91:03:19:15
Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
Transmission Control Protocol, Src Port: 554 (554), Dst Port: 6479 (6479), Seq: 2025, Ack: 888, Len: 245
Real Time Streaming Protocol
  RTSP/1.0 200 OK
    Status: 200
    Server: OSS/5.0.3.2 (Build/452.22.3; Platform/win32; Release/Panther; Update/3GPP; )\n\n
    CSeq: 3\n\n
    Session: 89361089572884\n\n
    Range: npt=now-\n\n
    RTP-Info: url=rtsp://192.168.1.2/v.sdp/trackID=2,url=rtsp://192.168.1.2/v.sdp/trackID=1\n\n
  \n\n
  
```

*1 : This is the 'Reply' of 'PLAY'. 'RTSP/1.0 200 OK' means 'Reply(Start the video streaming)'

*2 : 'C Seq = 3', 'PLAY' with added session number and both 'Track ID = 1' and 'Track ID = 2'.

Video Streaming (RTSP)

Protocol Sequence (RTP Payload)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 20 | 19.238721 | 192.168.1.1 | 192.168.1.2 | RTSP | PLAY rtsp://192.168.1.2/v.sdp RTSP/1.0 |
| 21 | 19.239450 | 192.168.1.2 | 192.168.1.1 | RTSP | Reply: RTSP/1.0 200 OK |
| 22 | 19.239967 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=unknown (96), SSRC=216656599, |

```

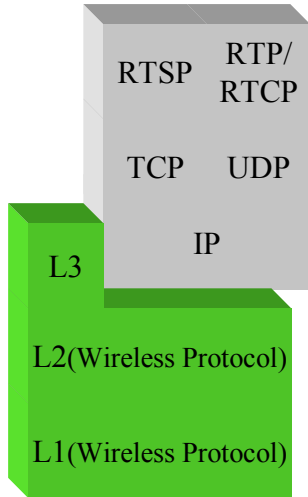
Frame 22 (629 bytes on wire, 629 bytes captured)
Ethernet II, Src: 00:10:72:100:56:b9, Dst: 00:00:91:03:19:15
Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
User Datagram Protocol, Src Port: 6970 (6970), Dst Port: 7002 (7002)
Real Time Transport Protocol
  [Stream setup by RTSP (Frame 18)]
  [Setup Frame: 18]
  [Setup Method: RTSP]
  10. .... = Version: RFC 1889 Version
  ..0. .... = Padding: False
  ...0 .... = Extensions: False
  ... 0000 = Contributing source identifiers count: 0
  1... .... = Marker: True
  Payload type: unknown (96)
  Sequence number: 40838
  Timestamp: 863397351
  Synchronization source identifier: 216656599
  Payload: 00001b654330a918014389424a830a8f84256a818f8666...
  
```

*1 : The video streaming content is downloaded with 'RTP' protocol.

*2 : 'frame 18' means 'Line No. 18 (SETUP for Track ID = 1 : video). You can confirm this (for Track ID = 1) with the port number information (7002).

Video Streaming (RTSP)

Protocol Sequence (RTP Payload)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 22 | 19.233997 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (96), SSRC=216656599 |
| 23 | 19.240053 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (96), SSRC=216656599 |
| 24 | 19.247155 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (97), SSRC=216644310 |


```

# Frame 24 (215 bytes on wire, 215 bytes captured)
# Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:91:03:19:15
# Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
# User Datagram Protocol, Src Port: 6970 (6970), Dst Port: 7000 (7000)
# Real-Time Transport Protocol
  # [Stream setup by RTSP (Frame 16)]
    # [Setup frame: 16]
    # [Setup Method: RTSP]
    10... .. = Version: RFC 1889 Version
    ..0... .. = Padding: False
    ....0000 = Contributing source identifiers count: 0
    0... .. = Marker: False
    Payload type: Unknown (97)
    Sequence number: 62514
    Timestamp: 76774278
    Synchronization source identifier: 216644310
    Payload: F08CBCCBC3C28967AF997DE39E1E015F10546071515427...
    
```

*1 : The video streaming content is downloaded with 'RTP' protocol.

*2 : 'frame 18' means 'Line No. 16 (SETUP for Track ID = 2 : video). You can confirm this (for Track ID = 2) with the port number information (7000).

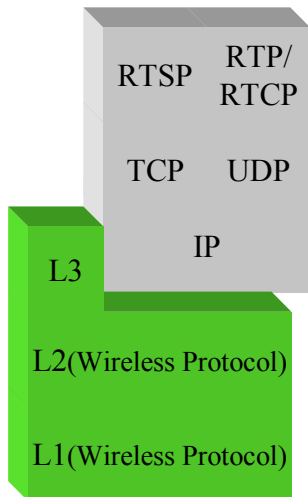
Discover What's Possible™
MD8470A-E-E-1

Slide 63

Anritsu

Video Streaming (RTSP)

Protocol Sequence (RTP Payload)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 58 | 20.437853 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (96), SSRC=216656599 |
| 59 | 20.524294 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (97), SSRC=216644310 |
| 60 | 20.524636 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (96), SSRC=216656599 |
| 61 | 20.590545 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (96), SSRC=216656599 |
| 62 | 20.623755 | 192.168.1.2 | 192.168.1.1 | RTCP | Sender Report *1 |
| 63 | 20.624034 | 192.168.1.2 | 192.168.1.1 | RTCP | Sender Report *1 |
| 64 | 20.624250 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (97), SSRC=216644310 |
| 65 | 20.636801 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (96), SSRC=216656599 |
| 66 | 20.723920 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (97), SSRC=216644310 |
| 67 | 20.724260 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (96), SSRC=216656599 |
| 68 | 20.799470 | 192.168.1.1 | 192.168.1.2 | RTCP | Receiver Report *2 |
| 69 | 20.802161 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (96), SSRC=216656599 |
| 70 | 20.818462 | 192.168.1.1 | 192.168.1.2 | RTCP | Receiver Report *2 |
| 71 | 20.824380 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (97), SSRC=216644310 |
| 72 | 20.857384 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (96), SSRC=216656599 |
| 73 | 20.924547 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (97), SSRC=216644310 |
| 74 | 20.924891 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (96), SSRC=216656599 |
| 75 | 20.989794 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (96), SSRC=216656599 |
| 76 | 21.023871 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (97), SSRC=216644310 |
| 77 | 21.057211 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (96), SSRC=216656599 |
| 78 | 21.124192 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (97), SSRC=216644310 |
| 79 | 21.124535 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (96), SSRC=216656599 |
| 80 | 21.190416 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (96), SSRC=216656599 |
| 81 | 21.190511 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (96), SSRC=216656599 |
| 82 | 21.190546 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=Unknown (96), SSRC=216656599 |

*1 : sometimes there are some 'Sender Reports'.

*2 : sometimes there are some 'Receiver Reports'.

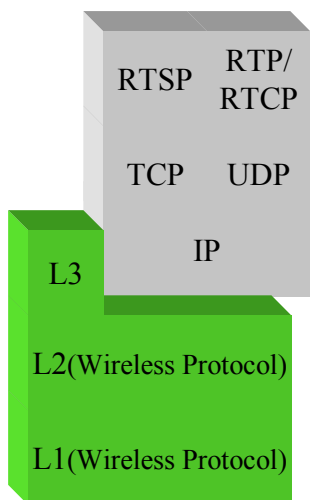
Discover What's Possible™
MD8470A-E-E-1

Slide 64

Anritsu

Video Streaming (RTSP)

Protocol Sequence (RTSP TEARDOWN)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 937 | 50.818469 | 192.168.1.2 | 192.168.1.1 | RTP | Payload type=unknown (96), SSRC=216656599, |
| 938 | 50.836275 | 192.168.1.1 | 192.168.1.2 | RTSP | TEARDOWN rtsp://192.168.1.2/v.sdp RTSP/1.0 |
| 939 | 50.836978 | 192.168.1.2 | 192.168.1.1 | RTSP | Reply: RTSP/1.0 200 OK |


```

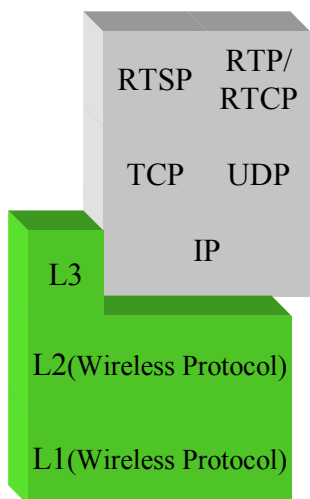
# Frame 938 (153 bytes on wire, 153 bytes captured)
# Ethernet II, Src: 00:00:91:03:19:15, Dst: 00:10:71:00:56:b9
# Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
# Transmission Control Protocol, Src Port: 6479 (6479), Dst Port: 554 (554), Seq: 888, Ack: 2270, Len: 99
# Real Time Streaming Protocol
  # TEARDOWN rtsp://192.168.1.2/v.sdp RTSP/1.0\r\n
    Method: TEARDOWN
    URL: rtsp://192.168.1.2/v.sdp
    CSeq: 4\r\n
    Session: 89361089572884\r\n
    Session: 89361089572884
    Connection: close\r\n
  \r\n
  
```

*1 : 'TEARDOWN' means 'Request to stop the video streaming'.

*2 : 'C Seq = 4'. And 'TEARDOWN' with added session number

Video Streaming (RTSP)

Protocol Sequence (RTSP Reply)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 938 | 50.836275 | 192.168.1.1 | 192.168.1.2 | RTSP | TEARDOWN rtsp://192.168.1.2/v.sdp RTSP/1.0 |
| 939 | 50.836978 | 192.168.1.2 | 192.168.1.1 | RTSP | Reply: RTSP/1.0 200 OK |
| 940 | 50.837097 | 192.168.1.2 | 192.168.1.1 | TCP | 554 > 6479 [FIN, ACK] Seq=2428 Ack=987 win= |


```

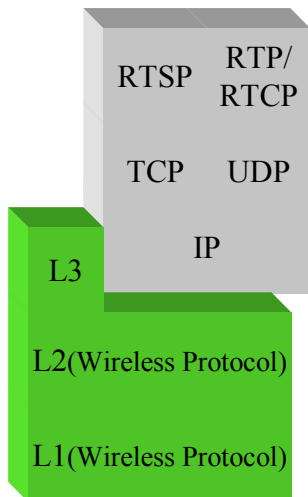
# Frame 939 (212 bytes on wire, 212 bytes captured)
# Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:91:03:19:15
# Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
# Transmission Control Protocol, Src Port: 554 (554), Dst Port: 6479 (6479), Seq: 2270, Ack: 987, Len: 158
# Real Time Streaming Protocol
  # RTSP/1.0 200 OK\r\n
    Status: 200
    Server: DSS/5.0.3.2 (Bu11d/452.22.3; Platform/WIn32; Release/Panther; update/3GPP; )\r\n
    CSeq: 4\r\n
    Session: 89361089572884\r\n
    Session: 89361089572884
    Connection: close\r\n
  \r\n
  
```

*1 : This is the 'Reply' of 'TEARDOWN'. 'RTSP/1.0 200 OK' means 'Reply(Stop video streaming)'

*2 : 'C Seq = 4', 'TEARDOWN' with added session number.

Video Streaming (RTSP)

Protocol Sequence (TCP FIN & Ack & RST)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 939 | 50.886978 | 192.168.1.2 | 192.168.1.1 | RTSP | Reply: RTSP/1.0 200 OK |
| 940 | 50.897097 | 192.168.1.2 | 192.168.1.1 | TCP | 554 > 6479 [FIN, ACK] Seq=2428 Ack=987 wfin= |
| 941 | 53.311627 | 192.168.1.2 | 192.168.1.1 | RTSP | Reply: RTSP/1.0 200 OK |
| 942 | 53.555953 | 192.168.1.1 | 192.168.1.2 | TCP | 6479 > 554 [FIN, ACK] Seq=987 Ack=2270 wfin= |
| 943 | 53.555987 | 192.168.1.2 | 192.168.1.1 | TCP | 554 > 6479 [ACK] Seq=2428 Ack=988 wfin=31782 |
| 944 | 53.573923 | 192.168.1.1 | 192.168.1.2 | TCP | 6479 > 554 [RST] Seq=988 Ack=3370548527 wfin= |

```

# Frame 944 (60 bytes on wire, 60 bytes captured)
# Ethernet II, Src: 00:00:91:03:19:15, Dst: 00:10:71:00:56:b9
# Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
# Transmission Control Protocol, Src Port: 6479 (6479), Dst Port: 554 (554), Seq: 988, Ack: 3370548527, Len:
  Source port: 6479 (6479)
  Destination port: 554 (554)
  Sequence number: 988 (relative sequence number)
  Header length: 20 bytes
  # Flags: 0x0004 (RST)
    0... .. = Congestion window reduced (CWR): Not set
    ..0. .... = ECN-Echo: Not set
    ...0. .... = Urgent: Not set
    ...0. .... = Acknowledgment: Not set
    ....0. .... = Push: Not set
    ....1.. = Reset: Set
    .....0. = Syn: Not set
    ....0. = Fin: Not set
    Window size: 39200
    Checksum: 0x12df (correct)
    
```

*Line940 : FIN means 'request to disconnect', and Ack means 'Ack for previous TEARDOWN'. (UE to Server)

*Line941 : Retransmit of Line 939?

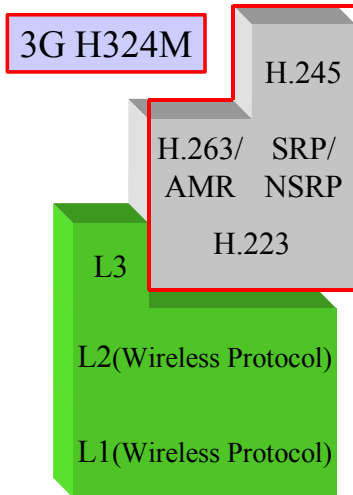
*Line942 : FIN means 'request to disconnect', and Ack means 'Ack for previous 'Reply of TEARDOWN' (Server to UE)

*Line943,944 : Ack for previous TCP and RST(disconnect)

Part 3: Video Telephony

Video Telephony

Protocol Stack



3G H324M : 3G 324M consist of 'H.245' and 'H.263/AMR' (Multiplexing in H.223)

H.245 : Protocol for the 'control' part of video telephony

H.263 : Protocol for the 'video' part of the video telephony

AMR : Protocol for the 'Voice' part of the video telephony

H.223 : Protocol for multiplexing 'H.245' and 'H.263/AMR'.

L2(W-CDMA) : MAC / RLC

MAC : Media Access Control

RLC : Radio Link Control

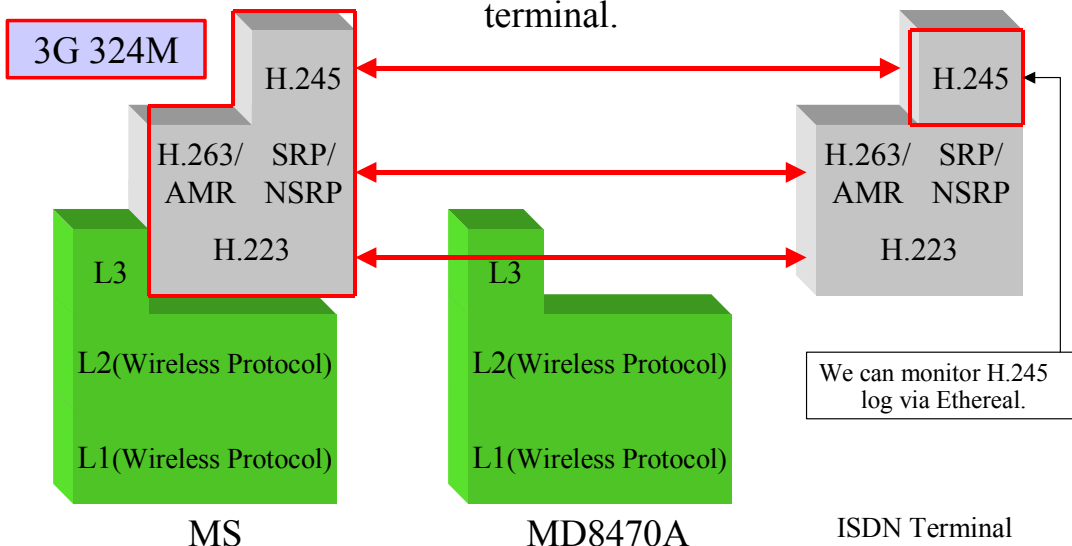
L2(GSM/GPRS) : DL /

DL : Data Link

L1(W / G) : PHY(Physical Layer)

Video Telephony

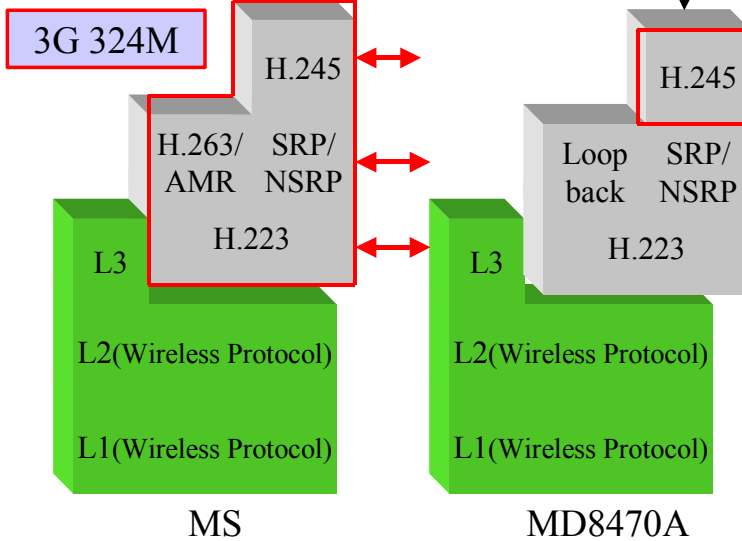
ISDN Terminal Method



In this method, MD8470A sends through all 3G 324M data to ISDN terminal.

Video Telephony

Loopback Method

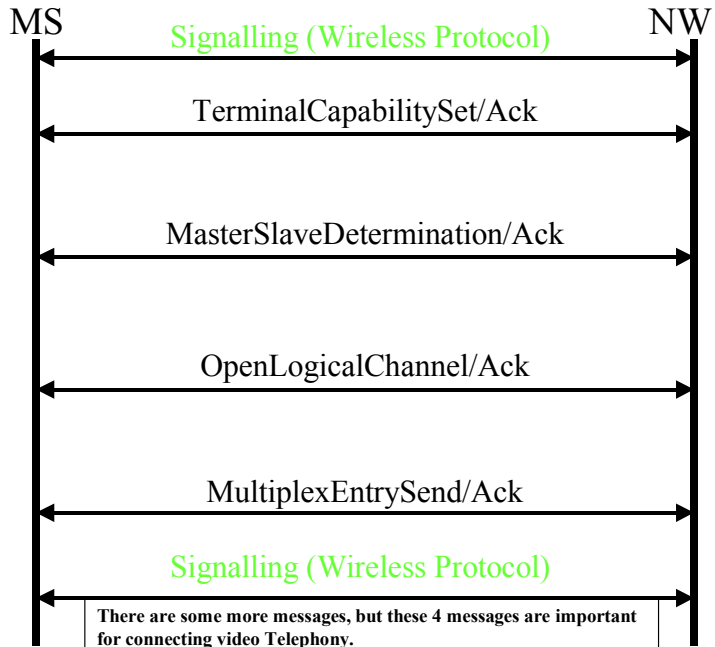
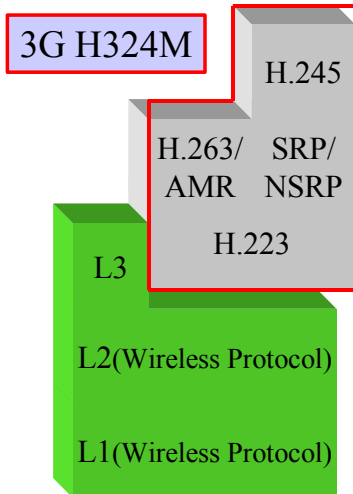


In this method, MD8470A has the following functions:

1. MD8470A's H.245 sends a response for the command from MS (always waiting for MS command)
2. The content of MD8470A's response message is just copying from the command message from MS
3. SRP is used in the case of 'Command'; NSRP is used in the case of 'Response'.

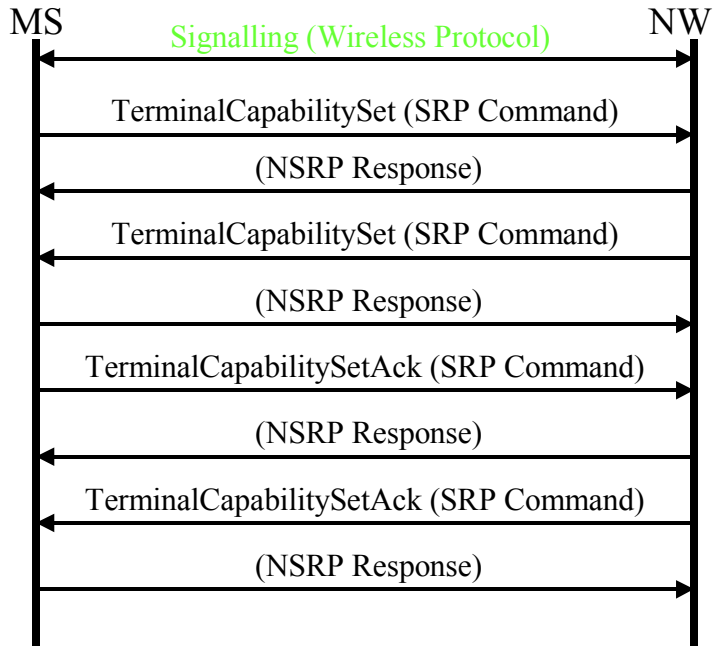
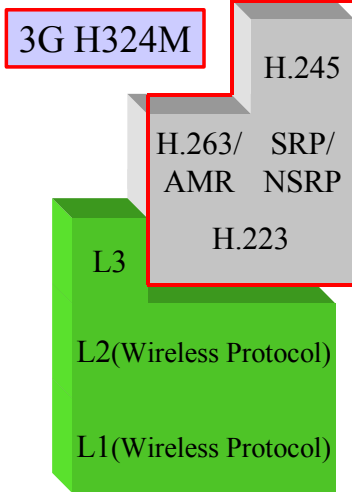
Video Telephony

General Protocol Sequence



Video Telephony

Protocol Sequence
(TerminalCapabilitySet)



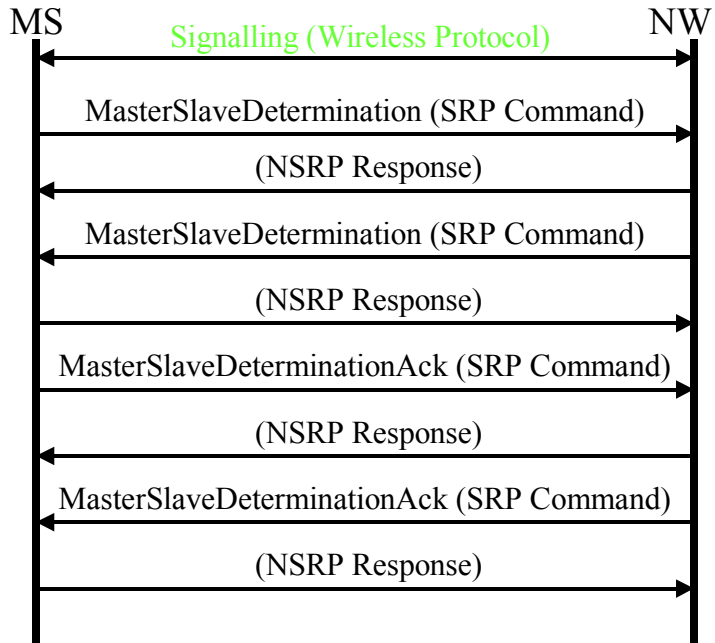
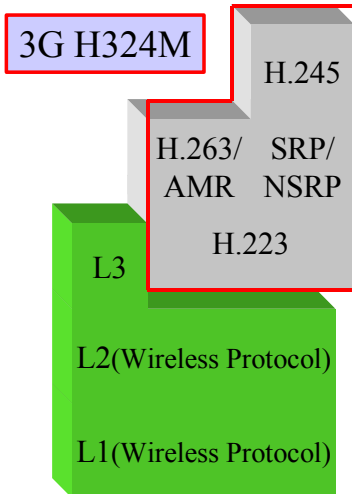
Discover What's Possible™
MD8470A-E-E-1

Slide 73

Anritsu

Video Telephony

Protocol Sequence
(MasterSlaveDetermination)



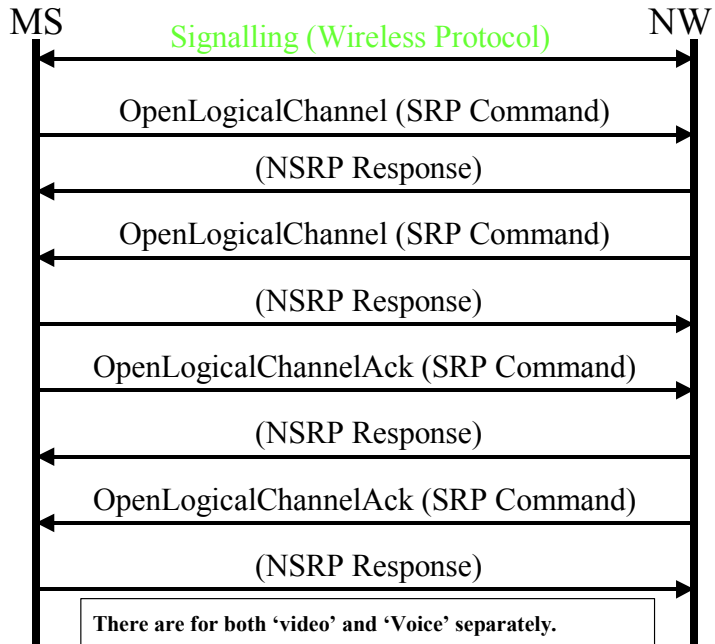
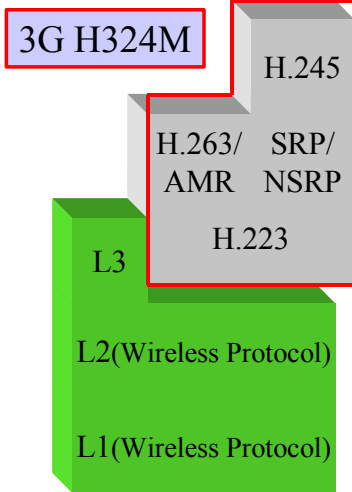
Discover What's Possible™
MD8470A-E-E-1

Slide 74

Anritsu

Video Telephony

Protocol Sequence
(OpenLogicalChannel)



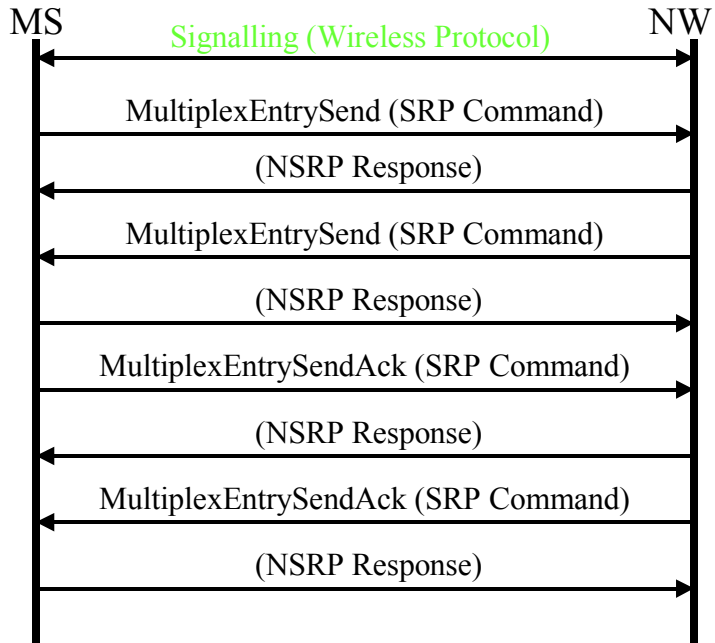
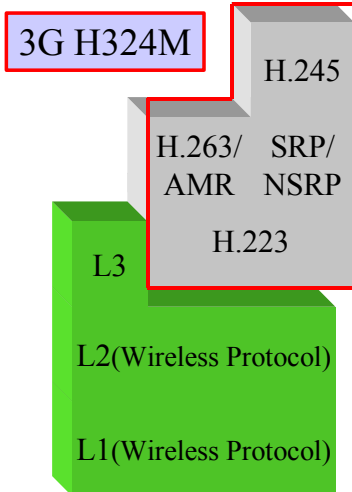
Discover What's Possible™
MD8470A-E-E-1

Slide 75



Video Telephony

Protocol Sequence
(MultiplexEntrySend)



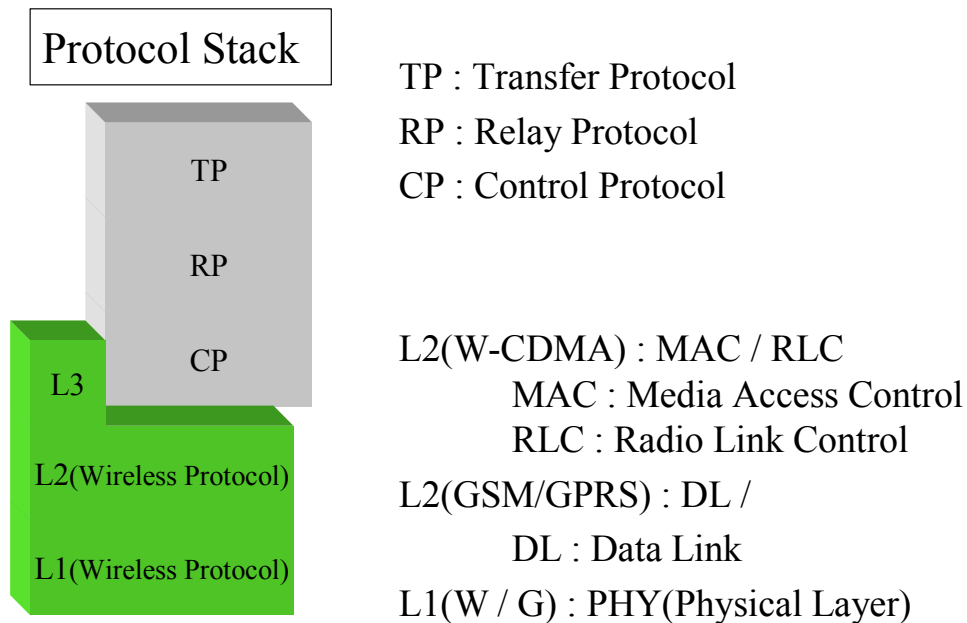
Discover What's Possible™
MD8470A-E-E-1

Slide 76



Part 4: SMS (Single SMS, Concatenated SMS)

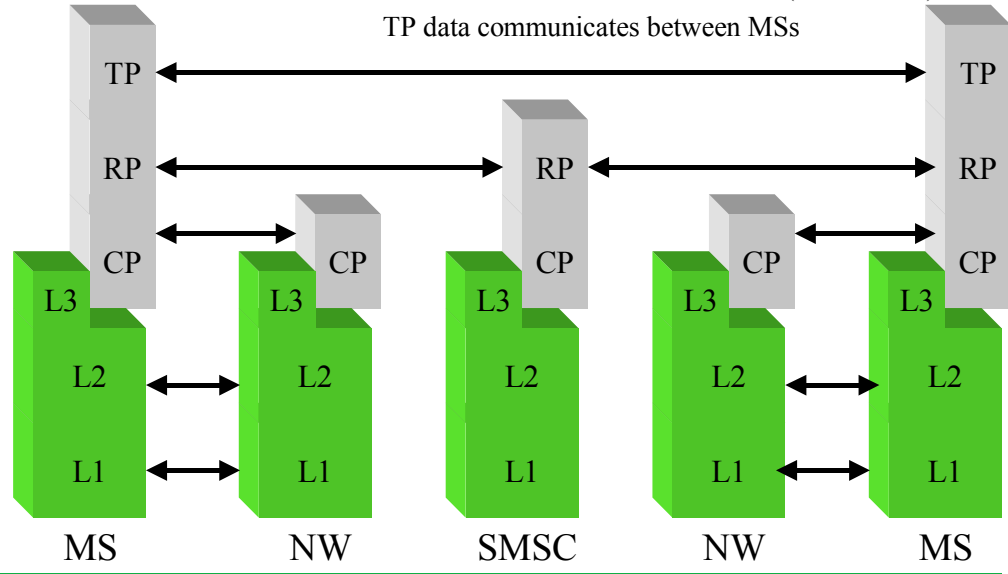
SMS (SMS – single SMS)



SMS (SMS – single SMS)

Protocol Stack

CP data is terminated in the network
 RP data is terminated in the SMSC (SMS center)
 TP data communicates between MSs



Discover What's Possible™
 MD8470A-E-E-1

Slide 79



Signalling Message (NAS message) structure

| Num. Of Octets | bit no. | | | | | | | |
|----------------|-----------------|----------|---|------------------------|--------|---|---|---|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | TI flag | TI value | | Protocol Discriminator | | | | |
| 1 | CP-Message type | | | | | | | |
| 1 | CP-UD length | | | | | | | |
| 1 | spare(0) | | | | RP-MTI | | | |
| 1 | RP-MR | | | | | | | |
| n | RP-OA | | | | | | | |
| n | RP-DA | | | | | | | |
| 1 | RP-UD length | | | | | | | |
| n | RP-UD | | | | | | | |

RP-UD structure
 1)SMS sending

| Num. Of Octets | bit no. | | | | | | | |
|----------------|---------|---------|--------|--------|---|--------|--------|---|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | TP-RP | TP-UDHI | TP-SRI | unused | | TP-MMS | TP-MTI | |
| 2 to 12 | TP-OA | | | | | | | |
| 1 | TP-PID | | | | | | | |
| 1 | TP-DCS | | | | | | | |
| 7 | TP-SCTS | | | | | | | |
| 1 | TP-UDL | | | | | | | |
| 0 to 140 | TP-UD | | | | | | | |

2)SMS receiving

| Num. Of Octets | bit no. | | | | | | | |
|----------------|---------|---------|--------|--------|-------|--------|---|---|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | TP-RP | TP-UDHI | TP-SRR | TP-VPF | TP-RD | TP-MTI | | |
| 1 | TP-MR | | | | | | | |
| 2 to 12 | TP-DA | | | | | | | |
| 1 | TP-PID | | | | | | | |
| 1 | TP-DCS | | | | | | | |
| 0, 1 or 7 | TP-VP | | | | | | | |
| 1 | TP-UDL | | | | | | | |
| 0 to 140 | TP-UD | | | | | | | |

SMS data structure

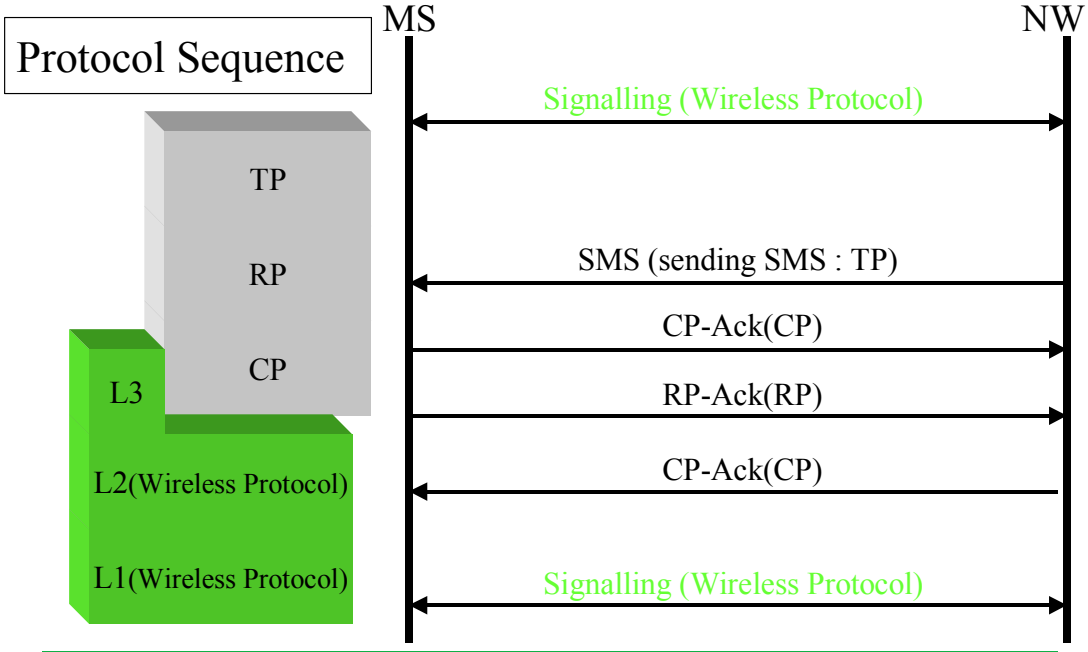
| Field Name | supported in |
|------------------------|--------------|
| TI flag + TI value | Scenario |
| Protocol Discriminator | SMS Library |
| CP-Message Type | SMS Library |
| CP-UD Length | SMS Library |
| RP-MTI | SMS Library |
| RP-MR | SMS Library |
| RP-OA | Scenario |
| RP-DA | Scenario |
| RP-UDL | SMS Library |
| TP-RP | SMS Library |
| TP-UDHI | SMS Library |
| TP-SRI | SMS Library |
| TP-MMS | SMS Library |
| TP-MTI | SMS Library |
| TP-OA | FakeSMSC |
| TP-PID | FakeSMSC |
| TP-DCS | FakeSMSC |
| TP-SCTS | FakeSMSC |
| TP-UDL | SMS Library |
| TP-UDH | FakeSMSC |
| TP-UD | SMS data |

Discover What's Possible™
 MD8470A-E-E-1

Slide 80



SMS (SMS – single SMS)

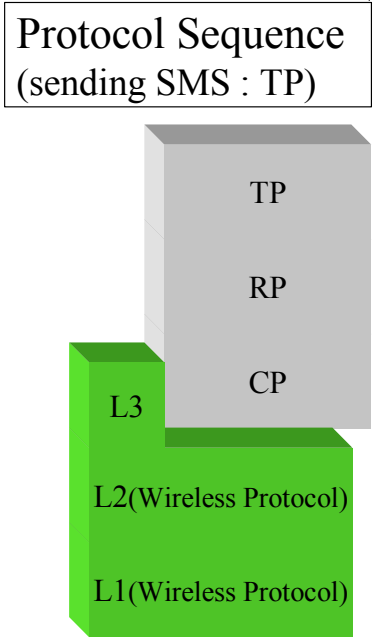


Discover What's Possible™
MD8470A-E-E-1

Slide 81



SMS (SMS – single SMS)



| PHY | MAC | RLC | TE | L3 | BTS | Primitive | Channel | Message |
|--|--|------------|----|----|-----|-----------------|---------|---------|
| | | | | | | RLC_AM_DATA_REQ | D | DCCH 3 |
| SMS MESSAGES | | | | | | | | |
| Sender = 0x70 Receiver = 0x40 Type = 0x4031 Type2 = 0x00 Unit = 0x00 | | | | | | | | |
| Channel = 0x45 Ch_No = 0x63 Op1 = 0x0000 Op2 = 0x0000 Message Length = 177 bytes | | | | | | | | |
| RR | NAS | SMS | | | | | | |
| Field | Value | Type | | | | | | |
| CP-DATA *1 | | DIVISION | | | | | | |
| Transaction identifier | | V | | | | | | |
| TI tag | The message is sent from the side that originates the TI | CHOICE | | | | | | |
| TI value | 0 | CHOICE | | | | | | |
| SMS messages protocol discriminator | | V | | | | | | |
| Protocol Discriminator | 9 | PD | | | | | | |
| CP-DATA Message type | | V | | | | | | |
| Message type | 01 | MSG | | | | | | |
| CP-User data *1 | | LV | | | | | | |
| Octet1 | | DIVISION | | | | | | |
| Length indicator *1 | 166 | LEN | | | | | | |
| Octet2-Octet249 | | DIVISION | | | | | | |
| RPDU | 010005899998888009C0405892143F50005050426172... | OCTETARRAY | | | | | | |

*1 : we can confirm the CP layer data. We can see the CP layer data on the 'NAS' tag (RP, TP layer data on the 'SMS' tag)

*2 : we can see the raw RP layer data

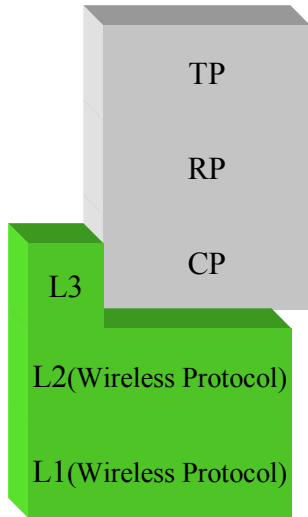
Discover What's Possible™
MD8470A-E-E-1

Slide 82



SMS (SMS – single SMS)

Protocol Sequence (sending SMS : TP)



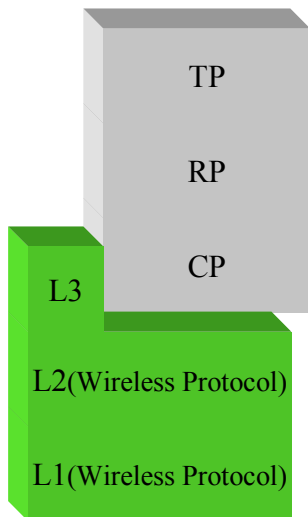
| PHY | MAC | RLC | TE | L3 | BTS | Primitive | Channel | Message |
|--|---------------------|---|-----------|----|-----|-----------------|----------|--------------|
| | | | | | | RLC_AM_DATA_R20 | P DCCH 3 | SMS MESSAGES |
| Sender = 0x70 Receiver = 0x40 Type = 0x4031 Type2 = 0x00 Unit = 0x00 Channel = 0x45 CH_No = 0x03 Opt1 = 0x0000 Opt2 = 0x0000 Message Length = 177 bytes | | | | | | | | |
| RRC | NAS | SMS | | | | | | |
| Field | Value | Description | | | | | | |
| CP-User data | | | | | | | | |
| Spare | 00000 | | | | | | | |
| RP-Message Type | 001 | RP-DATA(n->ms) | *1 | | | | | |
| RP-Message Reference | 0 | | | | | | | |
| RP-Originator Address | | | | | | | | |
| RP-Destination Address | | | | | | | | |
| RP-User Data | | | | | | | | |
| Length of RP-User Data | 156 | | *2 | | | | | |
| TP-MTI | 00 | SMS-DELIVER(SC to MS) | | | | | | |
| TP-MMS | 1 | No more messages are waiting for the MS in this SC | | | | | | |
| Spare | 00 | | | | | | | |
| TP-SRI | 0 | A status report shall not be returned to the SME | | | | | | |
| TP-UDHI | 0 | The TP-UD field contains only the short message | | | | | | |
| TP-RP | 0 | TP-Reply-Path parameter is not set in this SMS-SUBMIT/DELIVER | | | | | | |
| TP-OA | | | | | | | | |
| TP-PID | | | | | | | | |
| TP-DCS | | | | | | | | |
| BIT-6 | 00 | General Data Coding Indication | | | | | | |
| Indicates the text | 0 | Indicates the text is uncompressed | | | | | | |
| Indicates message class | 0 | Have no message class | | | | | | |
| Alphabet | 00 | GSM 7 bit default alphabet | *2 | | | | | |
| Message class | 00 | Class 0 | *2 | | | | | |
| TP-SCTS | 50504261720463 | | | | | | | |
| TP-UDL | 160 | Length of the TP-User-Data to follow | | | | | | |
| TP-UD | E170381C0E87C3E1... | | *2 | | | | | |

***1 : we can confirm the RP layer data (from NW to MS)**

***2 : we can see the TP layer data (SMS Deliver, 7bit default alphabet, message class, SMS data itself, e.t.c. ... Message class is from 0 to 3. '0' means 'Display as soon as possible'. '1' means 'Store the selected devices(SIM or memory)'. '2' means 'Store the SIM'. '3' means 'Forwarding to PC or PDA connected with MS.)**

SMS (SMS – single SMS)

Protocol Sequence (CP Ack : CP)

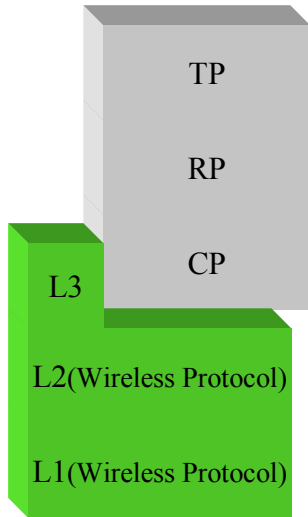


| PHY | MAC | RLC | TE | L3 | BTS | Primitive | Channel | Message |
|---|-------|----------|-----------|----|-----|-----------------|----------|--------------|
| | | | | | | MAC_DATA_IND | U DCCH 3 | |
| | | | | | | RLC_AM_DATA_IND | U DCCH 3 | SMS MESSAGES |
| Sender = 0x40 Receiver = 0x70 Type = 0x4131 Type2 = 0x00 Unit = 0x00 Channel = 0x45 CH_No = 0x03 Opt1 = 0x0000 Opt2 = 0x0000 Message Length = 10 bytes | | | | | | | | |
| RRC | NAS | | | | | | | |
| Field | Value | Type | | | | | | |
| CP-ACK | | | | | | | | |
| Transaction identifier | | DIVISION | | | | | | |
| TI flag | | V | | | | | | |
| TI0 | | CHOICE | | | | | | |
| | | CHOICE | | | | | | |
| | | V | | | | | | |
| SMS messages protocol discriminator | | | | | | | | |
| Protocol Discriminator | 9 | PD | | | | | | |
| CP-ACK Message type | | | | | | | | |
| Message type | 04 | V | *1 | | | | | |
| | | MSG | | | | | | |

***1 : we can confirm the CP Ack data (from MS to NW). This is just CP layer(no RP, TP layer) data.**

SMS (SMS – single SMS)

Protocol Sequence
(RP Ack : RP)



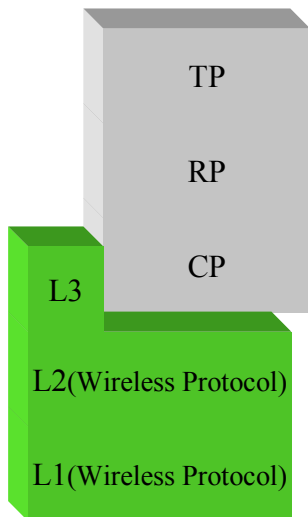
| PHY | MAC | RLC | TE | L3 | BTS | Primitive | Channel | Message |
|---|-----|-----|----|----|-----|-----------------|----------|--------------|
| | | | | | | MAC_DATA_IND | U DCCH 3 | |
| | | | | | | RLC_AM_DATA_IND | U DCCH 3 | SMS MESSAGES |
| Sender = 0x40 Receiver = 0x70 Type = 0x4131 Type2 = 0x00 Unit = 0x00 Channel = 0x05 CH_No = 0x03 Opt1 = 0x0000 Opt2 = 0x0000 Message Length = 13 bytes | | | | | | | | |
| RRC NAS SMS | | | | | | | | |
| Field Value Type | | | | | | | | |
| CP-DATA DIVISION | | | | | | | | |
| Transaction identifier V | | | | | | | | |
| TI flag The message is sent to the side... CHOICE | | | | | | | | |
| TIO TI value 0 CHOICE | | | | | | | | |
| SMS messages protocol discriminator V | | | | | | | | |
| Protocol Discriminator 9 PD | | | | | | | | |
| CP-DATA Message type V | | | | | | | | |
| Message type 01 MSG | | | | | | | | |
| CP-User data LV | | | | | | | | |
| Octet1 DIVISION *1 | | | | | | | | |
| Length indicator 2 LEN | | | | | | | | |
| Octet2-Octet249 DIVISION | | | | | | | | |
| RPDU 0200 OCTETARRAY | | | | | | | | |
| | | | | | | MAC_DATA_IND | U DCCH 3 | |
| | | | | | | RLC_AM_DATA_IND | U DCCH 3 | SMS MESSAGES |
| Sender = 0x40 Receiver = 0x70 Type = 0x4131 Type2 = 0x00 Unit = 0x00 Channel = 0x05 CH_No = 0x03 Opt1 = 0x0000 Opt2 = 0x0000 Message Length = 13 bytes | | | | | | | | |
| RRC NAS SMS | | | | | | | | |
| Field Value Description | | | | | | | | |
| CP-User data | | | | | | | | |
| Spare 00000 | | | | | | | | |
| RP-Message Type 010 RP-ACK(fms->ms) *2 | | | | | | | | |
| RP-Message Reference 0 | | | | | | | | |
| RP-User Data | | | | | | | | |

*1 : we can confirm the RP raw data (from MS to NW).

*2 : This is RP Ack data (from MS to NW)

SMS (SMS – single SMS)

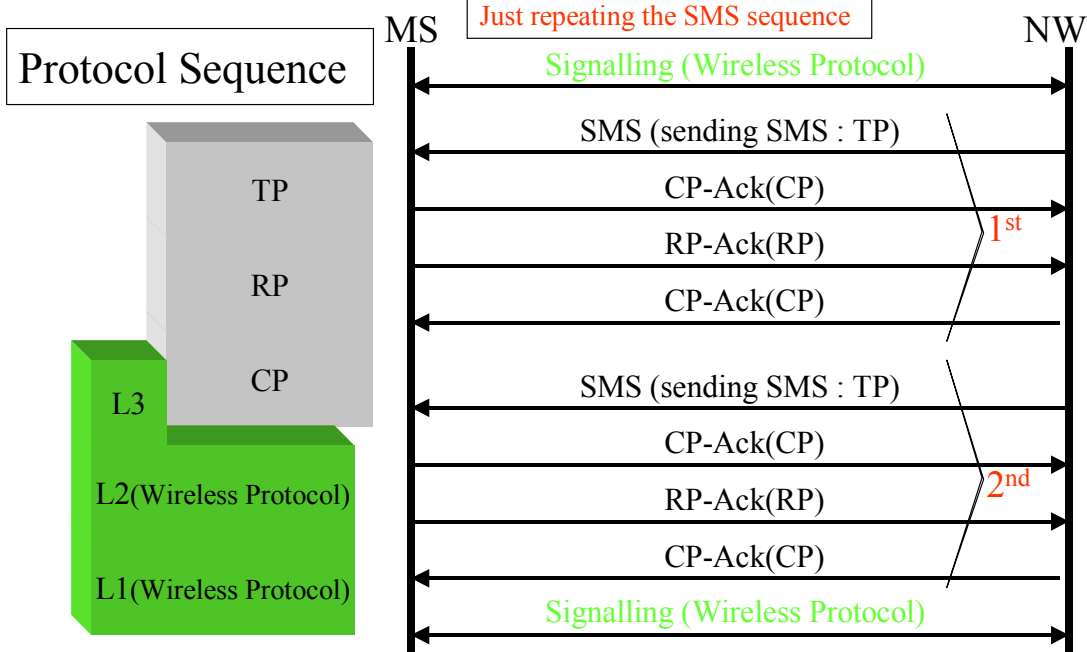
Protocol Sequence
(CP Ack : CP)



| PHY | MAC | RLC | TE | L3 | BTS | Primitive | Channel | Message |
|---|-----|-----|----|----|-----|-----------------|---------|--------------|
| | | | | | | PHY_DATA_REQ | D DCH 0 | |
| | | | | | | RLC_AM_DATA_REQ | D DCH 3 | SMS MESSAGES |
| Sender = 0x70 Receiver = 0x40 Type = 0x4031 Type2 = 0x00 Unit = 0x00 Channel = 0x05 CH_No = 0x03 Opt1 = 0x0000 Opt2 = 0x0000 Message Length = 10 bytes | | | | | | | | |
| RRC NAS | | | | | | | | |
| Field Value Type | | | | | | | | |
| CP-ACK DIVISION | | | | | | | | |
| Transaction identifier V | | | | | | | | |
| TI flag The message is sent from the side that originates the TI CHOICE | | | | | | | | |
| TIO TI value 0 CHOICE | | | | | | | | |
| SMS messages protocol discriminator V | | | | | | | | |
| Protocol Discriminator 9 PD | | | | | | | | |
| CP-ACK Message type V *1 | | | | | | | | |
| Message type 04 MSG | | | | | | | | |

*1 : we can confirm the CP Ack data for previous RP Ack (from NW to MS). This is just CP layer(no RP, TP layer data) data.

SMS (SMS – Concatenated)

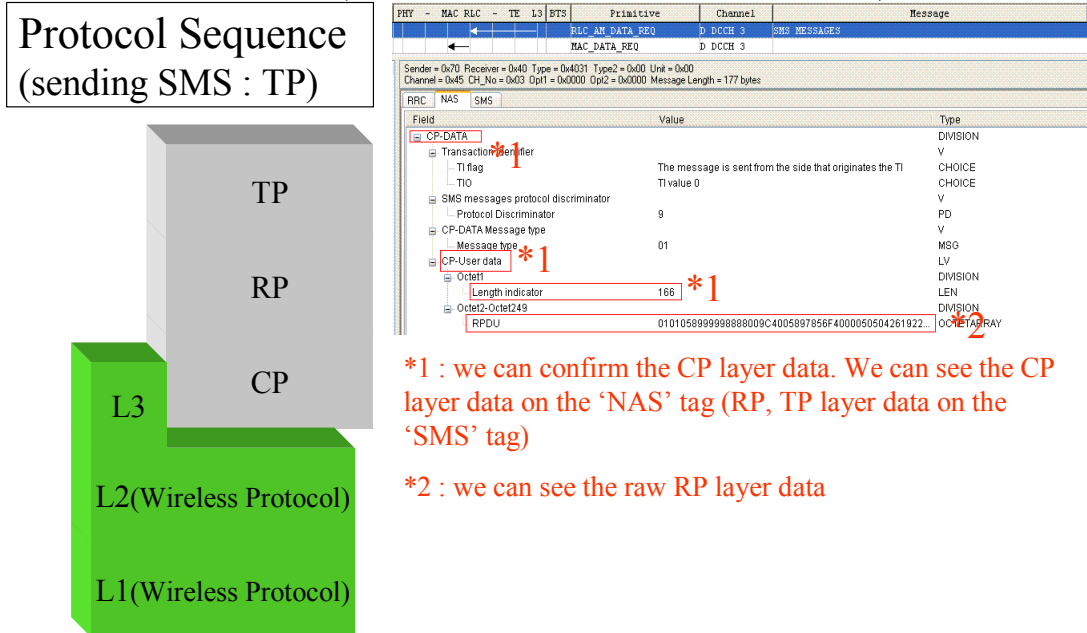


Discover What's Possible™
MD8470A-E-E-1

Slide 87

Anritsu

SMS (SMS – Concatenated)



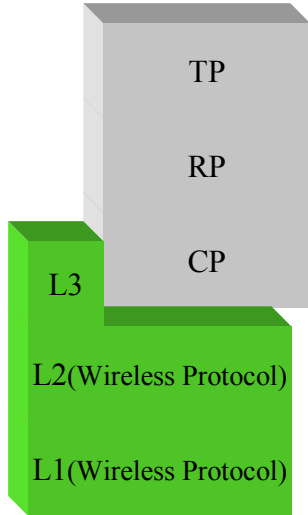
Discover What's Possible™
MD8470A-E-E-1

Slide 88

Anritsu

SMS (SMS – Concatenated)

Protocol Sequence (sending SMS : TP)



| PHY | MAC | RLC | TE | L3 | BTS | Primitive | Channel | Message |
|--|-----|-----|-------------------------|----|-----|-----------------|-----------|--------------|
| | | | | | | RLC_AM_DATA_REQ | D DCCCH 3 | SMS MESSAGES |
| Sender = 0x70 Receiver = 0x40 Type = 0x4031 Type2 = 0x00 Unit = 0x00 Channel = 0x45 CH_No = 0x03 Opt1 = 0x0000 Opt2 = 0x0000 Message Length = 177 bytes | | | | | | | | |
| FRC | NAS | SMS | Field Value Description | | | | | |
| CP-User data | | | | | | | | |
| Spare 00000 | | | | | | | | |
| RP-Message Type 001 RP-DATA(n->ms) *1 | | | | | | | | |
| RP-Message Reference 1 | | | | | | | | |
| RP-Originator Address | | | | | | | | |
| RP-Destination Address | | | | | | | | |
| RP-User Data | | | | | | | | |
| Length of RP-User Data 156 *2 | | | | | | | | |
| TP-MTI 00 SMS-DELIVER(SC to MS) *3 | | | | | | | | |
| TP-MMS 0 More messages are waiting for the MS in this SC *3 | | | | | | | | |
| Spare 00 | | | | | | | | |
| TP-SRI 0 A status report shall not be returned to the SME | | | | | | | | |
| TP-UDHI 1 The beginning of the TP-UD field contains a Header in addition to the short message *3 | | | | | | | | |
| TP-REP 0 | | | | | | | | |
| TP-OA 0 TP-Reply-Path parameter is not set in this SMS-SUBMITDELIVER *3 | | | | | | | | |
| TP-PID | | | | | | | | |
| TP-DCS | | | | | | | | |
| BIT-6 00 General Data Coding indication | | | | | | | | |
| Indicates the text 0 Indicates the text is uncompressed | | | | | | | | |
| Indicates message cla... 0 Have no message class | | | | | | | | |
| Alphabet 00 GSM 7 bit default alphabet *2 | | | | | | | | |
| Message class 00 Class 0 *2 | | | | | | | | |
| TP-SCTS 50504261922563 | | | | | | | | |
| TP-UDL 160 Length of the TP-User-Data field to follow *3 | | | | | | | | |
| TP-UD 050003000201C462B1... *3 | | | | | | | | |

*1 : we can confirm the RP layer data (from NW to MS)

*2 : TP layer data (SMS Deliver, 7bit default alphabet, message class, e.t.c.)

*3 : this is not last message(concatenated message) this TP-UD has TP-UDH(User Data Header). TP-UD '050003000201...' means '05' is the length of UDH, next '00' means IEI(in the case of concatenate SMS), next '03' means the length of this IEI, next '00' means sms sequence number, next '02' means 'how many concatenate' next '01' means 1st block of concatenated SMS.

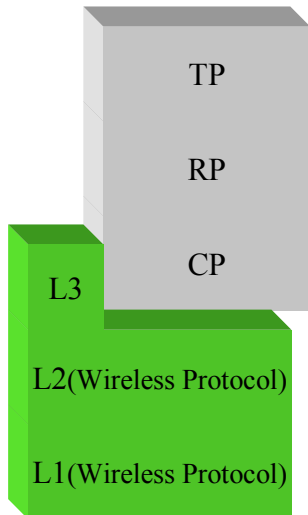
Discover What's Possible™
MD8470A-E-E-1

Slide 89

Anritsu

SMS (SMS – Concatenated)

Protocol Sequence (CP Ack : CP)



| PHY | MAC | RLC | TE | L3 | BTS | Primitive | Channel | Message |
|---|-----|------------------|----|----|-----|--------------|-----------|--------------|
| | | | | | | MAC_DATA_IND | U DCCCH 3 | SMS MESSAGES |
| Sender = 0x40 Receiver = 0x70 Type = 0x4131 Type2 = 0x00 Unit = 0x00 Channel = 0x45 CH_No = 0x03 Opt1 = 0x0000 Opt2 = 0x0000 Message Length = 10 bytes | | | | | | | | |
| FRC | NAS | Field Value Type | | | | | | |
| CP-ACK *1 | | | | | | | | |
| Transaction identifier | | | | | | | | |
| TI flag The message is sent to the side that originates the TI CHOICE | | | | | | | | |
| TIO TI value 0 CHOICE | | | | | | | | |
| SMS messages protocol discriminator | | | | | | | | |
| Protocol Discriminator 9 PD | | | | | | | | |
| CP-ACK Message type *1 | | | | | | | | |
| Message type 04 V MSG | | | | | | | | |

*1 : we can confirm the CP Ack data (from MS to NW). This is just CP layer(no RP, TP layer) data.

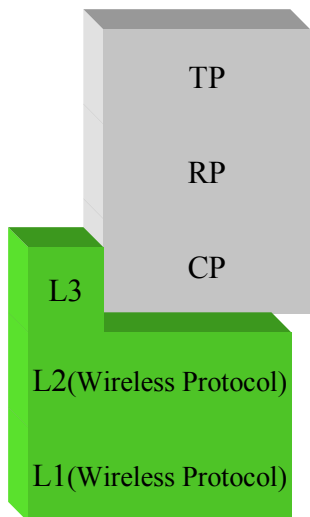
Discover What's Possible™
MD8470A-E-E-1

Slide 90

Anritsu

SMS (SMS – Concatenated)

Protocol Sequence
(RP Ack : RP)



| PHY | MAC | RLC | TE | L3 | BTS | Primitive | Channel | Message |
|-----|-----|-----|----|----|-----|-----------------|----------|--------------|
| | | | | | | MAC_DATA_IND | D DCCH 3 | |
| | | | | | | RLC_AM_DATA_IND | D DCCH 3 | SMS MESSAGES |

Sender = 0x40 Receiver = 0x70 Type = 0x4131 Type2 = 0x00 Unit = 0x00
Channel = 0x05 CH_No = 0x03 Opt1 = 0x0000 Opt2 = 0x0000 Message Length = 13 bytes

| RRC | NAS | SMS | Field | Value | Type |
|-----|-----|-----|-------------------------------------|--|------------|
| | | | CP-DATA | | DIVISION |
| | | | Transaction identifier | | V |
| | | | TI flag | The message is sent to the side that originates the TI | CHOICE |
| | | | TIO | TI value 0 | CHOICE |
| | | | SMS messages protocol discriminator | | V |
| | | | Protocol Discriminator | 9 | PD |
| | | | CP-DATA Message type | | V |
| | | | Message type | 01 | MSG |
| | | | CP-User data | | LV |
| | | | Octet1 | | DIVISION |
| | | | Length indicator | 2 | LEN |
| | | | Octet2-Octet249 | | DIVISION |
| | | | RPDU | 0201 | OCTETARRAY |

*1

| RRC | NAS | SMS | Field | Value | Description |
|-----|-----|-----|----------------------|-------|--------------|
| | | | CP-User data | | |
| | | | Spare | 00000 | |
| | | | RP-Message Type | 010 | RP-ACK(ms->) |
| | | | RP-Message Reference | 1 | |
| | | | RP-User Data | | |

*2

*1 : we can confirm the RP raw data (from MS to NW).
*2 : This is RP Ack data (from MS to NW)

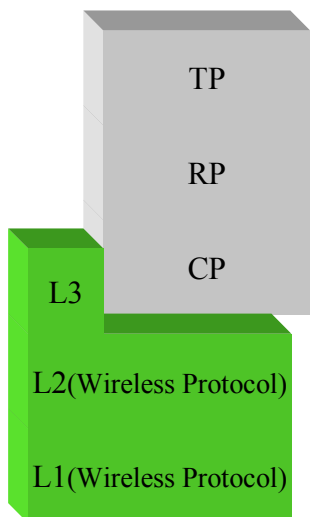
Discover What's Possible™
MD8470A-E-E-1

Slide 91

Anritsu

SMS (SMS – Concatenated)

Protocol Sequence
(CP Ack : CP)



| PHY | MAC | RLC | TE | L3 | BTS | Primitive | Channel | Message |
|-----|-----|-----|----|----|-----|-----------------|----------|--------------|
| | | | | | | PHY_DATA_REQ | D DCCH 0 | |
| | | | | | | RLC_AM_DATA_REQ | D DCCH 3 | SMS MESSAGES |

Sender = 0x70 Receiver = 0x40 Type = 0x4031 Type2 = 0x00 Unit = 0x00
Channel = 0x05 CH_No = 0x03 Opt1 = 0x0000 Opt2 = 0x0000 Message Length = 10 bytes

| RRC | NAS | SMS | Field | Value | Type |
|-----|-----|-----|-------------------------------------|--|----------|
| | | | CP-ACK | | DIVISION |
| | | | Transaction identifier | | V |
| | | | TI flag | The message is sent from the side that originates the TI | CHOICE |
| | | | TIO | TI value 0 | CHOICE |
| | | | SMS messages protocol discriminator | | V |
| | | | Protocol Discriminator | 9 | PD |
| | | | CP-ACK Message type | | V |
| | | | Message type | 04 | MSG |

*1

*1 : we can confirm the CP Ack data for previous RP Ack (from NW to MS). This is just CP layer(no RP, TP layer data) data.

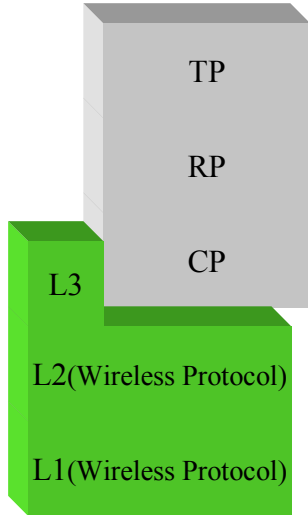
Discover What's Possible™
MD8470A-E-E-1

Slide 92

Anritsu

SMS (SMS – Concatenated)

Protocol Sequence
(sending SMS : TP)



| PHY | MAC | RLC | TE | L3 | BTS | Primitive | Channel | Message |
|-----|-----|-----|----|----|-----|-----------------|----------|--------------|
| | | | | | | RLC_AM_DATA_REQ | D DCCH 3 | SMS MESSAGES |
| | | | | | | MAC_DATA_REQ | D DCCH 3 | |

Sender = 0x70 Receiver = 0x40 Type = 0x4031 Type2 = 0x00 Unit = 0x00
Channel = 0x45 CH_No = 0x03 Dp1 = 0x0000 Dp2 = 0x0000 Message Length = 177 bytes

| RRC | NAS | SMS |
|-----|-----|-----|
| | | |

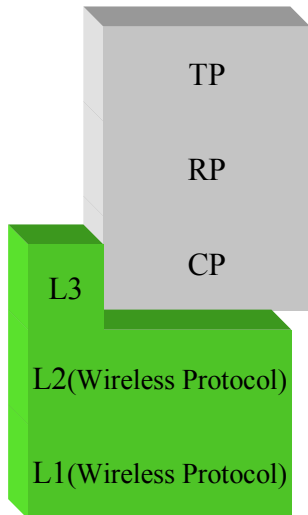
| Field | Value | Type |
|-------------------------------------|--|----------|
| CP-DATA | | DIVISION |
| Transaction Identifier | | V |
| - TI flag | The message is sent from the side that originates the TI | CHOICE |
| - TIO | Ti value 1 | CHOICE |
| SMS messages protocol discriminator | | V |
| - Protocol Discriminator | 9 | PD |
| CP-DATA Message type | | V |
| Message type | 01 | MSG |
| CP-User data | | LV |
| Length indicator | 166 | DIVISION |
| Octet2-Octet249 | | LEN |
| RPDU | 0102059999998888009C4405897856F4000050504261922563400500030... | DIVISION |

*1 : we can confirm the CP layer data. We can see the CP layer data on the 'NAS' tag (RP, TP layer data on the 'SMS' tag)

*2 : we can see the raw RP layer data

SMS (SMS – Concatenated)

Protocol Sequence
(sending SMS : TP)



| PHY | MAC | RLC | TE | L3 | BTS | Primitive | Channel | Message |
|-----|-----|-----|----|----|-----|-----------------|----------|--------------|
| | | | | | | RLC_AM_DATA_REQ | D DCCH 3 | SMS MESSAGES |

Sender = 0x70 Receiver = 0x40 Type = 0x4031 Type2 = 0x00 Unit = 0x00
Channel = 0x45 CH_No = 0x03 Dp1 = 0x0000 Dp2 = 0x0000 Message Length = 177 bytes

| RRC | NAS | SMS |
|-----|-----|-----|
| | | |

| Field | Value | Description |
|-------------------------|---------------------|--|
| CP-User data | | |
| Spare | 00000 | |
| RP-Message Type | 001 | RP-DATA(n->ms) *1 |
| RP-Message Reference | 2 | |
| RP-Originator Address | | |
| RP-Destination Address | | |
| RP-User Data | | |
| Length of RP-User Data | 156 | *2 |
| TP-MTI | 00 | SMS-DELIVER(SC to MS) *2 |
| TP-MMS | 1 | No more messages are waiting for the MS in this SC *3 |
| Spare | 00 | |
| TP-SRI | 0 | A status report shall not be returned to the SME |
| TP-UDHI | 1 | The beginning of the TP-UD field contains a Header in addition to the short message *3 |
| TP-RP | 0 | TP-Reply-Path parameter is not set in this SMS-SUBMIT/DELIVER *3 |
| TP-OA | | |
| TP-PID | | |
| TP-DCS | | |
| BIT-6 | 00 | General Data Coding Indication |
| Indicates the text | 0 | Indicates the text is uncompressed |
| Indicates message class | 0 | Have no message class |
| Alphabet | 00 | GSM 7 bit default alphabet *2 |
| Message class | 00 | Class 0 *2 |
| TP-SCTS | 50504261922563 | |
| TP-UDL | 160 | Length of the TP-User-Data field to follow *3 |
| TP-UD | 050003000202C462... | *3 |

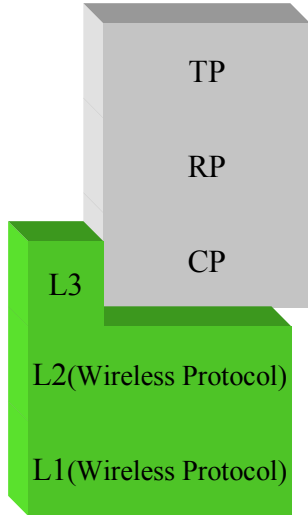
*1 : we can confirm the RP layer data (from NW to MS)

*2 : TP layer data (SMS Deliver, 7bit default alphabet, message class, e.t.c.)

*3 : this is the last message(concatenated message) this TP-UD has TP-UDH(User Data Header). TP-UD '050003000202...' means '05' is the length of UDH, next '00' means IEI(in the case of concatenate SMS), next '03' means the length of this IEI, next '00' means sms sequence number, next '02' means 'how many concatenate' next '02' means 2nd block of concatenated SMS.

SMS (SMS – Concatenated)

Protocol Sequence
(CP Ack : CP)



| PHY | MAC | RLC | TE | L3 | BTS | Primitive | Channel | Message |
|-----|-----|-----|----|----|-----|-----------------|----------|--------------|
| | | | | | | MAC_DATA_IND | U DCCH 3 | |
| | | | | | | RLC_AI_DATA_IND | U DCCH 3 | SMS MESSAGES |

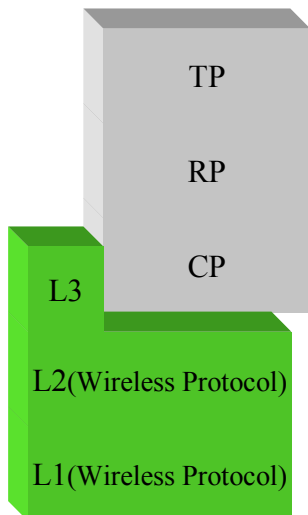
Sender = 0x40 Receiver = 0x70 Type = 0x4131 Type2 = 0x00 Unit = 0x00
Channel = 0x5 CH_No = 0x03 Opt1 = 0x0000 Opt2 = 0x0000 Message Length = 10 bytes

| RRC | NAS | Field | Value | Type |
|-----|-----|-------------------------------------|--|----------|
| | | CP-ACK | | DIVISION |
| | | Transaction identifier | | V |
| | | TI flag | The message is sent to the side that originates the TI | CHOICE |
| | | TIO | TI value 0 | CHOICE |
| | | SMS messages protocol discriminator | | V |
| | | Protocol Discriminator | 9 | PD |
| | | CP-ACK Message type | | V |
| | | Message type | 04 | MSG |

*1 : we can confirm the CP Ack data (from MS to NW).
This is just CP layer(no RP, TP layer data) data.

SMS (SMS – Concatenated)

Protocol Sequence
(RP Ack : RP)



| PHY | MAC | RLC | TE | L3 | BTS | Primitive | Channel | Message |
|-----|-----|-----|----|----|-----|-----------------|----------|--------------|
| | | | | | | MAC_DATA_IND | U DCCH 3 | |
| | | | | | | RLC_AI_DATA_IND | U DCCH 3 | SMS MESSAGES |

Sender = 0x40 Receiver = 0x70 Type = 0x4131 Type2 = 0x00 Unit = 0x00
Channel = 0x5 CH_No = 0x03 Opt1 = 0x0000 Opt2 = 0x0000 Message Length = 13 bytes

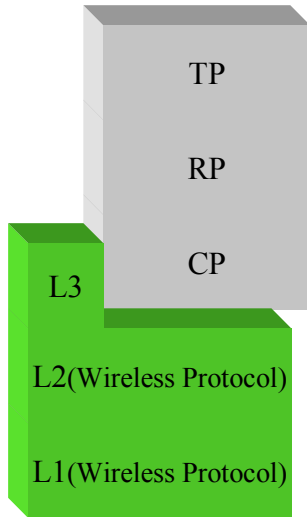
| RRC | NAS | SMS | Field | Value | Type |
|-----|-----|-----|-------------------------------------|--|------------|
| | | | CP-DATA | | DIVISION |
| | | | Transaction identifier | | V |
| | | | TI flag | The message is sent to the side that originates the TI | CHOICE |
| | | | TIO | TI value 1 | CHOICE |
| | | | SMS messages protocol discriminator | | V |
| | | | Protocol Discriminator | 9 | PD |
| | | | CP-DATA Message type | | V |
| | | | Message type | 01 | MSG |
| | | | CP-User data | | LV |
| | | | Octet1 | | DIVISION |
| | | | Length indicator | 2 | LEN |
| | | | Octet2-Octet249 | | DIVISION |
| | | | RPDU | 0202 | OCTETARRAY |

| RRC | NAS | SMS | Field | Value | Description |
|-----|-----|-----|----------------------|-------|---------------|
| | | | CP-User data | | |
| | | | Spare | 00000 | |
| | | | RP-Message Type | 010 | RP-ACK(ms->n) |
| | | | RP-Message Reference | | |
| | | | RP-User Data | 2 | |

*1 : we can confirm the RP raw data (from MS to NW).
*2 : This is RP Ack data (from MS to NW)

SMS (SMS – Concatenated)

Protocol Sequence
(CP Ack : CP)

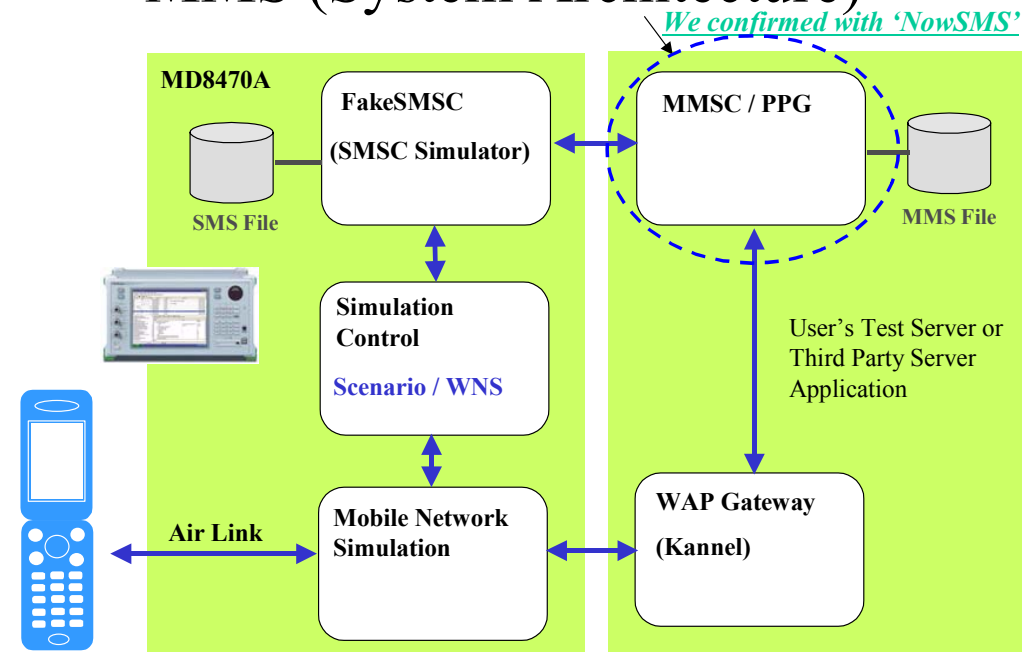


| PHY | MAC | RLC | TE | L3 | BTS | Primitive | Channel | Message |
|---|--|----------|----|----|-----|-----------------|---------|--------------|
| | | | | M1 | | PHY_DATA_REQ | D DCH 0 | |
| | | | | | | RLC_AM_DATA_REQ | D DCH 3 | SMS MESSAGES |
| Sender = 0x70 Receiver = 0x40 Type = 0x4031 Type2 = 0x00 Unit = 0x00 Channel = 0x45 CH_No = 0x03 Dpt1 = 0x0000 Dpt2 = 0x0000 Message Length = 10 bytes | | | | | | | | |
| RRC | NAS | | | | | | | |
| Field | Value | Type | | | | | | |
| CP-ACK | | DIVISION | | | | | | |
| Transaction identifier | | V | | | | | | |
| TI flag | The message is sent from the side that originates the TI | CHOICE | | | | | | |
| TI0 | TI value 0 | CHOICE | | | | | | |
| SMS messages protocol discriminator | | V | | | | | | |
| Protocol Discriminator | 9 | FD | | | | | | |
| CP-ACK Message type | | V | | | | | | |
| Message type | 04 | MSG *1 | | | | | | |

*1 : we can confirm the CP Ack data for previous RP Ack (from NW to MS). This is just CP layer(no RP, TP layer data) data.

Part 5: MMS (HTTP-Submit, HTTP-Retrieve, WAP-Submit, WAP-Retrieve)

MMS (System Architecture)



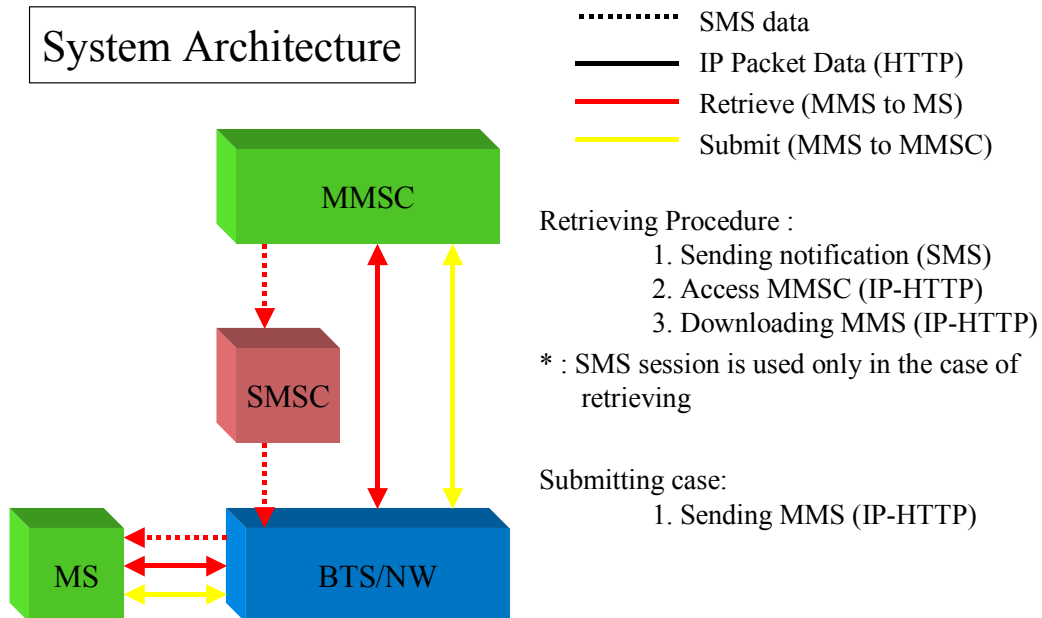
Discover What's Possible™
MD8470A-E-E-1

Slide 99

Anritsu

MMS (MMS – HTTP)

System Architecture

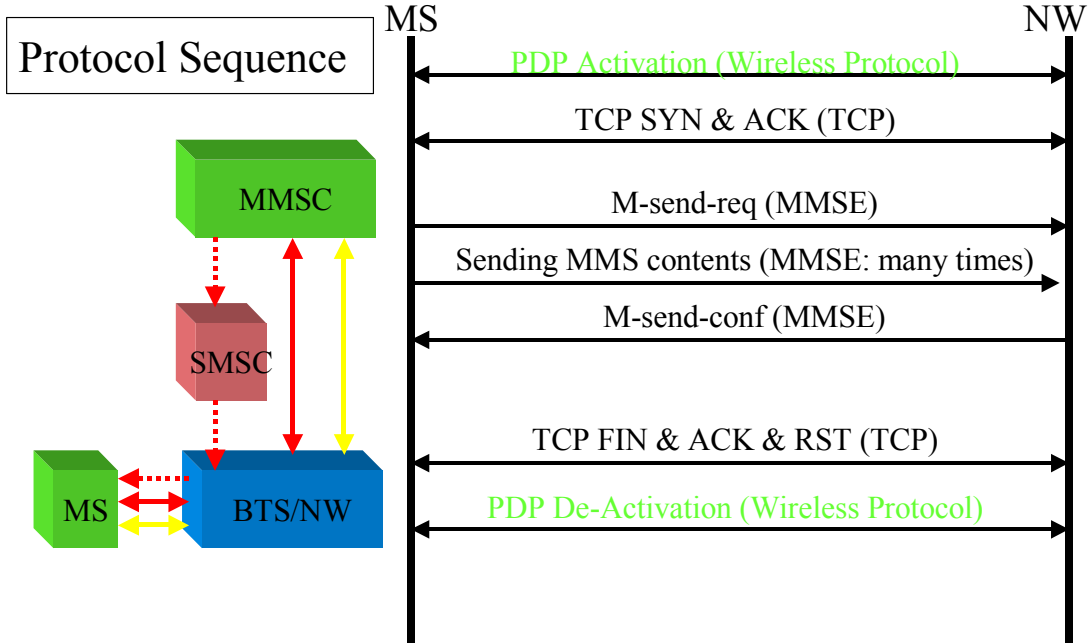


Discover What's Possible™
MD8470A-E-E-1

Slide 100

Anritsu

MMS Submit (MMS – HTTP)

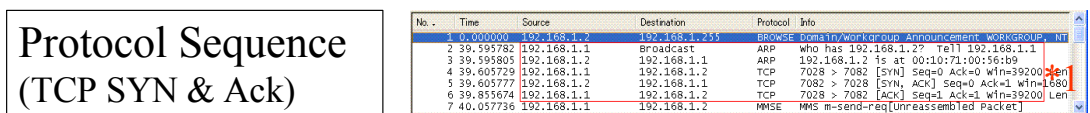


Discover What's Possible™
MD8470A-E-E-1

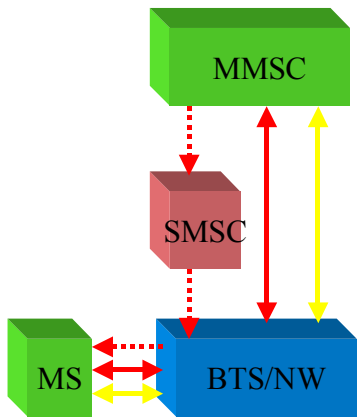
Slide 101

Anritsu

MMS Submit (MMS – HTTP)



| No. | Time | Source | Destination | Protocol | Info |
|-----|--------------|-------------|---------------|-----------|---|
| 1 | 00:00:00.000 | 192.168.1.2 | 192.168.1.255 | Broadcast | who has 192.168.1.2? Tell 192.168.1.1 |
| 2 | 39.593782 | 192.168.1.1 | Broadcast | ARP | 192.168.1.2 is at 00:10:17:100:56:b9 |
| 3 | 39.595805 | 192.168.1.2 | 192.168.1.1 | ARP | 192.168.1.2 is at 00:10:17:100:56:b9 |
| 4 | 39.603729 | 192.168.1.1 | 192.168.1.2 | TCP | 7028 > 7082 [SYN] Seq=0 Ack=0 Win=39200 Len=0 |
| 5 | 39.605777 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [SYN, ACK] Seq=0 Ack=1 Win=1680 Len=0 |
| 6 | 39.855674 | 192.168.1.1 | 192.168.1.2 | TCP | 7028 > 7082 [ACK] Seq=1 Ack=1 Win=39200 Len=0 |
| 7 | 40.057736 | 192.168.1.1 | 192.168.1.2 | MMSE | MMS m-send-req[Unreassembled Packet] |



*1 : These process is same as the part 'HTTP content download (Please refer to part 1 'HTTP' in detail)'.

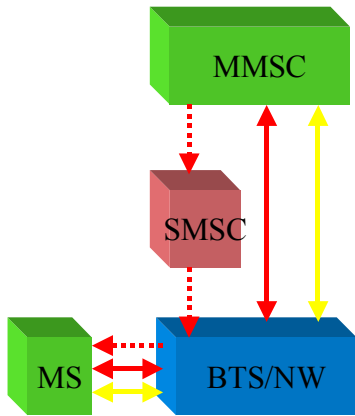
Discover What's Possible™
MD8470A-E-E-1

Slide 102

Anritsu

MMS Submit (MMS – HTTP)

Protocol Sequence (m-send-req)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 6 | 39.855674 | 192.168.1.1 | 192.168.1.2 | TCP | 7028 > 7082 [ACK] Seq=1 Ack=1 wIn=39200 Len=... |
| 7 | 40.037270 | 192.168.1.1 | 192.168.1.2 | MIME | MMS:m-send-req[1116948389] Len=1400 |
| 8 | 40.173639 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [ACK] Seq=1 Ack=1401 wIn=16800 Len=... |

```

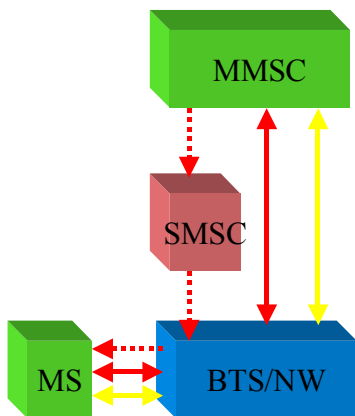
    POST /xst=test HTTP/1.1\r\n
    Connection: keep-alive\r\n
    Host: 192.168.1.2:7082\r\n
    User-Agent: Samsung Mocha Messenger 1.5\r\n
    Content-Type: application/vnd.wap.mms-message\r\n
    Content-Length: 12767\r\n
    Profile: http://wap.samsungobile.com/uaprof/z105UAProf.rdf\r\n
    X-VODAFONE-Sspoccontext: yss\r\n
    \r\n
    MMS Message Encapsulation, Type: m-send-req
    X-Mms-Message-Type: m-send-req (0x80)
    X-Mms-Transaction-ID: 1116948389, 6722
    X-Mms-MMS-Version: 1.0
    Date: May 25, 2005 01:26:28.000000000
    From: <insert address>
    To: 6666666/TYPE=PLMN
    Subject: nk
    X-Mms-Message-Class: Personal (0x80)
    X-Mms-Priority: Normal (0x81)
    Sender-visibility: Show (0x81)
    X-Mms-Delivery-Report: No (0x81)
    X-Mms-Read-Report: No (0x81)
    Content-Type: application/vnd.wap.multipart.related; type=application/smil; start=<1116948389>
  
```

*1 : 'POST' means 'sending from MS to Server'. 'xst=test' means 'UserID=xst, Password=test' for MMSC server. 'HOST' is '192.168.1.2(IP address), 7082(port number).

*2 : Message Type = 'm-send-req (sending MMS)' And we can confirm 'Class', 'Priority', 'deliver report', 'read report', e.t.c. .

MMS Submit (MMS – HTTP)

Protocol Sequence (m-send-req)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 6 | 39.855674 | 192.168.1.1 | 192.168.1.2 | TCP | 7028 > 7082 [ACK] Seq=1 Ack=1 wIn=39200 Len=... |
| 7 | 40.037270 | 192.168.1.1 | 192.168.1.2 | MIME | MMS:m-send-req[1116948389] Len=1400 |
| 8 | 40.173639 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [ACK] Seq=1 Ack=1401 wIn=16800 Len=... |

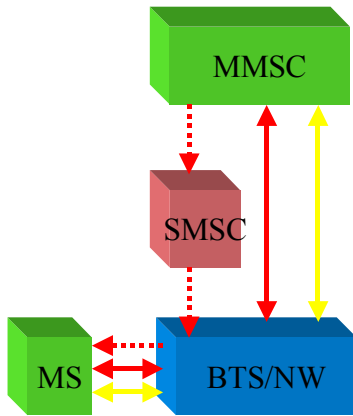
```

    X-Mms-Transaction-ID: 1116948389, 6722
    X-Mms-MMS-Version: 1.0
    Date: May 25, 2005 01:26:28.000000000
    From: <insert address>
    To: 6666666/TYPE=PLMN
    Subject: nk
    X-Mms-Message-Class: Personal (0x80)
    X-Mms-Priority: Normal (0x81)
    Sender-visibility: Show (0x81)
    X-Mms-Delivery-Report: No (0x81)
    X-Mms-Read-Report: No (0x81)
    Content-Type: application/vnd.wap.multipart.related; type=application/smil; start=<1116948389>
    Data (POST)
    Multipart body
    Part: 1, content-type: application/smil
    Content-Type: application/smil; charset=utf-8
    Headers
    extensible Markup Language
    <smil xmlns="http://www.w3.org/2000/SMIL20/CR/Language">
    <head>
    </head>
    <body>
    </smil>
    [Unreassembled Packet: MMSE]
  
```

*1 : 'Data' means MMS data itself. MMS data is big, so MS send MMS data divided many blocks. This is the first block of MMS data (not completed yet).

MMS Submit (MMS – HTTP)

Protocol Sequence (sending MMS content)

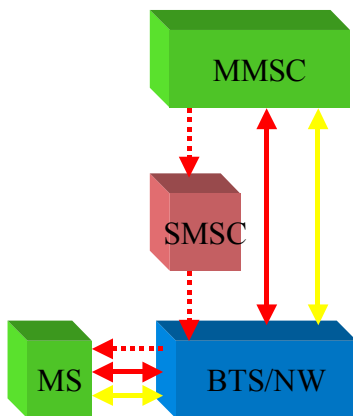


| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 7 | 40.057736 | 192.168.1.1 | 192.168.1.2 | MMSE | MMS m-send-req[Unreassembled Packet] |
| 8 | 40.179639 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [ACK] Seq=1 Ack=1401 win=16800 |
| 9 | 40.437920 | 192.168.1.1 | 192.168.1.2 | HTTP | Continuation or non-HTTP traffic |
| 10 | 40.574902 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [ACK] Seq=1 Ack=2801 win=16800 |
| 11 | 41.417624 | 192.168.1.1 | 192.168.1.2 | HTTP | Continuation or non-HTTP traffic |
| 12 | 41.578099 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [ACK] Seq=1 Ack=4201 win=16800 |
| 13 | 42.417946 | 192.168.1.1 | 192.168.1.2 | HTTP | Continuation or non-HTTP traffic |
| 14 | 42.581318 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [ACK] Seq=1 Ack=5601 win=16800 |
| 15 | 43.417476 | 192.168.1.1 | 192.168.1.2 | HTTP | Continuation or non-HTTP traffic |
| 16 | 43.584502 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [ACK] Seq=1 Ack=7001 win=16800 |
| 17 | 44.417409 | 192.168.1.1 | 192.168.1.2 | HTTP | Continuation or non-HTTP traffic |
| 18 | 44.587707 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [ACK] Seq=1 Ack=8401 win=16800 |
| 19 | 45.417336 | 192.168.1.1 | 192.168.1.2 | HTTP | Continuation or non-HTTP traffic |
| 20 | 45.590916 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [ACK] Seq=1 Ack=9801 win=16800 |
| 21 | 46.437264 | 192.168.1.1 | 192.168.1.2 | HTTP | Continuation or non-HTTP traffic |
| 22 | 46.594095 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [ACK] Seq=1 Ack=11201 win=16800 |
| 23 | 47.437298 | 192.168.1.1 | 192.168.1.2 | HTTP | Continuation or non-HTTP traffic |
| 24 | 47.597299 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [ACK] Seq=1 Ack=12601 win=16800 |
| 25 | 48.295681 | 192.168.1.1 | 192.168.1.2 | HTTP | Continuation or non-HTTP traffic |
| 26 | 48.399864 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [ACK] Seq=1 Ack=13044 win=16357 |
| 27 | 48.455404 | 192.168.1.2 | 192.168.1.1 | MMSE | MMS m-send-conf |
| 28 | 48.456470 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [FIN, ACK] Seq=163 Ack=13044 wi |

*1 : 'Continuation or non-HTTP traffic' means 'MMS Data sending(from MS to Server)'. We can confirm the TCP Ack(from Server to MS) for each data. (Server side is smooth enough, so we can confirm the Ack for each data.)

MMS Submit (MMS – HTTP)

Protocol Sequence (m-send-conf)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 26 | 48.399864 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [ACK] Seq=1 Ack=13044 win=16357 |
| 27 | 48.455404 | 192.168.1.2 | 192.168.1.1 | MMSE | MMS m-send-conf |
| 28 | 48.456470 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [FIN, ACK] Seq=163 Ack=13044 wi |

```

# Frame 27 (216 bytes on wire, 216 bytes captured)
# Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:91:03:19:15
# Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
# Transmission Control Protocol, Src Port: 7082 (7082), Dst Port: 7028 (7028), Seq: 1, Ack: 13044, Len: 162
# Hypertext Transfer Protocol
# HTTP/1.0 200 OK\r\n
# Connection: close\r\n
# Content-Type: application/vnd.wap.mms-message\r\n
# Content-Length: 57\r\n
# \r\n
# MMS Message Encapsulation, Type: m-send-conf
# X-Mms-Message-Type: m-send-conf (0x81)
# X-Mms-Transaction-ID: 1116648389.6722
# X-Mms-Version: 1.0
# Response-Status: Ok (0x80)
# Message-ID: 20050524/16/C5E00A96@192.168.1.2
    
```

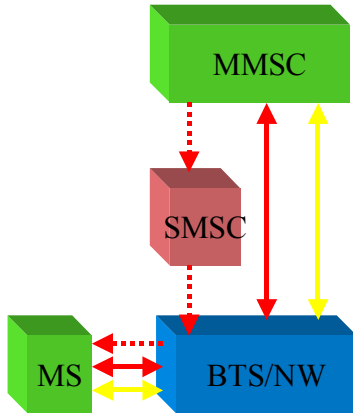
*1 : '200 OK' means 'succeed of sending data at HTTP level.'

*2 : 'm-send-conf' means the acknowledge from Server to MS for the success of MMS sending.

MMS Submit (MMS – HTTP)

Protocol Sequence (TCP FIN & Ack)

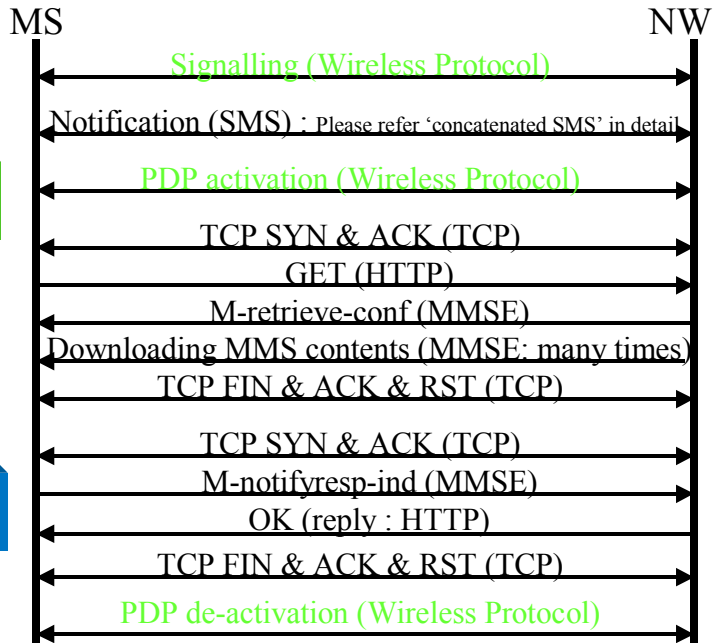
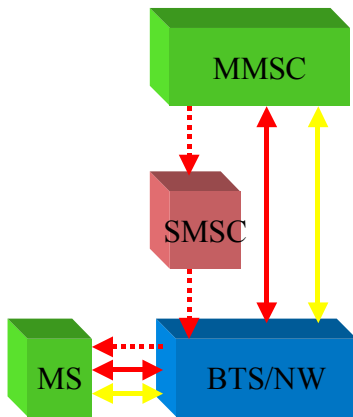
| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 27 | 48.455404 | 192.168.1.2 | 192.168.1.1 | MMSE | MMS m-send-conf |
| 28 | 48.456470 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [FIN, ACK] seq=163 Ack=13044 win=163 |
| 29 | 48.714999 | 192.168.1.1 | 192.168.1.2 | TCP | 7028 > 7082 [ACK] seq=13044 Ack=164 win=164 |
| 30 | 48.735011 | 192.168.1.1 | 192.168.1.2 | TCP | 7028 > 7082 [FIN, ACK] seq=13044 Ack=164 win=164 |
| 31 | 48.735040 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7028 [ACK] seq=164 Ack=13045 win=163 |



*1 : These process is almost same as the part 'HTTP content download (Please refer the part 1 'HTTP' in detail)'. No.28 : Ack for No.27 and disconnect request from Server to MS. No.29 is Ack for No.28. No.30 is disssconnect request from MS to Server. No.31 is Ack for No.30 .

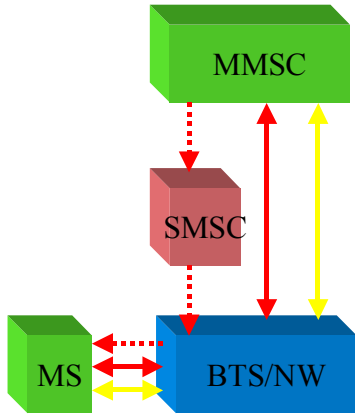
MMS Retrieve (MMS – HTTP)

Protocol Sequence



MMS Retrieve (MMS – HTTP)

Protocol Sequence (Notification (SMS))



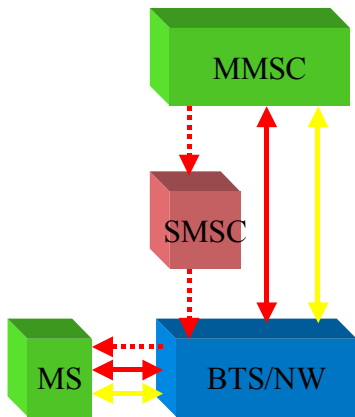
| PHY | MAC | RLC | TE | L3 | BTS | Primitive | Channel | Message |
|-----|-----|-----|----|----|-----|-------------------|---------|--------------|
| | | | | | | MAC_DATA_REQ | D DCH 3 | |
| | | | | | | PHY_DATA_REQ | D DCH 0 | |
| | | | | | | RRC_INIT_DATA_REQ | D DCH 3 | SMS MESSAGES |
| | | | | | | MAC_DATA_REQ | D DCH 3 | |
| | | | | | | MAC_DATA_REQ | D DCH 3 | |
| | | | | | | MAC_DATA_REQ | D DCH 3 | |

| Field | Value | Description |
|----------------|--------------------------|--------------|
| TP-DCS | | |
| Bit7-6 | 11 | Message |
| Bit5-4 | 11 | Data coding |
| Reserved | 0 | |
| Message coding | 1 | 8-bit data |
| Message class | 01 | Class 1 Det |
| TP-SCTS | 50504261938363 | |
| TP-UDL | 22 | Length of th |
| TP-UDH | 0B000300020205040B8423F0 | |

Please refer to the 'mms-http-retrieve.lgx' file. We can confirm 2 concatenated SMS procedure. The detail of this concatenated SMS, please refer the Part3 'Concatenated SMS'. TP-UDH is '0B 00 03 00 02 02 05 04 0B 84 23 F0'. First '0B' means total TP-UDH length. Next '00 03 00 02 02' is 2nd block of 2 concatenated SMS. Last '05 04 0B 84 23 F0' means '05' -> IEI of this information, '04' means this length, '0B 84' means the port number(2948) for destination (MS side), '23 F0' means the port number(9200) for source (Server side) The port number 2948 is used for 'WAP push'. 'WAP push' technology is used in this notification.

MMS Retrieve (MMS – HTTP)

Protocol Sequence (TCP SYN & Ack)

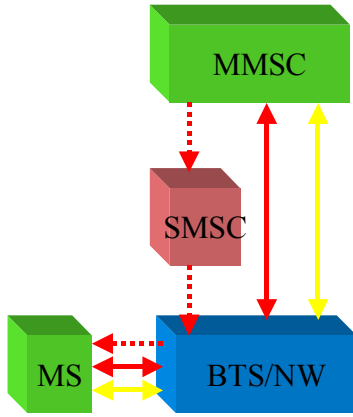


| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|--|
| 1 | 0.000000 | 192.168.1.2 | 192.168.1.2 | TCP | 7251 → 7082 [ESTAB] Seq=6640 Len=0 |
| 2 | 0.000079 | 192.168.1.2 | Broadcast | ARP | who has 192.168.1.1? If=1 192.168.1.2 |
| 3 | 0.000892 | 192.168.1.1 | 192.168.1.2 | ARP | 192.168.1.1 is at 00:00:00:00:00:00 |
| 4 | 0.009918 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7681 [SYN, ACK] Seq=6640 Ack=1 Win=1680 Len=0 |
| 5 | 0.219992 | 192.168.1.1 | 192.168.1.2 | TCP | 7681 > 7082 [ACK] Seq=6640 Ack=1 Win=32768 Len=0 |
| 6 | 0.280295 | 192.168.1.1 | 192.168.1.2 | HTTP | GET /20050524/16/4292CB6A.MMS HTTP/1.1 |
| 7 | 0.282227 | 192.168.1.2 | 192.168.1.1 | MMS | MMS m-retrieve-conf (text/plain)[Unreassemb |

*1 : These process is same as the part 'HTTP content download (Please refer to part 1 'HTTP' in detail)'.

MMS Retrieve (MMS – HTTP)

Protocol Sequence (GET (HTTP))



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---|
| 5 | 0.259992 | 192.168.1.1 | 192.168.1.2 | TCP | 7681 > 7082 [ACK] Seq=1 Ack=1 win=39200 Len=0 |
| 6 | 0.280205 | 192.168.1.2 | 192.168.1.1 | HTTP | GET /20050524/16/4292CBB6A.MMS HTTP/1.1 |
| 7 | 0.282227 | 192.168.1.2 | 192.168.1.1 | MIME | MMS m-retrieve-conf (text/plain) |

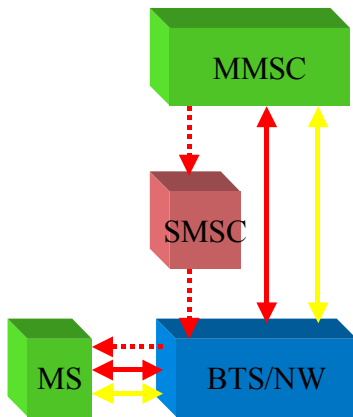
```

Frame 6 (276 bytes on wire, 276 bytes captured)
  Ethernet II, Src: 00:00:09:03:19:15, Dst: 00:10:71:00:56:b9
  Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
  Transmission Control Protocol, Src Port: 7681 (7681), Dst Port: 7082 (7082), Seq: 1, Ack: 1, Len: 222
  Hypertext Transfer Protocol
    GET /20050524/16/4292CBB6A.MMS HTTP/1.1\r\n
    Request Method: GET
    Request URI: /20050524/16/4292CBB6A.MMS
    Request Version: HTTP/1.1
    Connection: keep-alive\r\n
    Host: 192.168.1.2:7082\r\n
    User-Agent: Samsung Mocha Messenger 1.5\r\n
    Profile: http://wap.samsungobile.com/uaprof/2105UAPProf.rdf\r\n
    x-vodafone-sapcontext: yes\r\n
    \r\n
  
```

*1 : We can confirm the 'URI(path of MMS content)', Host. 'GET' means the 'request for downloading MMS'.

MMS Retrieve (MMS – HTTP)

Protocol Sequence (m-retrieve-conf)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---|
| 6 | 0.280295 | 192.168.1.1 | 192.168.1.2 | HTTP | GET /20050524/16/4292CBB6A.MMS HTTP/1.1 |
| 8 | 0.282271 | 192.168.1.2 | 192.168.1.1 | HTTP | continuation or non-HTTP traffic |

```

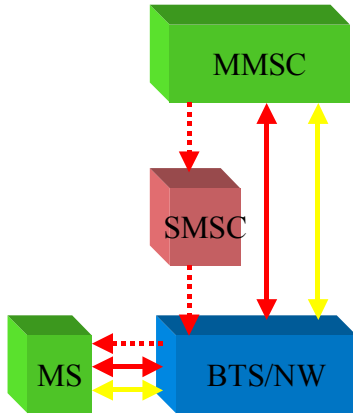
Transmission Control Protocol, Src Port: 7082 (7082), Dst Port: 7681 (7681), Seq: 1, Ack: 223, Len: 140
  Hypertext Transfer Protocol
    HTTP/1.0 200 OK\r\n
    Request Version: HTTP/1.0
    Response Code: 200
    Content-Length: 19941\r\n
    Content-Type: application/vnd.wap.mms-message\r\n
    Connection: close\r\n
    \r\n
  MMS Message Encapsulation, Type: m-retrieve-conf
    X-Mms-Message-Type: m-retrieve-conf (0x84)
    X-Mms-MMS-Version: 1.0
    Date: May 24, 2005 06:39:38.000000000
    Subject: demo
    From: 4444444444444444
    X-Mms-delivery-Report: No (0x81)
    X-Mms-Read-Report: No (0x81)
    X-Mms-Priority: Normal (0x81)
    X-Mms-Message-Class: Personal (0x80)
    Content-Type: application/vnd.wap.multipart.mixed
    Data (Post)
      Multipart body
        Part: 1, content-type: text/plain
        Content-Type: text/plain
        Headers
  
```

*1 : HTTP 200 OK (OK for the request of MMS content download)

*2 : we can confirm 'm-retrieve-conf' message, 'Subject', 'From', 'Delivery Report', 'Read Report', 'Priority', e.t.c.. 'Data' field is MMS content data itself (divided many blocks and sending: not completed yet)

MMS Retrieve (MMS – HTTP)

Protocol Sequence (downloading MMS)

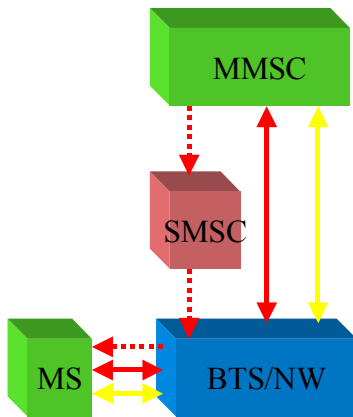


| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|--|
| 7 | 0.262227 | 192.168.1.2 | 192.168.1.1 | MMS | MMS retrieve-conf (text/plain) Unpasscmd |
| 8 | 0.262271 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 9 | 0.619947 | 192.168.1.1 | 192.168.1.2 | TCP | 7681 > 7082 [ACK] Seq=223 Ack=2801 Win=3780 |
| 10 | 0.620015 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 11 | 0.620049 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 12 | 0.620075 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 13 | 0.959922 | 192.168.1.1 | 192.168.1.2 | TCP | 7681 > 7082 [ACK] Seq=223 Ack=5601 Win=3780 |
| 14 | 0.959972 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 15 | 0.960007 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 16 | 0.960033 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 17 | 1.279896 | 192.168.1.1 | 192.168.1.2 | TCP | 7681 > 7082 [ACK] Seq=223 Ack=8401 Win=3780 |
| 18 | 1.279963 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 19 | 1.279996 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 20 | 1.280021 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 21 | 1.339895 | 192.168.1.1 | 192.168.1.2 | TCP | 7681 > 7082 [ACK] Seq=223 Ack=11201 Win=378 |
| 22 | 1.339946 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 23 | 1.339983 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 24 | 1.340009 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 25 | 1.619884 | 192.168.1.1 | 192.168.1.2 | TCP | 7681 > 7082 [ACK] Seq=223 Ack=14001 Win=378 |
| 26 | 1.619920 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 27 | 1.699863 | 192.168.1.1 | 192.168.1.2 | TCP | 7681 > 7082 [ACK] Seq=223 Ack=16801 Win=378 |
| 28 | 1.700651 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7681 [FIN, ACK] Seq=20050 Ack=223 Win=0 Len=0 |

*1 : 'Continuation or non-HTTP traffic' means 'MMS Data downloading (from Server to MS)'. We can confirm the TCP Ack (from MS to Server) for each 3 data. (MS side is not smooth enough, so we can confirm the Ack for each 3 data. But this is just our guess of this behavior.)

MMS Retrieve (MMS – HTTP)

Protocol Sequence (TCP FIN, Ack & RST)



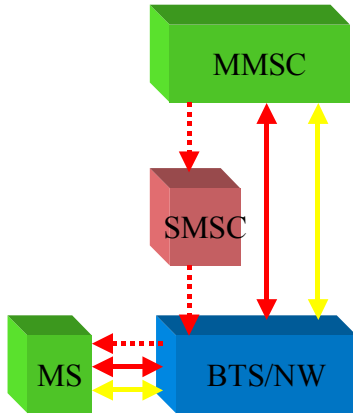
| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|--|
| 27 | 1.699863 | 192.168.1.1 | 192.168.1.2 | HTTP | Continuation or non-HTTP traffic |
| 28 | 1.700651 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7681 [FIN, ACK] Seq=20050 Ack=223 Win=0 Len=0 |
| 29 | 1.759854 | 192.168.1.1 | 192.168.1.2 | TCP | 7681 > 7082 [ACK] Seq=223 Ack=19601 Win=178 |
| 30 | 1.879857 | 192.168.1.1 | 192.168.1.2 | TCP | 7681 > 7082 [FIN, ACK] Seq=223 Ack=20050 Win=0 Len=0 |
| 31 | 1.879912 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7681 [ACK] Seq=20051 Ack=224 Win=165 |
| 32 | 1.899851 | 192.168.1.1 | 192.168.1.2 | TCP | 7681 > 7082 [RST] Seq=224 Ack=19594 Win=0 Len=0 |
| 33 | 3.579721 | 192.168.1.1 | 192.168.1.2 | TCP | 7682 > 7082 [SYN] Seq=0 Ack=0 Win=39200 Len=0 |

*1 : These process is almost same as the part 'HTTP content download (Please refer the part 1 'HTTP' in detail)'. No.27 is Ack for No.26. No.28 is disconnect request from Server to MS and Ack for No.27. No.29 is Ack for No.28. No.30 is disconnect request from MS to Server. No.31 is Ack for No.30. No.32 is disconnect.

MMS Retrieve (MMS – HTTP)

Protocol Sequence (TCP SYN & Ack)

| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|--|
| 32 | 1.899851 | 192.168.1.1 | 192.168.1.2 | TCP | 7681 > 7082 [RST] Seq=224 Ack=1959464178 wi |
| 33 | 3.579721 | 192.168.1.1 | 192.168.1.2 | TCP | 7682 > 7082 [SYN] Seq=0 Ack=0 wfin=39200 Len |
| 34 | 3.579902 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7682 [SYN, ACK] Seq=0 Ack=1 wi=6590 |
| 35 | 3.819730 | 192.168.1.1 | 192.168.1.2 | TCP | 7682 > 7082 [ACK] Seq=1 Ack=1 wfin=39200 Len |
| 36 | 3.880126 | 192.168.1.1 | 192.168.1.2 | MMSSE | MMS m-notifyresp-ind |



*1 : These process is same as the part 'HTTP content download (Please refer to part 1 'HTTP' in detail)'.

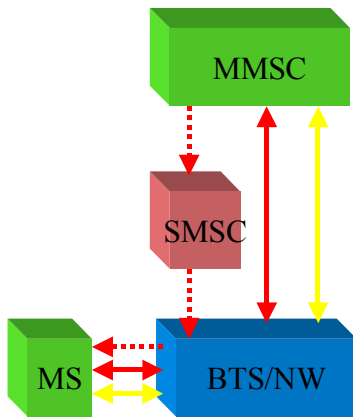
MMS Retrieve (MMS – HTTP)

Protocol Sequence (m-notifyresp-ind)

| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|--|
| 35 | 3.819730 | 192.168.1.1 | 192.168.1.2 | TCP | 7682 > 7082 [ACK] Seq=1 Ack=1 wfin=39200 Len |
| 37 | 3.882454 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.0 200 OK |


```

Frame 36 (345 bytes on wire, 345 bytes captured)
  Ethernet II, Src: 00:00:91:03:19:15, Dst: 00:10:71:00:56:b9
  Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
  Transmission Control Protocol, Src Port: 7682 (7682), Dst Port: 7082 (7082), Seq: 1, Ack: 1, Len: 291
  Hypertext Transfer Protocol
    POST /xst-test HTTP/1.1\r\n
    Request Method: POST
    Request URI: /xst-test
    Request version: HTTP/1.1
    Connection: keep-alive\r\n
    Host: 192.168.1.2:7082\r\n
    User-Agent: Samsung Mocha Messenger 1.5\r\n
    Content-type: application/vnd.wap.mms-message\r\n
    Content-Length:18\r\n
    Profile: http://wap.samsungmobile.com/uaprof/Z105UAPProf.rdf\r\n
    X-VODAFONE-3GPPContext: yes\r\n
    \r\n
  MMS Message Encapsulation, Type: m-notifyresp-ind
    X-Mms-Message-Type: m-notifyresp-ind (0x83)
    X-Mms-Transaction-ID: 4292CB7
    X-Mms-MMS-version: 1.0
    Status: Retrieved (0x81)
    X-Mms-Report-Allowed: No (0x81)
    
```

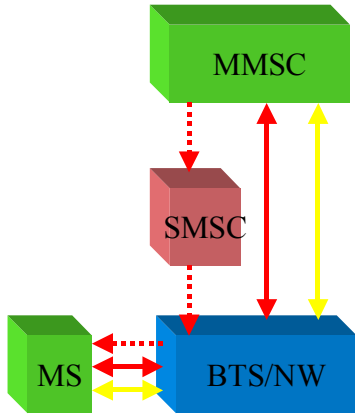


*1 : 'POST' means 'sending from MS to Server'. 'xst=test' means 'UserID=xst, Password=test' for MMSC server. 'HOST' is '192.168.1.2(IP address), 7082(port number)'.

*2 : Message Type = 'm-notifyresp-ind (Ack of Notification(SMS))' And we can confirm 'Status(Retrieved)', e.t.c . .

MMS Retrieve (MMS – HTTP)

Protocol Sequence (HTTP OK)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---|
| 36 | 3.880126 | 192.168.1.1 | 192.168.1.2 | MMSE | MMS m-notifysp-ind |
| 37 | 3.882593 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.0 200 OK |
| 38 | 3.882593 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7682 [FIN, ACK] Seq=77 Ack=292 wln=1 |

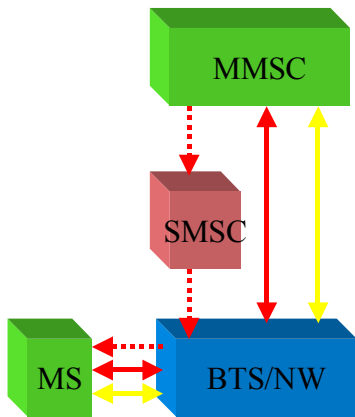

```

Frame 37 (130 bytes on wire, 130 bytes captured)
Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:91:03:19:15
Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
Hypertext Transfer Protocol
  HTTP/1.0 200 OK\r\n
  Request Version: HTTP/1.0
  Response Code: 200
  Connection: close\r\n
  Content-Type: */*\r\n
  Content-Length: 0\r\n
  \r\n
  
```

*1 : HTTP 200 OK (OK for the 'm-notifysp-ind' (POST : HTTP layer) message)

MMS Retrieve (MMS – HTTP)

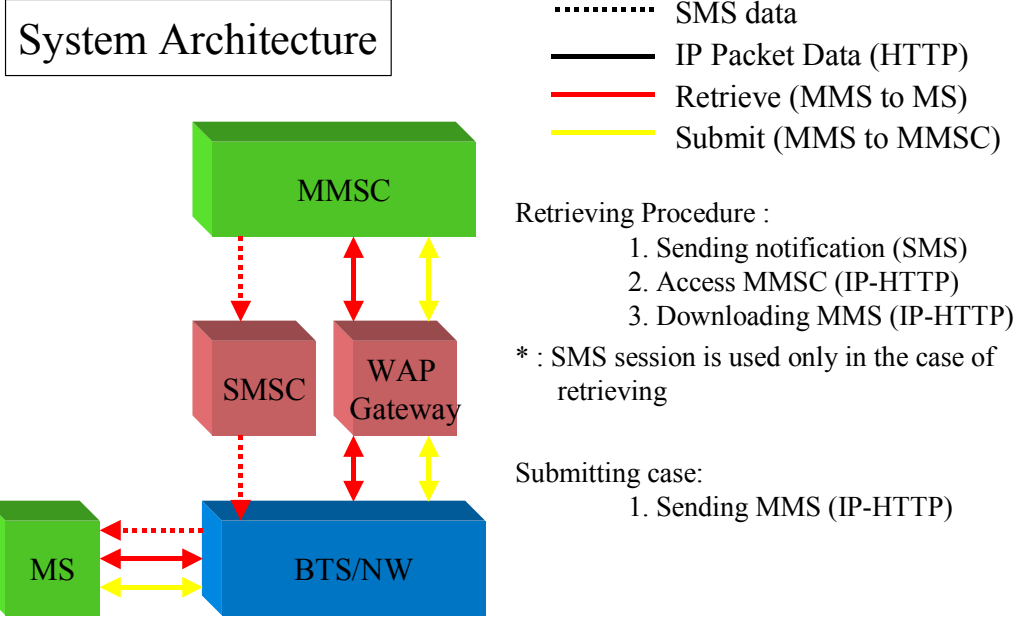
Protocol Sequence (TCP FIN, Ack & RST)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|--|
| 37 | 3.882593 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.0 200 OK |
| 38 | 3.882593 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7682 [FIN, ACK] Seq=77 Ack=292 wln=1 |
| 39 | 4.139678 | 192.168.1.1 | 192.168.1.2 | TCP | 7682 > 7082 [FIN, ACK] Seq=292 Ack=77 wln=1 |
| 40 | 4.139710 | 192.168.1.2 | 192.168.1.1 | TCP | 7082 > 7682 [ACK] Seq=78 Ack=293 wln=16503 |
| 41 | 4.159684 | 192.168.1.1 | 192.168.1.2 | TCP | 7682 > 7082 [RST] Seq=293 Ack=3566415876 wln=1 |

*1 : These process is almost the same as the part 'HTTP content download (Please refer to part 1 'HTTP' in detail)'. No.38 is disconnect request from Server to MS. No.39 is Ack for No.38 and disssconnect request from MS to Server. No.40 is Ack for No.39 . No.41 is disconnect.

MMS (MMS – WAP)

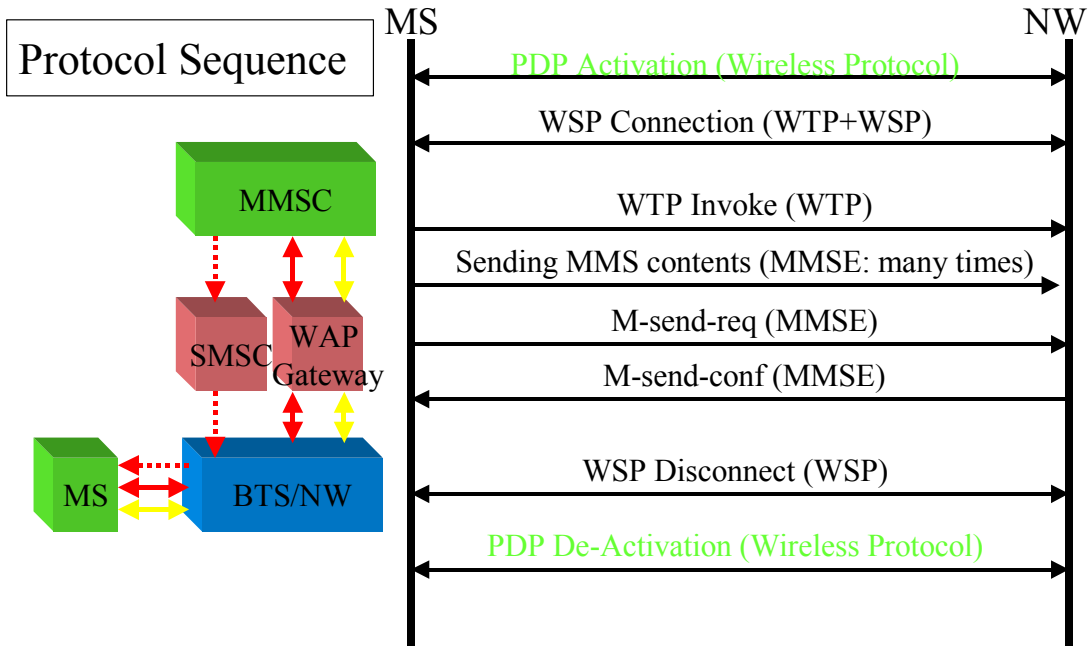


Discover What's Possible™
 MD8470A-E-E-1

Slide 119



MMS Submit (MMS – WAP)



Discover What's Possible™
 MD8470A-E-E-1

Slide 120

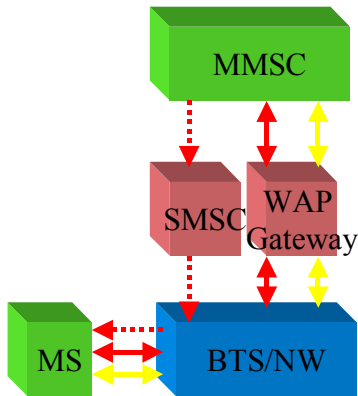


MMS Submit (MMS – WAP)

Protocol Sequence (WSP Connection)

| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|-------------------------|
| 1 | 0.000000 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WSP Connect (0x01) *1 |
| 2 | 0.026592 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WSP ConnectReply (0x02) |
| 3 | 0.299743 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Ack |

*1 : No.1 is 'WSP Connect' request (from MS to Server).
 No.2 is 'WSP ConnectReply' (from Server to MS). No.3 is Ack for No.2 . (Please refer to part1 'WAP content download' in detail.)



MMS Submit (MMS – WAP)

Protocol Sequence (WTP Invoke (WTP))

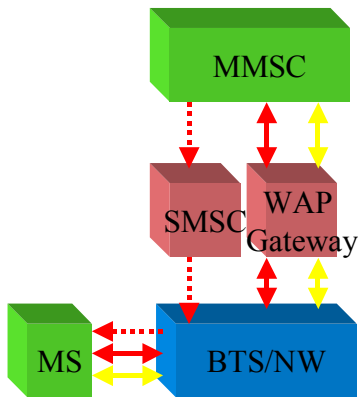
| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---|
| 3 | 0.299743 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Ack |
| 4 | 0.661813 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Invoke Class 2 (Unreassembled Fragment 0) |
| 5 | 0.661813 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP segmented invoke (1) (Unreassembled Fragment 1) |

```

# Frame 4 (1392 bytes on wire (1392 bytes captured)
# Ethernet II, Src: 00:00:91:03:19:15, Dst: 00:10:71:00:56:b9
# User Datagram Protocol, Src Port: 49152, Dst Port: 9201 (9201)
# Wireless Transaction Protocol, Src Port: 49152, Dst Port: 9201
# Transaction class: reliable invoke with reliable result (0)
1... .. = Continue Flag: TPI Present
.000 1... = PDU Type: Invoke (0x01) *1
....00.. = Trailer Flags: Not last packet (0x00)
.....0 = Retransmission indicator: First transmission
0... .. = TID Response: Original
.000 0000 0000 0110 = Transaction ID: 0x0006
00... .. = Version: Current (0x00)
..0... .. = TIDNew: TID is valid
...1... = U/P Flag: User Acknowledgement required
....00.. = Reserved: 0x00
....10.. = Transaction Class: reliable Invoke with reliable result (0x02)
# TPI: option (0x02)
# Reassembled in: 16 *2
# Payload
    
```

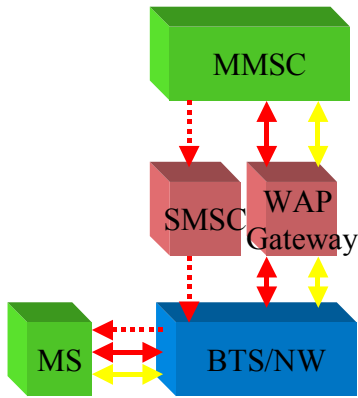
*1 : PDU Type : 'Invoke(request to upper layer)' This data is not last packet (This is the 1st block of the sending MMS data).

*2 : 'Payload' means 'sending MMS data'. The detail is described in No.16 (Reassembled in : 16).



MMS Submit (MMS – WAP)

Protocol Sequence (Sending MMS contents)



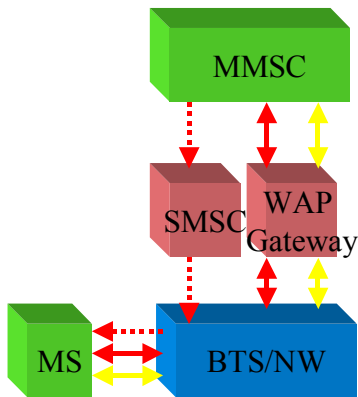
| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---|
| 4 | 0.481709 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Invoke Class 2 (Unreassembled Fragment 0) |
| 5 | 0.661813 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Segmented Invoke (1) (Unreassembled Fragment 1) |
| 6 | 0.821683 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Segmented Invoke (2) (Unreassembled Fragment 2) |
| 7 | 0.937014 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP ACK |
| 8 | 1.241653 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Segmented Invoke (3) (Unreassembled Fragment 3) |
| 9 | 1.421633 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Segmented Invoke (4) (Unreassembled Fragment 4) |
| 10 | 1.581630 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Segmented Invoke (5) (Unreassembled Fragment 5) |
| 11 | 1.596469 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP ACK |
| 12 | 2.001664 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Segmented Invoke (6) (Unreassembled Fragment 6) |
| 13 | 2.181568 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Segmented Invoke (7) (Unreassembled Fragment 7) |
| 14 | 2.249469 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Segmented Invoke (8) (Unreassembled Fragment 8) |
| 15 | 2.357851 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP ACK |
| 16 | 2.700757 | 192.168.1.1 | 192.168.1.2 | MMSE | MMS m-send-req (JPEG JFIF image) (text/plain) |

*1 : ‘Segmented Invoke’ means ‘MMS Data sending (from MS to Server)’. We can confirm the WTP Ack (from Server to MS) for once in each 3 data. (‘How often is this Ack’ is depending on the MS.)

*2 : ‘segmented Invoke (8)’ is the last packet of MMS data.

MMS Submit (MMS – WAP)

Protocol Sequence (m-send-req (MMSE))



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---|
| 15 | 2.357851 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP ACK |
| 16 | 2.400749 | 192.168.1.1 | 192.168.1.2 | MMSE | MMS m-send-req (JPEG JFIF image) (text/plain) |
| 17 | 3.153436 | 192.168.1.2 | 192.168.1.1 | MMSE | MMS m-send-conf |

```

Wireless Transaction Protocol, PDU: Segmented Invoke (5), Packet Sequence Number: 9
0... .. = Continue Flag: No TPI
.010 i... = PDU Type: Segmented Invoke (0x05)
.... ..0 = Trailer Flags: Last packet of message (0x01)
.... ..0 = Re-transmission Indicator: First transmission
0... .. = TID Response: original
.000 0000 0000 0110 = Transaction ID: 0x0006
Packet Sequence Number: 9

[WTP Fragments]
[Frame: 4, payload: 0-1341 (1342 bytes)]
[Frame: 5, payload: 1342-2687 (1346 bytes)]
[Frame: 6, payload: 2688-4033 (1346 bytes)]
[Frame: 8, payload: 4034-5379 (1346 bytes)]
[Frame: 9, payload: 5380-6725 (1346 bytes)]
[Frame: 10, payload: 6726-8071 (1346 bytes)]
[Frame: 12, payload: 8072-9417 (1346 bytes)]
[Frame: 13, payload: 9418-10763 (1346 bytes)]
[Frame: 14, payload: 10764-12109 (1346 bytes)]
[Frame: 16, payload: 12110-12930 (821 bytes)]

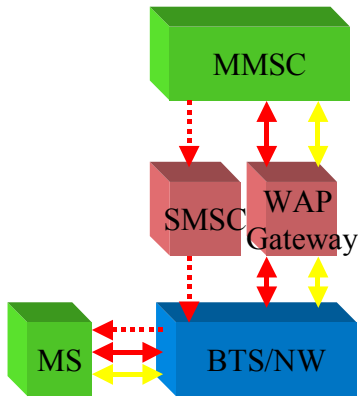
Wireless Session Protocol, Method: Post (0x60), URI: http://192.168.1.2:7082/xst-test, Content-Type: ap
PDU Type: Post (0x60)
URI Length: 32
URI: http://192.168.1.2:7082/xst-test
Headers Length: 129
Content-Type: application/vnd.wap.mms-message
Headers
    
```

*1 : we can confirm the result for sending MMS data in each data block.

*2 : PDU Type : ‘Post (sending from MS to Server)’ And we can confirm the URI (Http://192.168.1.2:7082/xst=test).

MMS Submit (MMS – WAP)

Protocol Sequence (m-send-req (MMSE))



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|--|
| 15 | 2.357851 | 192.168.1.2 | 192.168.1.1 | WTF+WSP | WTF Ack |
| 16 | 3.153436 | 192.168.1.2 | 192.168.1.1 | MMSE | MMS m-send-req (JPEG JFIF image) (text/plain) *1 |
| 17 | 3.153436 | 192.168.1.2 | 192.168.1.1 | MMSE | MMS m-send-conf |

```

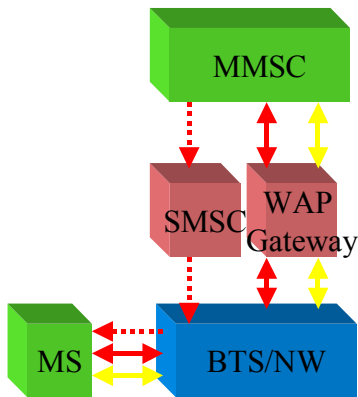
# Frame 14, payload: 10764-15109 (1346 bytes)
# Frame 16, payload: 13210-13930 (821 bytes)
# Wireless Session Protocol, Method: Post (0x60), URI: http://192.168.1.2:7082/xst-test, content-Type: ap
PDU Type: Post (0x60)
URI: http://192.168.1.2:7082/xst-test
URI Length: 32
Headers Length: 128
Content-Type: application/vnd.wap.mms-message
# Headers
# MMS Message Encapsulation, Type: m-send-req
X-Mms-Message-Type: m-send-req (0x80)
X-Mms-Transaction-ID: 1116948939.21084
X-Mms-MMS-version: 1.0
Date: May 25, 2005 01:35:38.000000000
From: <insert address>
To: 3333333/TYP=PLMN
Subject: M
X-Mms-Message-Class: Personal (0x80)
X-Mms-Priority: Normal (0x81)
Sender-visibility: show (0x81)
X-Mms-delivery-report: no (0x81)
X-Mms-Read-Report: no (0x81)
Content-Type: application/vnd.wap.multipart.related; type=application/sml; start=<1116948939>
Type: application/sml
Start: <1116948939>
# Data (Post) *2
    
```

*1 : Message Type = 'm-send-req (sending MMS)' And we can confirm 'Class', 'Priority', 'deliver report', 'read report', e.t.c. . In the case of 'MMS-Submit(HTTP)', at first 'm-send-req' then 'MMS data' itself, but in the case of 'MMS-Submit(WAP)', at first sending 'MMS data' itself (segmented Invoke), then 'm-send-req'.

*2 : we can confirm the MMS data in detail here.

MMS Submit (MMS – WAP)

Protocol Sequence (m-send-conf (MMSE))



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---|
| 16 | 2.700737 | 192.168.1.1 | 192.168.1.2 | MMSE | MMS m-send-req (JPEG JFIF image) (text/plain) |
| 17 | 4.159468 | 192.168.1.1 | 192.168.1.2 | WTF+WSP | WTF Ack *2 |
| 18 | 4.159468 | 192.168.1.1 | 192.168.1.2 | WTF+WSP | WTF Ack |

```

# Frame 17 (140 bytes on wire, 140 bytes captured)
# Ethernet II, Src: 00:10:71:00:56:b9, Dst: 00:00:91:03:19:15
# Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
# User Datagram Protocol, Src Port: 9201 (9201), Dst Port: 49152 (49152)
# Wireless Transaction Protocol, PDU: Result (2)
0... .. = Continue Flag: No TPI
.001 0... = PDU Type: Result (0x02)
.... .01 = Trailer Flags: Last packet of message (0x01)
.... .0 = Re-transmission Indicator: First transmission
1... .. = TID Response: Response
.000 0000 0000 0110 = Transaction ID: 0x0006
# Wireless Session Protocol, Method: Reply (0x04), Status: 200 OK (0x20), Content-Type: application/vnd.wap.
PDU Type: Reply (0x04)
Status: 200 OK (0x20) *1
# Headers
# MMS Message Encapsulation, Type: m-send-conf
X-Mms-Message-Type: m-send-conf (0x81)
X-Mms-Transaction-ID: 1116948939.21084
X-Mms-MMS-version: 1.0
Response-Status: ok (0x80)
Message-ID: 20050524/16/A2F32255@192.168.1.2
    
```

*1 : PDU Type : 'Reply' is response for 'POST(WSP)'.

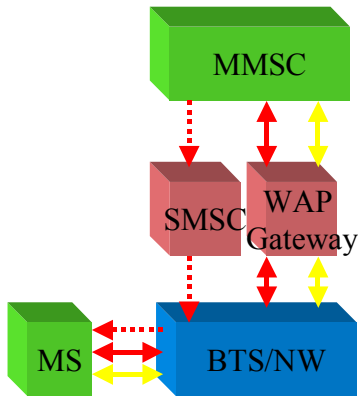
*2 : Message-Type : 'm-send-conf' is Ack for 'm-send-req'

MMS Submit (MMS – WAP)

Protocol Sequence (WSP Disconnect)

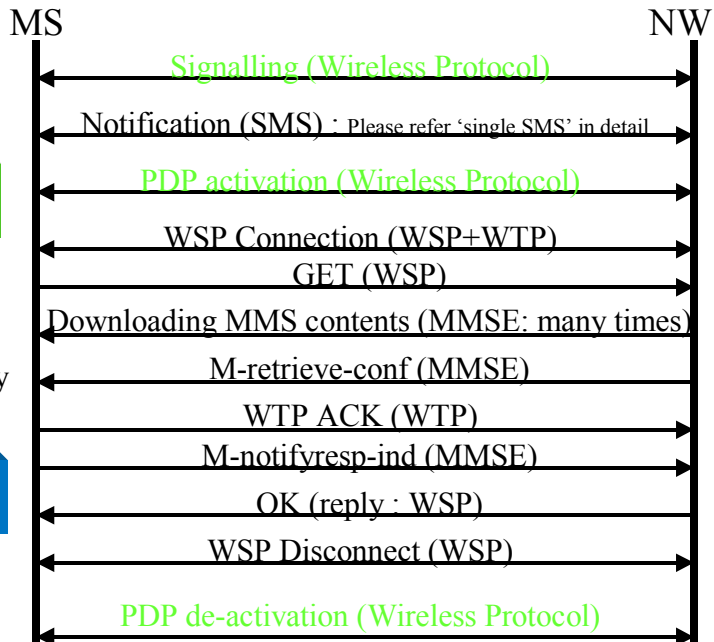
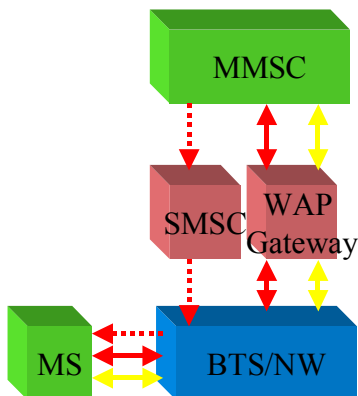
| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|--------------------------|
| 17 | 3.153436 | 192.168.1.2 | 192.168.1.1 | MMSE | MMS m-send-conf |
| 18 | 4.159468 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Ack |
| 19 | 4.169431 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WSP Disconnect (0x05) *1 |

*1 : No.18 is Ack for No.17 (WTP level). No.19 is WSP disconnect.



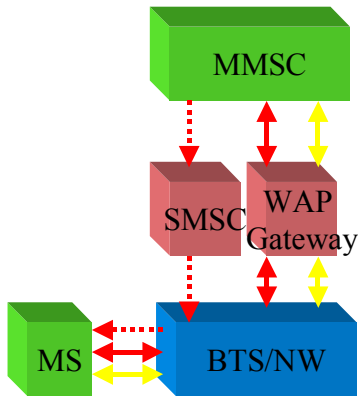
MMS Retrieve (MMS – WAP)

Protocol Sequence



MMS Retrieve (MMS – WAP)

Protocol Sequence (Notification (SMS))



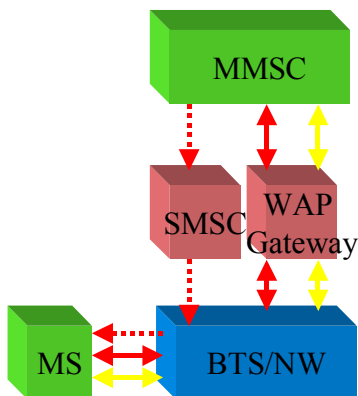
| PHY | MAC | RLC | TE | L3 | BTS | Primitive | Channel | Message |
|-----|-----|-----|----|----|-----|--------------|----------|--------------|
| | | | | | | PHY_DATA_REQ | D DCH 0 | |
| | | | | | | MAC_DATA_REQ | D DCCH 3 | SMS MESSAGES |
| | | | | | | MAC_DATA_REQ | D DCCH 3 | |
| | | | | | | MAC_DATA_REQ | D DCCH 3 | |

| Field | Value | Description |
|------------------------------------|---|-------------------|
| - Assigns bits | 00 | Assigns bits 0.5 |
| - Indicates telematic interworking | 0 | No interworking |
| BII4-0 | 00000 | |
| TP-DCS | | Message |
| - BII7-6 | 11 | Data coding/mes |
| - BII5-4 | 11 | |
| - Reserved | 0 | |
| - Message coding | 1 | 8-bit data |
| - Message class | 01 | Class 1 Default |
| TP-SCTS | 50504261736363 | |
| TP-UUDL | 140 | Length of the TP- |
| TP-UD | 0605040B8423F0F06226470706C69636174696F6E2F766... | |

Please refer to 'mms-wap-retrieve.lgx' file. We can confirm a single SMS procedure. The detail of this concatenated SMS, please refer the Part3 'Single SMS'. TP-UDH is '06 05 04 0B 84 23 F0'. First '06' means total TP-UDH length. Next '05 04 0B 84 23 F0' means '05' -> IEI of this information, '04' means this length, '0B 84' means the port number(2948) for destination (MS side), '23 F0' means the port number(9200) for source (Server side) The port number 2948 is used for 'WAP push'. 'WAP push' technology is used in this notification.

MMS Retrieve (MMS – WAP)

Protocol Sequence (WSP Connection)

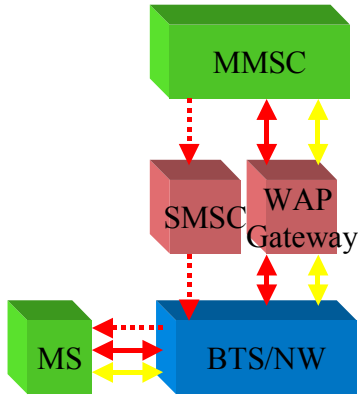


| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|--|
| 1 | 0.000000 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WSP Connect (0x01) |
| 2 | 0.014895 | 192.168.1.2 | Broadcast | ARP | who has 192.168.1.1? Tell 192.168.1.2 |
| 3 | 0.019671 | 192.168.1.1 | 192.168.1.2 | ARP | 192.168.1.1 Is at 00:00:91:03:19:15 |
| 4 | 0.019685 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Ack |
| 5 | 0.259747 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Ack |
| 6 | 0.278281 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WSP ConnectReply (0x02) |
| 7 | 0.559941 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Ack, WSP Get (0x40) http://192.168.1.2:7082/20 |

*1 : No.1 is 'WSP Connect' request (from MS to Server). No.4 is Ack for No.1. No.5 is Ack for No.4. No.6 is 'WSP ConnectReply' (from Server to MS). (Please refer to part1 'WAP content download' in detail.)

MMS Retrieve (MMS – WAP)

Protocol Sequence (GET (WSP))



```

No.  Time      Source      Destination  Protocol  Info
6 0.278281 192.168.1.2 192.168.1.1 WTP+WSP  WSP ConnectReply (0x02)
7 0.975741 192.168.1.2 192.168.1.2 WTP+WSP  WTP Ack (0x40) http://192.168.1.2:7082/20050524/16/4292CB6A.MMS
8 0.975741 192.168.1.2 192.168.1.1 WTP+WSP  WTP Result (Unreassembled Fragment 0) *1

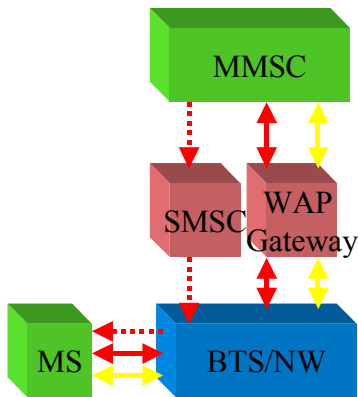
Wireless Transaction Protocol, PDU concatenation, PDU count: 3
Sub PDU size: 3
Wireless Transaction Protocol, PDU: ACK (3) *1
0... .. = Continue Flag: NO TPI
.001 1... = PDU Type: Ack (0x03)
.... 0.. = Tr/TOR Flag: False
.... 0.. = Re-transmission indicator: First transmission
0... .. = TID Response: Original
.000 0000 0000 0001 = Transaction ID: 0x0001
Sub PDU size: 151
Wireless Transaction Protocol, PDU: Invoke (1), Transaction Class: Reliable Invoke with Reliable Resu
1... .. = Continue Flag: TPI Present *1
.000 1... = PDU Type: Invoke (0x01)
.... 01.. = Trailer Flags: Last packet of message (0x01)
.... 0.. = Re-transmission indicator: First transmission
0... .. = TID Response: Original
.000 0000 0000 0010 = Transaction ID: 0x0002
00... .. = Version: Current (0x00)
..0... .. = TIDNew: TID is valid
...1... .. = U/P Flag: user Acknowledgement required
.... 00.. = Reserved: 0x00
.... ..10 = Transaction Class: Reliable Invoke with Reliable result (0x02)
TPI: option (0x02)
option value
option: Maximum group (0x04)
option value
Wireless Session Protocol, Method: Get (0x40), URI: http://192.168.1.2:7082/20050524/16/4292CB6A.MMS

```

*1 : First WTP's PDU Type is Ack for No.6(WSP Connect reply). Second WTP's PDU Type is Invoke (request for upper layer).

MMS Retrieve (MMS – WAP)

Protocol Sequence (GET (WSP))



```

No.  Time      Source      Destination  Protocol  Info
6 0.278281 192.168.1.2 192.168.1.1 WTP+WSP  WSP ConnectReply (0x02)
7 0.975741 192.168.1.2 192.168.1.2 WTP+WSP  WTP Ack (0x40) http://192.168.1.2:7082/20050524/16/4292CB6A.MMS
8 0.975741 192.168.1.2 192.168.1.1 WTP+WSP  WTP Result (Unreassembled Fragment 0) *1

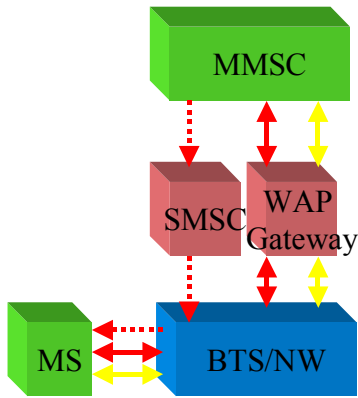
.000 0000 0000 0001 = Transaction ID: 0x0001
Sub PDU size: 151
Wireless Transaction Protocol, PDU: Invoke (1), Transaction Class: Reliable Invoke with Reliable Resu
1... .. = Continue Flag: TPI Present
.000 1... = PDU Type: Invoke (0x01)
.... 01.. = Trailer Flags: Last packet of message (0x01)
.... 0.. = Re-transmission indicator: First transmission
0... .. = TID Response: Original
.000 0000 0000 0010 = Transaction ID: 0x0002
00... .. = Version: Current (0x00)
..0... .. = TIDNew: TID is valid
...1... .. = U/P Flag: user Acknowledgement required
.... 00.. = Reserved: 0x00
.... ..10 = Transaction Class: Reliable Invoke with Reliable Result (0x02)
TPI: option (0x02)
option value
option: Maximum group (0x04)
option value
Wireless Session Protocol, Method: Get (0x40) *1
PDU Type: Get (0x40)
URI Length: 49
URI: http://192.168.1.2:7082/20050524/16/4292CB6A.MMS
Headers
User-Agent: samsung Mocha Messenger 1.5
Accept: application/vnd.wap.mms-message
Accept: */*
X-VODAFONE-3GPPDPCContext: yes

```

*1 : PDU Type : GET (request for downloading MMS content) We can confirm URI.

MMS Retrieve (MMS – WAP)

Protocol Sequence (Downloading MMS)

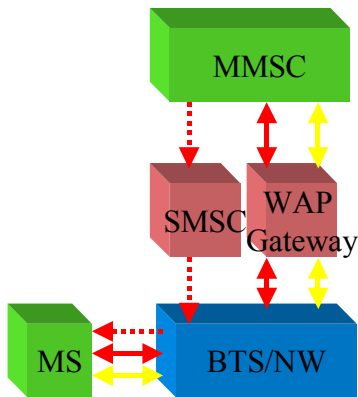


| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---|
| 7 | 0.559941 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Ack, WSP Get (0x40) http://192.168.1.2:7082/20 |
| 9 | 0.976173 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Segmented Result (1) (Unreassembled Fragment 1) |
| 10 | 0.976915 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Segmented Result (2) (Unreassembled Fragment 2) |
| 11 | 1.279675 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Ack |
| 12 | 1.290505 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Segmented Result (3) (Unreassembled Fragment 3) |
| 13 | 1.291215 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Segmented Result (4) (Unreassembled Fragment 4) |
| 14 | 1.291375 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Segmented Result (5) (Unreassembled Fragment 5) |
| 15 | 1.619673 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Ack |
| 16 | 1.646940 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Segmented Result (6) (Unreassembled Fragment 6) |
| 17 | 1.647443 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Segmented Result (7) (Unreassembled Fragment 7) |
| 18 | 1.648094 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Segmented Result (8) (Unreassembled Fragment 8) |
| 19 | 1.959614 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Ack |
| 20 | 1.980714 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Segmented Result (9) (Unreassembled Fragment 9) |
| 21 | 1.981531 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Segmented Result (10) (Unreassembled Fragment 10) |
| 22 | 1.981922 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Segmented Result (11) (Unreassembled Fragment 11) |
| 23 | 2.279391 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Ack |
| 24 | 2.296966 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Segmented Result (12) (Unreassembled Fragment 12) |
| 25 | 2.297410 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Segmented Result (13) (Unreassembled Fragment 13) |
| 26 | 2.298301 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Segmented Result (14) (Unreassembled Fragment 14) |
| 27 | 2.599368 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Ack |

*1 : 'Segmented Result' is reply for previous Invoke (WSP layer : Get). 'Payload' means 'MMS Data downloading (from Server to MS)'. We can confirm the WTP Ack (from MS to Server) for once in each 3 data. ('How often is this Ack' is depending on the MS.) This is not last Packet. 'MMS downloading' is still going on.

MMS Retrieve (MMS – WAP)

Protocol Sequence (m-retrieve-conf(MMSE))



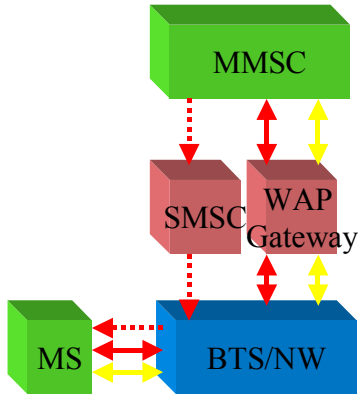
| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---|
| 49 | 4.201370 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Segmented result (31) (Unreassembled Fragment 31) |
| 50 | 4.202106 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Segmented result (32) (Unreassembled Fragment 32) |
| 51 | 4.519428 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Ack |
| 52 | 4.538998 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WTP Segmented result (33) (Unreassembled Fragment 33) |
| 53 | 4.539369 | 192.168.1.2 | 192.168.1.1 | MMSE | MMS Retrieve-conf (0x07/0110) (Spec: 0110) (Info: 0110) |
| 54 | 4.839401 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Ack |

*1 : 'Segmented Result' is reply for previous Invoke (WSP layer : Get). This is the last packet of 'MMS downloading'.

*2 : We can confirm the downloaded MMS result.

MMS Retrieve (MMS – WAP)

Protocol Sequence (m-retrieve-conf)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---|
| 49 | 4.201370 | 192.168.1.2 | 192.168.1.1 | WTP-WSP | WTP Segmented Result (31) (Unreassembled Fragment |
| 50 | 4.202106 | 192.168.1.2 | 192.168.1.1 | WTP-WSP | WTP Segmented Result (32) (Unreassembled Fragment |
| 51 | 4.519428 | 192.168.1.1 | 192.168.1.2 | WTP-WSP | WTP Ack |
| 52 | 4.538998 | 192.168.1.2 | 192.168.1.1 | WTP-WSP | WTP Segmented Result (33) (Unreassembled Fragment |
| 53 | 4.839401 | 192.168.1.1 | 192.168.1.2 | WTP-WSP | WTP Ack |

```

[Frame: 48, payload: 17240-17253 (14 bytes)]
[Frame: 49, payload: 17856-18431 (576 bytes)]
[Frame: 50, payload: 18432-19007 (576 bytes)]
[Frame: 51, payload: 19008-19583 (576 bytes)]
[Frame: 52, payload: 19584-19979 (396 bytes)]
Wireless Session Protocol, Method: Reply (0x04), Status: 200 OK (0x20), content-Type: application/vnd.w
PDU Type: Reply (0x04) *1
Status: 200 OK (0x20)
Headers Length: 36
Content-Type: application/vnd.wap.mms-message
MMS Message Encapsulation, Type: m-retrieve-conf
X-Mms-Message-Type: m-retrieve-conf (0x84)
X-Mms-MMS-Version: 1.0
Date: May 24, 2005 06:39:38.000000000
Subject: demo
From: 44444444444444
X-Mms-Delivery-Report: No (0x81)
X-Mms-Read-Report: No (0x81)
X-Mms-Priority: Normal (0x81)
X-Mms-Message-Class: Personal (0x80)
Content-Type: application/vnd.wap.multipart.mixed
Data (Post) *2
    
```

- *1 : PDU Type : 'Reply' is response for 'GET(WSP)'.
- *2 : We can confirm 'm-retrieve-conf' message. And also can confirm 'Subject', 'From', 'Deliver Report', 'Read Report', 'Priority', e.t.c.. 'Data' field is MMS content data itself. In the case of 'MMS-Retrieve(HTTP)', at first 'm-retrieve-conf' then 'MMS data' itself, but in the case of 'MMS-Retrieve(WAP)', at first sending 'MMS data' itself (WSP-GET), then 'm-retrieve-conf'.

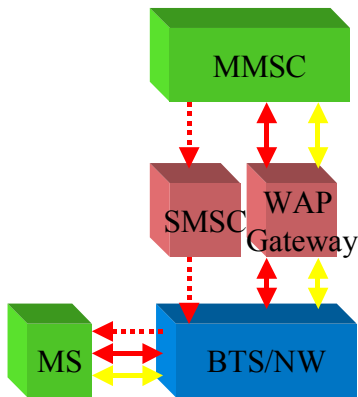
Discover What's Possible™
MD8470A-E-E-1

Slide 135

Anritsu

MMS Retrieve (MMS – WAP)

Protocol Sequence (m-notifysp-ind)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---|
| 53 | 4.539689 | 192.168.1.2 | 192.168.1.1 | MMSE | MMS m-retriev-conf (text/plain) (JPEG JFIF image, |
| 54 | 4.839401 | 192.168.1.1 | 192.168.1.2 | WTP-WSP | WTP Ack *1 |
| 55 | 6.379589 | 192.168.1.1 | 192.168.1.2 | MMSE | MMS m-notifysp-ind *3 |

```

Frame 55 (236 bytes on wire, 236 bytes captured)
Ethernet II, Src: 00:00:91:03:19:15, Dst: 00:10:71:00:56:b9
Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
User Datagram Protocol, Src Port: 49152 (49152), Dst Port: 9201 (9201)
Wireless Transaction Protocol, PDU concatenation, PDU count: 3
Sub PDU size: 3
Wireless Transaction Protocol, PDU: ACK (3)
Sub PDU size: 187
Wireless Transaction Protocol, PDU: Invoke (1), Transaction Class: Reliable Invoke with Reliable Result
Wireless Session Protocol, Method: Post (0x60), URI: http://192.168.1.2:7082/xst-test, content-Type: app
PDU Type: Post (0x60) *2
URI Length: 32
URI: http://192.168.1.2:7082/xst-test
Headers Length: 126
Content-Type: application/vnd.wap.mms-message
MMS Message Encapsulation, Type: m-notifysp-ind *3
X-Mms-Message-Type: m-notifysp-ind (0x83)
X-Mms-Transaction-ID: 4292CB87
X-Mms-MMS-Version: 1.0
Status: Retrieved (0x81)
X-Mms-Report-Allowed: No (0x81)
    
```

- *1 : This WTP Ack is for 'm-retrieve-conf'.(WTP layer).
- *2 : PDU Type : POST (from MS to Server) We can confirm the URI .
- *3 : Message Type = 'm-notifysp-ind (Ack of Notification(SMS))' And we can confirm 'Status(Retrieved)', e.t.c. .

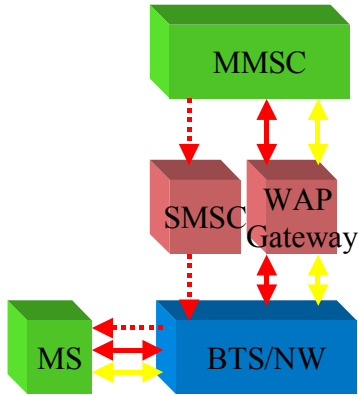
Discover What's Possible™
MD8470A-E-E-1

Slide 136

Anritsu

MMS Retrieve (MMS – WAP)

Protocol Sequence (WSP OK)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---------------------------------|
| 55 | 6.379589 | 192.168.1.1 | 192.168.1.2 | MMSE | MMS m-notifyresp-ind |
| 56 | 6.467933 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WSP Reply (0x04): 200 OK (0x20) |
| 57 | 6.719271 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Ack |

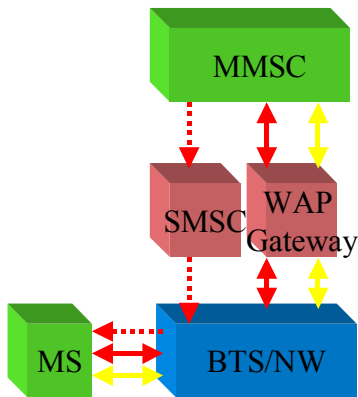
```

# Frame 56 (51 bytes on wire, 51 bytes captured)
# Ethernet II, Src: 00:10:27:1:00:56:89, Dst: 00:00:91:03:19:15
# Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
# User Datagram Protocol, Src Port: 9201 (9201), Dst Port: 49152 (49152)
# Wireless Transaction Protocol, PDU: Result (2)
# Wireless Session Protocol, Method: Reply (0x04), status: 200 OK (0x20), content-type: */*
  PDU Type: Reply (0x04) *1
  Status: 200 OK (0x20)
  Headers Length: 3
  Content-Type: */*
# Headers
    
```

*1 : WSP 200 OK (OK for the 'm-notifyresp-ind' (POST : WSP layer) message)

MMS Retrieve (MMS – WAP)

Protocol Sequence (WSP Disconnect)



| No. | Time | Source | Destination | Protocol | Info |
|-----|----------|-------------|-------------|----------|---------------------------------|
| 56 | 6.467933 | 192.168.1.2 | 192.168.1.1 | WTP+WSP | WSP Reply (0x04): 200 OK (0x20) |
| 57 | 6.719271 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WTP Ack |
| 58 | 6.729237 | 192.168.1.1 | 192.168.1.2 | WTP+WSP | WSP Disconnect (0x05) *1 |

*1 : No.57 is Ack for No.56 (WTP Layer). No.58 is WSP disconnect.

Part 6: DRM

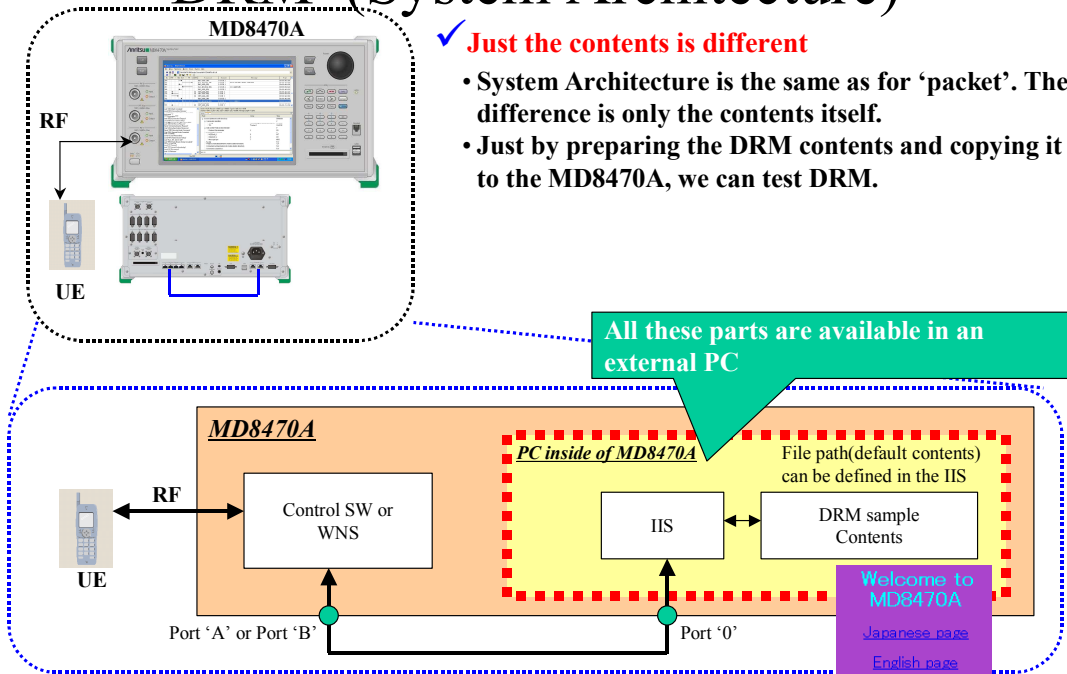
DRM (What is DRM?)

- DRM is Digital Rights Management
- The OMA (Open Mobile Alliance) promotes standardization
- DRM consists of 'Content' and a 'Rights object'
- A 'Rights object' contains information such as 'times for using' and 'term (how many days) for using'

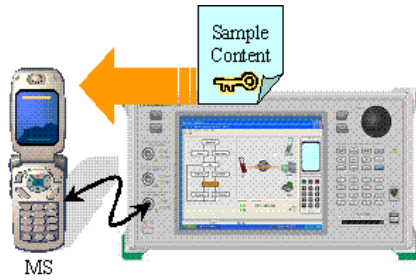
DRM (What is DRM?)

- There are 3 DRM methods:
 - Forward lock
 - This DRM contents supports only ‘cannot forward’.
 - Combined delivery lock
 - Server side replies with ‘content’ and ‘Rights object’ at the same time to MS in this method. A ‘Rights object’ support ‘can not forward’, ‘times for displaying/playing/executing/printing out’ and ‘How many days(absolute/relative expression) for displaying/playing/executing /printing out’.
 - Separate delivery lock
 - Server side replies with ‘content’ and ‘Rights object’ separately(*1). A functions of a ‘Rights object’ has is same as ‘Combined delivery lock’. Available for copying only content(‘Rights object is not allowed to forward’).
- (*1) : There are 2 ways for downloading ‘Rights Object’. One is downloading on the Packet session; the other is downloading on the SMS session (WAP Push).

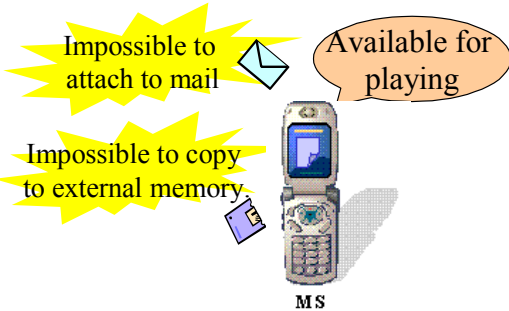
DRM (System Architecture)



DRM (Forward Lock)



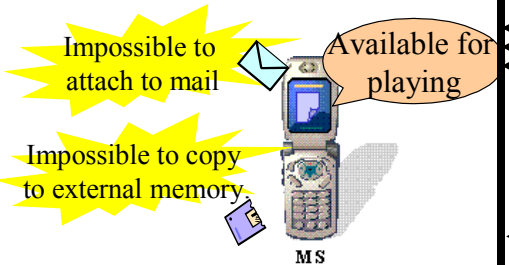
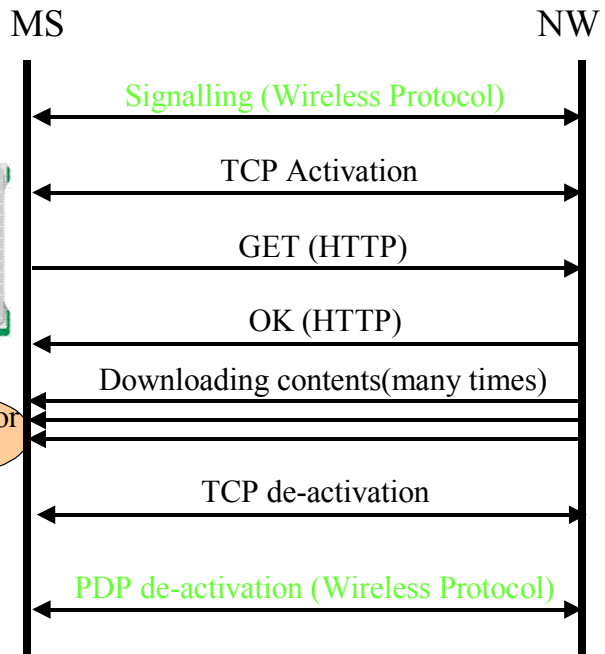
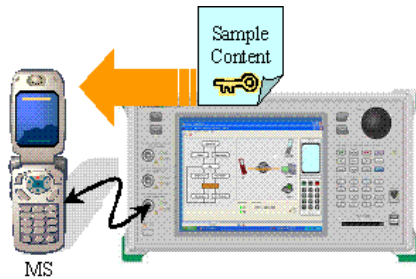
- Protocol Sequence for downloading 'Forward Lock' content is same as 'Packet communication (HTTP, WAP)' (Please refer to Part1 in detail)



- Please confirm downloaded content is 'Forward Lock' content and cannot copy/forward

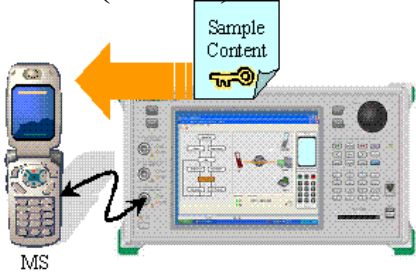
DRM (Forward Lock)

Protocol Sequence

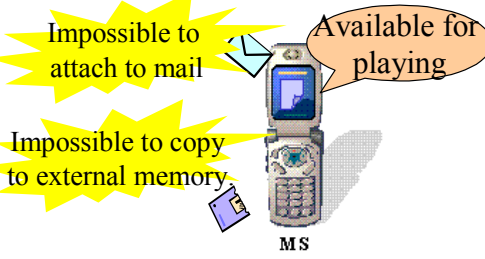


DRM (Forward Lock)

TCP Activation & Get (HTTP)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 1 | 0.000000 | 192.168.1.1 | 192.168.1.2 | TCP | 5541 > http [SYN] Seq=0 Ack=0 win=65535 Len=0 |
| 2 | 0.000078 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5541 [SYN, ACK] Seq=0 Ack=1 win=1646 |
| 3 | 0.240016 | 192.168.1.1 | 192.168.1.2 | TCP | 5541 > http [ACK] Seq=1 Ack=1 win=65535 Len=0 |
| 4 | 0.300411 | 192.168.1.1 | 192.168.1.2 | HTTP | GET http://anritsu.ne.jp/ HTTP/1.1 |
| 5 | 0.472670 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5541 [ACK] Seq=1 Ack=290 win=16175 Len=0 |
| 6 | 0.774431 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (text/html) |
| 7 | 1.499908 | 192.168.1.1 | 192.168.1.2 | TCP | 5541 > http [ACK] Seq=290 Ack=896 win=65535 |
| 8 | 22.539009 | 192.168.1.1 | 192.168.1.2 | HTTP | GET http://anritsu.ne.jp/DRM/drmm.html HTTP/1.1 |
| 9 | 22.643368 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5541 [ACK] Seq=896 Ack=591 win=15874 |
| 10 | 22.944742 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (text/html) |
| 11 | 23.498515 | 192.168.1.1 | 192.168.1.2 | TCP | 5541 > http [ACK] Seq=591 Ack=1702 win=6553 |
| 12 | 37.490731 | 192.168.1.2 | 192.168.1.1 | HTTP | GET http://anritsu.ne.jp/DRM/ForwardLock.dm HTTP/1.1 |
| 13 | 37.490731 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5541 [ACK] Seq=1702 Ack=898 win=1556 |
| 14 | 37.792096 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (application/vnd.oma.drmm.message) |
| 15 | 37.792096 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (application/vnd.oma.drmm.message) |
| 16 | 37.792134 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 17 | 37.792134 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 18 | 37.792134 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |



*1 : We can confirm the TCP connection process. (we can also confirm the .html content access for welcome pages.) In No.12, we can confirm the access for DRM content and file type is '.dm'.

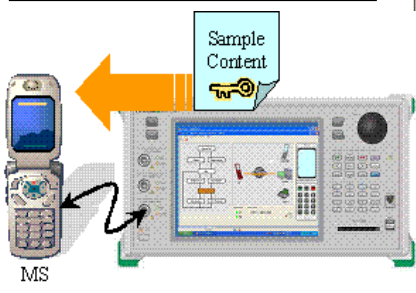
Discover What's Possible™
MD8470A-E-E-1

Slide 145

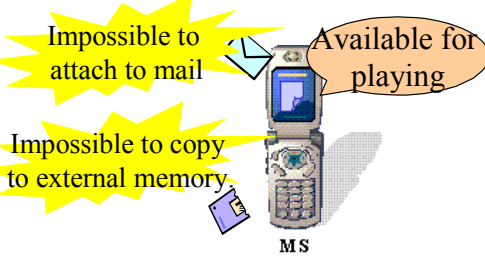


DRM (Forward Lock)

OK (HTTP)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 12 | 37.318063 | 192.168.1.1 | 192.168.1.2 | HTTP | GET http://anritsu.ne.jp/DRM/ForwardLock.dm |
| 13 | 37.490731 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5541 [ACK] Seq=1702 Ack=898 win=1556 |
| 14 | 37.792096 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (application/vnd.oma.drmm.message) |
| 15 | 37.792134 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 16 | 37.792159 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 17 | 37.792193 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 18 | 38.077595 | 192.168.1.1 | 192.168.1.2 | TCP | 5541 > http [ACK] Seq=898 Ack=4446 win=6416 |



*1 : We can confirm the downloading DRM content. We can also confirm the 'Content-Type' is DRM.

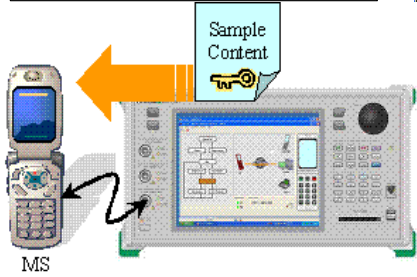
Discover What's Possible™
MD8470A-E-E-1

Slide 146

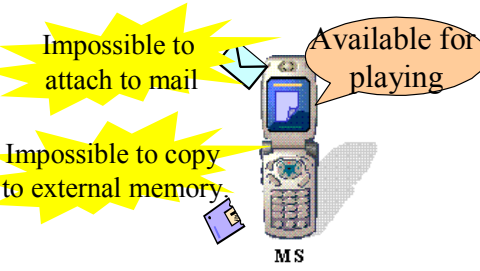


DRM (Forward Lock)

Downloading & TCP De-activation

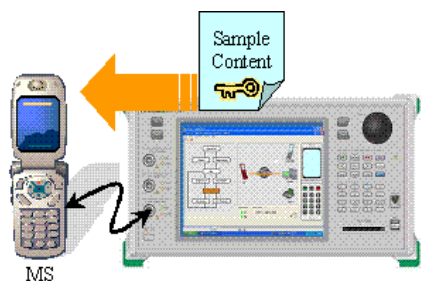


| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 42 | 39.537468 | 192.168.1.1 | 192.168.1.2 | TCP | 5541 > http [ACK] Seq=898 Ack=27650 wfin=64163 Le |
| 43 | 39.537545 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 44 | 39.537580 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 45 | 39.537603 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 46 | 39.757470 | 192.168.1.1 | 192.168.1.2 | TCP | 5541 > http [ACK] Seq=898 Ack=31746 wfin=64163 Le |
| 47 | 39.797517 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 48 | 39.797550 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 49 | 39.797575 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 50 | 39.997464 | 192.168.1.1 | 192.168.1.2 | TCP | 5541 > http [ACK] Seq=898 Ack=34470 wfin=65535 Le |
| 51 | 39.997524 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 52 | 39.997559 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 53 | 39.997582 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 54 | 40.077453 | 192.168.1.1 | 192.168.1.2 | TCP | 5541 > http [ACK] Seq=898 Ack=37214 wfin=64163 Le |
| 55 | 40.297442 | 192.168.1.2 | 192.168.1.1 | TCP | 5541 > http [ACK] Seq=898 Ack=41310 wfin=64163 Le |
| 56 | 40.297485 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 57 | 40.297519 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 58 | 40.297542 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 59 | 40.557401 | 192.168.1.1 | 192.168.1.2 | TCP | 5541 > http [ACK] Seq=898 Ack=44084 wfin=64163 Le |
| 60 | 40.997396 | 192.168.1.1 | 192.168.1.2 | TCP | 5541 > http [ACK] Seq=898 Ack=46276 wfin=65535 Le |
| 61 | 47.856954 | 192.168.1.1 | 192.168.1.2 | TCP | 5541 > http [FIN, ACK] Seq=898 Ack=46276 wfin=655 |
| 62 | 47.856992 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5541 [ACK] Seq=46276 Ack=899 wfin=3567 Le |
| 63 | 47.857198 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5541 [FIN, ACK] Seq=46276 Ack=899 wfin=35 |
| 64 | 48.006970 | 192.168.1.1 | 192.168.1.2 | TCP | 5541 > http [ACK] Seq=899 Ack=46276 wfin=65535 Le |



*1 : We can confirm the downloading DRM content. After downloading, We can also confirm the TCP disconnection process.
 * : Then please confirm the downloaded content is 'DRM (Forward Lock)' by checking the detail content information.

DRM (Combined Delivery Lock)



- Protocol Sequence for downloading 'Combined Delivery Lock' content is same as 'Packet communication(HTTP, WAP)' (Please refer to Part1 in detail)
- Please confirm downloaded content is 'Combined Delivery Lock' content and cannot copy/forward and check the limit of 'How many times' and 'until when'

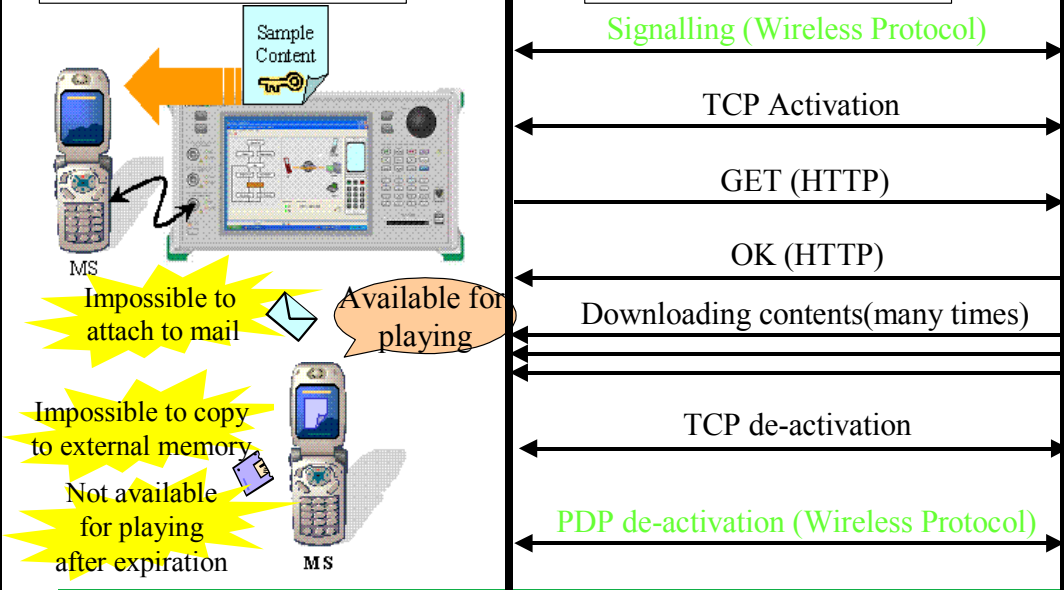
DRM (Combined Delivery Lock)

Protocol Sequence

MS

Protocol Sequence is same as
'Forward Lock method.'

NW



Discover What's Possible™
MD8470A-E-E-1

Slide 149

Anritsu

DRM (Combined Delivery Lock)

TCP Activation & Get (HTTP)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 1 | 0.000000 | 192.168.1.1 | 192.168.1.2 | TCP | 5543 > httpd [SYN] Seq=0 Ack=0 win=65535 Len=0 MS |
| 2 | 0.000050 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5543 [SYN, ACK] Seq=0 Ack=1 win=6464 Len |
| 3 | 0.239962 | 192.168.1.1 | 192.168.1.2 | TCP | 5543 > http [ACK] Seq=1 Ack=1 win=65535 Len=0 |
| 4 | 0.300437 | 192.168.1.1 | 192.168.1.2 | HTTP | GET http://anritsu.ne.jp/ HTTP/1.1 |
| 5 | 0.423316 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5543 [ACK] Seq=1 Ack=290 win=16175 Len=0 |
| 6 | 0.723129 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK [text/html] |
| 7 | 1.499903 | 192.168.1.1 | 192.168.1.2 | TCP | 5543 > http [ACK] Seq=290 Ack=896 win=65535 Len=0 |
| 8 | 6.879984 | 192.168.1.1 | 192.168.1.2 | HTTP | GET http://anritsu.ne.jp/DRM/drm.html HTTP/1.1 |
| 9 | 7.044631 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5543 [ACK] Seq=896 Ack=591 win=15874 Len=0 |
| 10 | 7.346002 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK [text/html] |
| 11 | 7.999476 | 192.168.1.1 | 192.168.1.2 | TCP | 5543 > http [ACK] Seq=591 Ack=1702 win=65535 Len=0 |
| 12 | 13.966970 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5543 [ACK] Seq=1702 Ack=899 win=15566 Len=0 |
| 14 | 14.268064 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (application/vnd.oma.drm.message |
| 15 | 14.268105 | 192.168.1.2 | 192.168.1.1 | HTTP | continuation or non-HTTP traffic |

*1 : We can confirm the TCP connection process. (we can also confirm the .html content access for welcome pages.) In No.12, we can confirm the access for DRM content and file type is '.dm'.

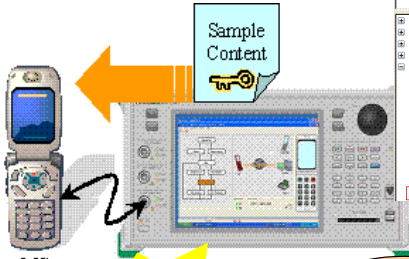
Discover What's Possible™
MD8470A-E-E-1

Slide 150

Anritsu

DRM (Combined Delivery Lock)

OK (HTTP)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|---|
| 12 | 13.89547 | 192.168.1.1 | 192.168.1.2 | HTTP | GET http://anritsu.ne.jp/DRM/CombinedLock.dm HTTP |
| 13 | 13.96670 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5543 [ACK] Seq=1702 Ack=899 win=15566 Len |
| 14 | 14.06004 | 192.168.1.2 | 192.168.1.1 | HTTP | http://anritsu.ne.jp/DRM/CombinedLock.dm message |
| 15 | 14.268105 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 16 | 14.268129 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 17 | 14.268162 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 18 | 14.599046 | 192.168.1.1 | 192.168.1.2 | TCP | 5543 > http [ACK] Seq=899 Ack=4446 win=64163 Len |


```

# Frame 14 (1426 bytes on wire, 1426 bytes captured)
# Ethernet II, Src: 00:10:71:00:6d:15, Dst: 00:00:91:04:74:3d
# Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
# Transmission Control Protocol, Src Port: http (80), Dst Port: 5543 (5543), Seq: 1702, Ack: 899, Len: 1372
# Hypertext Transfer Protocol
# HTTP/1.1 200 OK\r\n
Server: Microsoft-IIS/5.1\r\n
X-Powered-By: ASP.NET\r\n
Date: Tue, 28 Jun 2005 06:20:12 GMT\r\n
Content-Type: application/vnd.oma.drm.message\r\n
Accept-Ranges: bytes\r\n
Last-Modified: Thu, 23 Jun 2005 02:39:36 GMT\r\n
Etag: "0e456b9c77c51a68"\r\n
Content-Length: 45199\r\n
\r\n
Media Type: application/vnd.oma.drm.message (1100 bytes)
    
```

- Impossible to attach to mail
- Impossible to copy to external memory
- Not available for playing after expiration
- Available for playing

*1 : We can confirm the downloading DRM content. We can also confirm the 'Content-Type' is DRM.

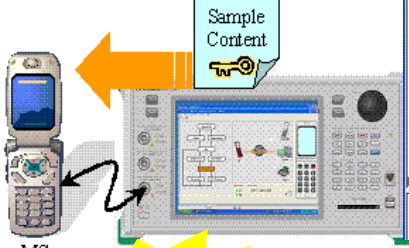
Discover What's Possible™
MD8470A-E-E-1

Slide 151



DRM (Combined Delivery Lock)

Downloading & TCP De-activation



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 44 | 16.059056 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 45 | 16.059080 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 46 | 16.318958 | 192.168.1.1 | 192.168.1.2 | TCP | 5543 > http [ACK] Seq=899 Ack=51746 win=64163 Le |
| 47 | 16.318985 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 48 | 16.319019 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 49 | 16.319042 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 50 | 16.498845 | 192.168.1.1 | 192.168.1.2 | TCP | 5543 > http [ACK] Seq=899 Ack=34470 win=65535 Le |
| 51 | 16.499007 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 52 | 16.499042 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 53 | 16.499065 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 54 | 16.598910 | 192.168.1.1 | 192.168.1.2 | TCP | 5543 > http [ACK] Seq=899 Ack=57214 win=64163 Le |
| 55 | 16.778903 | 192.168.1.1 | 192.168.1.2 | TCP | 5543 > http [ACK] Seq=899 Ack=41310 win=64163 Le |
| 56 | 16.778973 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 57 | 16.779007 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 58 | 16.779031 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 59 | 17.058881 | 192.168.1.1 | 192.168.1.2 | TCP | 5543 > http [ACK] Seq=899 Ack=44034 win=64163 Le |
| 60 | 17.058914 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 61 | 17.278865 | 192.168.1.1 | 192.168.1.2 | TCP | 5543 > http [ACK] Seq=899 Ack=47173 win=65120 Le |
| 62 | 17.498881 | 192.168.1.1 | 192.168.1.2 | TCP | [TCP window update] 5543 > http [ACK] Seq=899 Ac |
| 63 | 30.658012 | 192.168.1.1 | 192.168.1.2 | TCP | 5543 > http [FIN, ACK] Seq=899 Ack=47173 win=655 |
| 64 | 30.658066 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5543 [ACK] Seq=47173 Ack=900 win=15566 Le |
| 65 | 30.658203 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5543 [FIN, ACK] Seq=47173 Ack=900 win=155 |


```

# Frame 66 (94 bytes on wire, 94 bytes captured)
# Ethernet II, Src: 00:00:91:04:74:3d, Dst: 00:10:71:00:6d:15
# Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
# Transmission Control Protocol, Src Port: http (80), Dst Port: http (80), Seq: 900, Ack: 47174, Len: 0
    
```

- Impossible to attach to mail
- Impossible to copy to external memory
- Not available for playing after expiration
- Available for playing

*1 : We can confirm the downloading DRM content. After downloading, We can also confirm the TCP disconnection process.

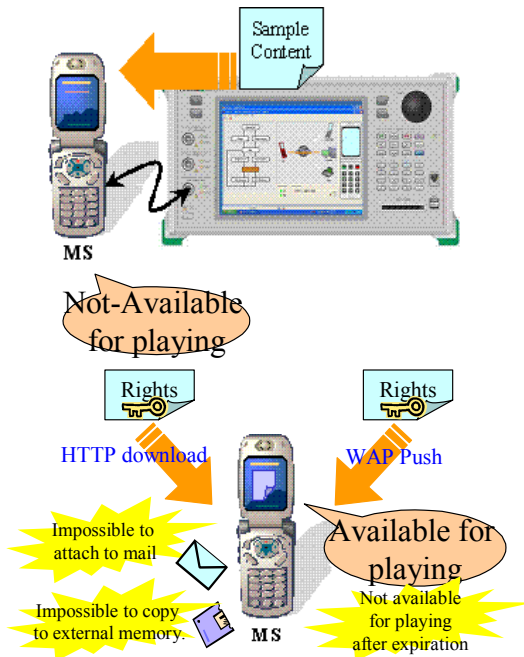
* : Then please confirm the downloaded content is 'DRM (Combined Delivery Lock)' by checking the detail content information (please check 'how many times for playing', etc).

Discover What's Possible™
MD8470A-E-E-1

Slide 152



DRM (Separate Delivery Lock)



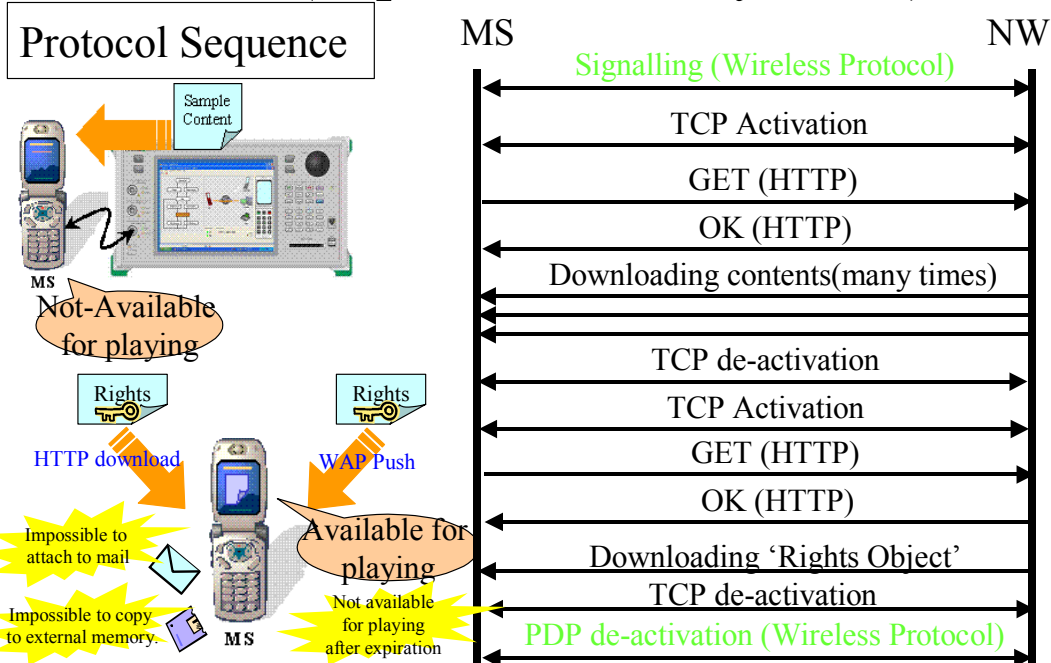
- Protocol Sequence for downloading 'Separate Delivery Lock' content is same as 'Packet communication(HTTP, WAP)' (Please refer to Part1 in detail) Please confirm 2 access(downloading), one is for 'downloading content', the other is for 'downloading Rights object'.
- Please confirm downloaded content is 'Separate Delivery Lock' content and can not copy/forward and check the limit of 'How many times' and 'until when'. And please confirm that you can copy the content, but cannot copy the 'Rights Object'.

Discover What's Possible™
MD8470A-E-E-1

Slide 153



DRM (Separate Delivery Lock)



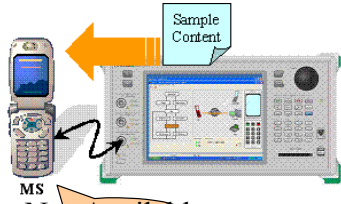
Discover What's Possible™
MD8470A-E-E-1

Slide 154



DRM (Separate Delivery Lock)

TCP Activation & Get (HTTP)



Not-Available for playing



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 1 | 0.000000 | 192.168.1.1 | 192.168.1.2 | TCP | 5549 > http [SYN] Seq=0 Ack=0 win=65535 Len=0 MS |
| 2 | 0.000054 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5549 [SYN, ACK] Seq=0 Ack=1 win=16464 Len |
| 3 | 0.239989 | 192.168.1.1 | 192.168.1.2 | TCP | 5549 > http [ACK] Seq=1 Ack=1 win=65535 Len=0 |
| 4 | 0.300409 | 192.168.1.1 | 192.168.1.2 | HTTP | GET http://anritsu.ne.jp/ HTTP/1.1 |
| 5 | 0.408330 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5549 [ACK] Seq=1 Ack=290 win=16175 Len=0 |
| 6 | 0.709929 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (text/html) |
| 7 | 1.499904 | 192.168.1.1 | 192.168.1.2 | TCP | 5549 > http [ACK] Seq=290 Ack=896 win=65535 Len= |
| 8 | 5.860075 | 192.168.1.1 | 192.168.1.2 | HTTP | GET http://anritsu.ne.jp/DRM/drm.html HTTP/1.1 |
| 9 | 6.026240 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5549 [ACK] Seq=896 Ack=591 win=15874 Len= |
| 10 | 6.327604 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (text/html) |
| 11 | 7.039548 | 192.168.1.1 | 192.168.1.2 | TCP | 5549 > http [ACK] Seq=591 Ack=1702 win=65535 Len= |
| 12 | 10.841598 | 192.168.1.2 | 192.168.1.1 | HTTP | GET http://anritsu.ne.jp/DRM/SeparateLock.dcf HTTP/1.1 |
| 13 | 10.841598 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5549 [ACK] Seq=1702 Ack=900 win=15165 Len |
| 14 | 11.142972 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (application/vnd.oma.drm.content |
| 15 | 11.143011 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |

*1 : We can confirm the TCP connection process. (we can also confirm the .html content access for welcome pages.) In No.12, we can confirm the access for DRM content and file type is '.dcf'.

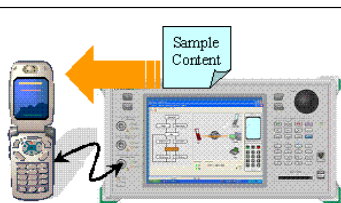
Discover What's Possible™
MD8470A-E-E-1

Slide 155

Anritsu

DRM (Separate Delivery Lock)

OK (HTTP)



Not-Available for playing



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 12 | 10.659777 | 192.168.1.1 | 192.168.1.2 | HTTP | GET http://anritsu.ne.jp/DRM/SeparateLock.dcf HT |
| 13 | 10.841598 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5549 [ACK] Seq=1702 Ack=900 win=15165 Len |
| 14 | 11.143011 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 15 | 11.143011 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 16 | 11.143035 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 17 | 11.143070 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation or non-HTTP traffic |
| 18 | 11.459268 | 192.168.1.1 | 192.168.1.2 | TCP | 5549 > http [ACK] Seq=900 Ack=4446 win=64163 Len |

*1 : We can confirm the downloading DRM content. We can also confirm the 'Content-Type' is DRM.

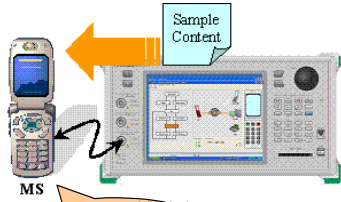
Discover What's Possible™
MD8470A-E-E-1

Slide 156

Anritsu

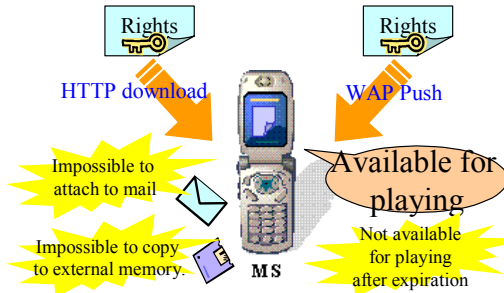
DRM (Separate Delivery Lock)

Downloading & TCP De-activation



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 44 | 13.199232 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation of non-HTTP traffic |
| 45 | 13.199268 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation of non-HTTP traffic |
| 46 | 13.199292 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation of non-HTTP traffic |
| 47 | 13.199332 | 192.168.1.1 | 192.168.1.2 | TCP | 5549 > http [ACK] Seq=900 Ack=31746 Win=64163 Len=0 |
| 48 | 13.199309 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation of non-HTTP traffic |
| 49 | 13.199241 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation of non-HTTP traffic |
| 50 | 13.199363 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation of non-HTTP traffic |
| 51 | 13.859114 | 192.168.1.1 | 192.168.1.2 | TCP | 5549 > http [ACK] Seq=900 Ack=35842 Win=64163 Len=0 |
| 52 | 13.819190 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation of non-HTTP traffic |
| 53 | 13.819227 | 192.168.1.1 | 192.168.1.2 | HTTP | Continuation of non-HTTP traffic |
| 54 | 13.819248 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation of non-HTTP traffic |
| 55 | 14.039100 | 192.168.1.1 | 192.168.1.2 | TCP | 5549 > http [ACK] Seq=900 Ack=38566 Win=65535 Len=0 |
| 56 | 14.039153 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation of non-HTTP traffic |
| 57 | 14.039188 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation of non-HTTP traffic |
| 58 | 14.039211 | 192.168.1.2 | 192.168.1.1 | HTTP | Continuation of non-HTTP traffic |
| 59 | 14.139091 | 192.168.1.1 | 192.168.1.2 | TCP | 5549 > http [ACK] Seq=900 Ack=41310 Win=64163 Len=0 |
| 60 | 14.139081 | 192.168.1.1 | 192.168.1.2 | TCP | 5549 > http [ACK] Seq=900 Ack=45416 Win=64163 Len=0 |
| 61 | 15.039031 | 192.168.1.1 | 192.168.1.2 | TCP | 5549 > http [ACK] Seq=900 Ack=46474 Win=65535 Len=0 |
| 62 | 25.378389 | 192.168.1.1 | 192.168.1.2 | TCP | 5549 > http [FIN, ACK] Seq=900 Ack=46473 Win=65535 Len=0 |
| 63 | 25.378429 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5549 [ACK] Seq=6473 Ack=901 Win=15565 Len=0 |
| 64 | 25.378647 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5549 [FIN, ACK] Seq=6473 Ack=901 Win=15565 Len=0 |
| 65 | 25.658352 | 192.168.1.1 | 192.168.1.2 | TCP | 5549 > http [EST, ACK] Seq=901 Ack=46473 Win=15565 Len=0 |
| 66 | 25.483343 | 192.168.1.1 | 192.168.1.2 | TCP | 5549 > http [RST, ACK] Seq=901 Ack=46474 Win=15565 Len=0 |

MS
Not-Available for playing



* 1 : We can confirm the downloading DRM content. After downloading, we can also confirm the TCP disconnection process.

* : Then please confirm the downloaded content can not open, because there is no 'Rights object'.

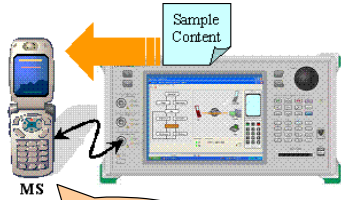
Discover What's Possible™
MD8470A-E-E-1

Slide 157

Anritsu

DRM (Separate Delivery Lock)

TCP Activation & Get (HTTP)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 65 | 25.658352 | 192.168.1.1 | 192.168.1.2 | TCP | 5549 > http [RST, ACK] Seq=901 Ack=46473 Win=15565 Len=0 |
| 66 | 25.718344 | 192.168.1.1 | 192.168.1.2 | TCP | 5549 > http [RST, ACK] Seq=901 Ack=46474 Win=15565 Len=0 |
| 67 | 45.037111 | 192.168.1.1 | 192.168.1.2 | TCP | 5550 > http [SYN] Seq=0 Ack=0 Win=65535 Len=0 MS |
| 68 | 45.037194 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5550 [SYN, ACK] Seq=0 Ack=0 Win=16464 Len=0 |
| 69 | 45.277087 | 192.168.1.1 | 192.168.1.2 | TCP | 5550 > http [ACK] Seq=0 Ack=1 Win=65535 Len=0 |
| 70 | 45.337547 | 192.168.1.1 | 192.168.1.2 | HTTP | GET http://rights-issuer.com/content/HTTP/1.1 |
| 71 | 45.452010 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5550 [ACK] Seq=1 Ack=305 Win=16160 Len=0 |
| 72 | 45.753352 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 302 Object Moved (text/html) |
| 73 | 46.317483 | 192.168.1.1 | 192.168.1.2 | HTTP | GET http://rights-issuer.com/content/HTTP/1.1 |
| 74 | 46.455211 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5550 [ACK] Seq=304 Ack=610 Win=15855 Len=0 |
| 75 | 46.759923 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (text/html) |
| 76 | 47.536668 | 192.168.1.2 | 192.168.1.2 | TCP | 5550 > http [ACK] Seq=610 Ack=1020 Win=65535 Len=0 |
| 77 | 55.584314 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5550 [ACK] Seq=1020 Ack=926 Win=64039 Len=0 |
| 79 | 55.885709 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK [application/vnd.oma.drms.rights+... |

MS
Not-Available for playing



* 1 : We can confirm the TCP connection process. (we can also confirm the .html content access for welcome pages.) In No.77, we can confirm the access for DRM Rights object and file type is '.dr'.

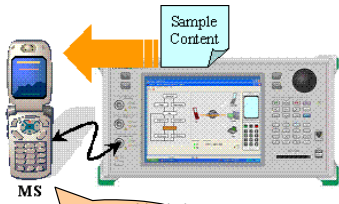
Discover What's Possible™
MD8470A-E-E-1

Slide 158

Anritsu

DRM (Separate Delivery Lock)

OK (HTTP)



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 73 | 46.317483 | 192.168.1.1 | 192.168.1.2 | HTTP | GET http://rights-issuer.com/content/ HTTP/1.1 |
| 74 | 46.452311 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5550 [ACK] Seq=304 Ack=610 Win=13835 Len= |
| 75 | 46.756923 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (text/html) |
| 76 | 47.536968 | 192.168.1.1 | 192.168.1.2 | TCP | 5550 > http [ACK] Seq=610 Ack=1020 Win=65535 Len= |
| 77 | 55.416998 | 192.168.1.1 | 192.168.1.2 | HTTP | GET http://rights-issuer.com/content/Separate.dr |
| 78 | 55.584324 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5550 [ACK] Seq=1020 Ack=926 Win=15539 Len= |
| 79 | 55.536391 | 192.168.1.1 | 192.168.1.2 | HTTP | HTTP/1.1 200 OK (application/vnd.oma.drm.rights+xml) |
| 80 | 56.536391 | 192.168.1.1 | 192.168.1.2 | TCP | 5550 > http [ACK] Seq=926 Ack=2235 Win=65535 Len= |
| 81 | 64.875832 | 192.168.1.1 | 192.168.1.2 | TCP | 5550 > http [FIN, ACK] Seq=926 Ack=2235 Win=65535 Len= |

Frame 79 (1269 bytes on wire, 1269 bytes captured)
 # Ethernet II, Src: 00:10:71:00:6d:15, dst: 00:00:91:04:74:3d
 # Internet Protocol, Src Addr: 192.168.1.2 (192.168.1.2), Dst Addr: 192.168.1.1 (192.168.1.1)
 # Transmission Control Protocol, Src Port: http (80), Dst Port: 5550 (5550), Seq: 1020, Ack: 926, Len: 1215
 # Hypertext Transfer Protocol
 # HTTP/1.1 200 OK\r\n
 Server: Microsoft-IIS/5.1\r\n
 x-Powered-By: ASP.NET\r\n
 Date: Tue, 28 Jun 2005 07:40:07 GMT\r\n
 Content-Type: application/vnd.oma.drm.rights+xml\r\n * 1
 Accept-Ranges: bytes\r\n
 Last-Modified: Tue, 28 Jun 2005 05:50:15 GMT\r\n
 ETag: "60281742a37bc51:a68"\r\n
 Content-Length: 941\r\n
 \r\n
 Media Type: application/vnd.oma.drm.rights+xml (941 bytes) * 1

Not-Available for playing



* 1 : We can confirm the downloading DRM Rights object. We can also confirm the 'Content-Type' is 'Rights+xml'. Rights object is small enough for downloading at once, so there is no more packet in the trace log.

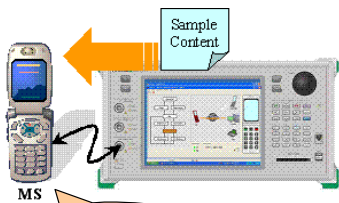
Discover What's Possible™
MD8470A-E-E-1

Slide 159



DRM (Separate Delivery Lock)

TCP De-activation



| No. | Time | Source | Destination | Protocol | Info |
|-----|-----------|-------------|-------------|----------|--|
| 77 | 55.416998 | 192.168.1.1 | 192.168.1.2 | HTTP | GET http://rights-issuer.com/content/Separate.dr |
| 78 | 55.584324 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5550 [ACK] Seq=1020 Ack=926 Win=15539 Len= |
| 79 | 55.887009 | 192.168.1.2 | 192.168.1.1 | HTTP | HTTP/1.1 200 OK (application/vnd.oma.drm.rights+ |
| 80 | 56.536391 | 192.168.1.1 | 192.168.1.2 | TCP | 5550 > http [ACK] Seq=926 Ack=2235 Win=65535 Len= |
| 81 | 64.875832 | 192.168.1.1 | 192.168.1.2 | TCP | 5550 > http [FIN, ACK] Seq=926 Ack=2235 Win=65535 Len= |
| 82 | 64.875898 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5550 [ACK] Seq=2235 Ack=927 Win=15539 Len= |
| 83 | 64.876044 | 192.168.1.2 | 192.168.1.1 | TCP | http > 5550 [FIN, ACK] Seq=2235 Ack=927 Win=15539 Len= |
| 84 | 65.135892 | 192.168.1.1 | 192.168.1.2 | TCP | 5550 > http [RST, ACK] Seq=927 Ack=2235 Win=15539 Len= |
| 85 | 65.135893 | 192.168.1.1 | 192.168.1.2 | TCP | 5550 > http [RST, ACK] Seq=927 Ack=2235 Win=15539 Len= |

Frame 85 (94 bytes on wire, 94 bytes captured)
 # Ethernet II, Src: 00:00:91:04:74:3d, Dst: 00:10:71:00:6d:15
 # Internet Protocol, Src Addr: 192.168.1.1 (192.168.1.1), Dst Addr: 192.168.1.2 (192.168.1.2)
 # Transmission Control Protocol, Src Port: 5550 (5550), Dst Port: http (80), Seq: 927, Ack: 2236, Len: 0

Not-Available for playing



* 1 : We can confirm the TCP disconnection process, after Rights object downloading.

* : Then please confirm the downloaded content is 'DRM (Separate Delivery Lock)' by checking the detail content information (please check 'how many times for playing', etc). And please confirm you can copy the content (but you cannot copy the 'Rights object').

Discover What's Possible™
MD8470A-E-E-1

Slide 160



Anritsu

Specifications are subject to change without notice.

ANRITSU CORPORATION

1800 Onna, Atsugi-shi, Kanagawa, 243-8555 Japan
Phone: +81-46-223-1111
Fax: +81-46-296-1264

● U.S.A.

ANRITSU COMPANY TX OFFICE SALES AND SERVICE

1155 East Collins Blvd., Richardson, TX 75081, U.S.A.
Toll Free: 1-800-ANRITSU (267-4878)
Phone: +1-972-644-1777
Fax: +1-972-644-3416

● Canada

ANRITSU ELECTRONICS LTD.
700 Silver Seven Road, Suite 120, Kanata,
ON K2V 1C3, Canada
Phone: +1-613-591-2003
Fax: +1-613-591-1006

● Brasil

ANRITSU ELETRÔNICA LTDA.
Praça Amadeu Amaral, 27 - 1 andar
01327-010 - Paraisópolis, Sao Paulo, Brazil
Phone: +55-11-3283-2511
Fax: +55-11-3886940

● U.K.

ANRITSU LTD.

200 Capability Green, Luton, Bedfordshire LU1 3LU, U.K.
Phone: +44-1582-433280
Fax: +44-1582-731303

● Germany

ANRITSU GmbH

Grafenberger Allee 54-56, 40237 Düsseldorf, Germany
Phone: +49-211-96855-0
Fax: +49-211-96855-55

● France

ANRITSU S.A.

9, Avenue du Québec Z.A. de Courtabœuf 91951 Les
Ulis Cedex, France
Phone: +33-1-60-92-15-50
Fax: +33-1-64-46-10-65

● Italy

ANRITSU S.p.A.

Via Elio Vittorini, 129, 00144 Roma EUR, Italy
Phone: +39-06-509-9711
Fax: +39-06-502-2425

● Sweden

ANRITSU AB

Borgafjordsgatan 13 164 40 Kista, Sweden
Phone: +46-853470700
Fax: +46-853470730

● Finland

ANRITSU AB

Teknobulevardi 3-5, FI-01530 Vantaa, Finland
Phone: +358-9-4355-220
Fax: +358-9-4355-2250

● Denmark

Anritsu AB Danmark

Korskildelund 6 DK - 2670 Greve, Denmark
Phone: +45-36915035
Fax: +45-43909371

● Singapore

ANRITSU PTE LTD.

10, Hoe Chiang Road #07-01/02, Keppel Towers,
Singapore 089315
Phone: +65-6282-2400
Fax: +65-6282-2533

● Hong Kong

ANRITSU COMPANY LTD.

Suite 923, 9/F., Chinachem Golden Plaza, 77 Mody
Road, Tsimshatsui East, Kowloon, Hong Kong, China
Phone: +852-2301-4980
Fax: +852-2301-3545

● P. R. China

ANRITSU COMPANY LTD.

Beijing Representative Office

Room 1515, Beijing Fortune Building, No. 5 North
Road, the East 3rd Ring Road, Chao-Yang District
Beijing 100004, P.R. China
Phone: +86-10-6590-9230

● Korea

ANRITSU CORPORATION

8F Hyun Juk Bldg. 832-41, Yeoksam-dong,
Kangnam-ku, Seoul, 135-080, Korea
Phone: +82-2-553-6603
Fax: +82-2-553-6604

● Australia

ANRITSU PTY LTD.

Unit 3/170 Forster Road Mt. Waverley, Victoria, 3149,
Australia
Phone: +61-3-9558-8177
Fax: +61-3-9558-8255

● Taiwan

ANRITSU COMPANY INC.

7F, No. 316, Sec. 1, NeiHu Rd., Taipei, Taiwan
Phone: +886-2-8751-1816
Fax: +886-2-8751-1817

050203