Nways Multiprotocol Routing Services

Using and Configuring Features Version 3.3

SC30-3992-01

Note
Before using this document, read the general information under “Notices” on page xix.

Second Edition (June 1999)

This edition applies to Version 3.3 of the IBM Nways Multiprotocol Routing Services and to all subsequent releases and modifications until otherwise indicated in new editions or technical newsletters.

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

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

Department CGF
Design & Information Development
IBM Corporation
P.O. Box 12195
RESEARCH TRIANGLE PARK NC 27709
USA

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

Contents

Figures

Tables

Notices

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

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785, USA.

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

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

Notice to Users of Online Versions of This Book

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

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

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

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

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

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

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

Advanced Peer-to-Peer Networking	IBM	PS/2
AIX	Micro Channel	RS/6000
AIXwindows	NetView	System/370
APPN	AS/400	Nways
VTAM	BookManager	

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

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

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

Preface

This manual contains the information that you will need to use the router user interface for configuration and operation of the features installed on your Nways device. A specific Nways device might not support all of the features described in this manual. If a feature is device-specific, you are informed of that by:

- A notice in the relevant chapter or section
- A section in the preface that lists the features and the devices that support them

This manual supports the IBM 2210 and refers to it as either a “router” or a “device”. The examples in the manual represent the configuration of an IBM 2210, but the actual output you see may vary. Use the examples as a guideline to what you might see while configuring your device.

Who Should Read This Manual

This manual is intended for persons who install and manage computer networks. Although experience with computer networking hardware and software is helpful, you do not need programming experience to use the protocol software.

To get additional information: Changes may be made to the documentation after the books are printed. If additional information is available or if changes are required after the books have been printed, the changes will be in a file (named README) on diskette 1 of the configuration program diskettes. You can view the file with an ASCII text editor.

About the Software

IBM Nways Multiprotocol Routing Services is the software that supports the IBM 2210 (licensed program number 5801-ARR). This software has these components:

- The base code, which consists of:
  - The code that provides the routing, bridging, data link switching, and SNMP agent functions for the device.
  - The router user interface, which allows you to configure, monitor, and use the Multiprotocol Routing Services base code installed on the device. The router user interface is accessed locally through an ASCII terminal or emulator attached to the service port, or remotely through a Telnet session or modem-attached device.

  The base code is installed at the factory on the 2210.
- The Configuration Program for IBM Nways Multiprotocol Routing Services (referred to in this book as the *Configuration Program*) is a graphical user interface that enables you to configure the device from a stand-alone workstation. The Configuration Program includes error checking and online help information.

  The Configuration Program is not pre-loaded at the factory; it is shipped separately from the device as part of the software order.

  You can also obtain the Configuration Program for IBM Nways Multiprotocol Routing Services from the IBM Networking Technical Support home page. See *Configuration Program User’s Guide for Nways Multiprotocol and Access Services Products,* GC30-3830, for the server address and directories.

Conventions Used in This Manual

The following conventions are used in this manual to show command syntax and program responses:

1. The abbreviated form of a command is underlined as shown in the following example:

   ```
   reload
   ```

   In this example, you can enter either the whole command (reload) or its abbreviation (rel).

2. Keyword choices for a parameter are enclosed in brackets and separated by the word or. For example:

   ```
   command [keyword1 or keyword2]
   ```

   Choose one of the keywords as a value for the parameter.

3. Three periods following an option mean that you enter additional data (for example, a variable) after the option. For example:

   ```
   time host ...
   ```

   In this example, you enter the IP address of the host in place of the periods, as explained in the description of the command.

4. In information displayed in response to a command, defaults for an option are enclosed in brackets immediately following the option. For example:

   ```
   Media (UTP/STP) [UTP]
   ```

   In this example, the media defaults to UTP unless you specify STP.

5. Keyboard key combinations are indicated in text in the following ways:
   - **Ctrl-P**
   - **Ctrl -**

   The key combination **Ctrl -** indicates that you should press the Ctrl key and the hyphen simultaneously. In certain circumstances, this key combination changes the command line prompt.

6. Names of keyboard keys are indicated like this: **Enter**

7. Variables (that is, names used to represent data that you define) are denoted by italics. For example:

   File Name: *filename.ext*

IBM 2210 Nways Multiprotocol Router Publications

Library reorganization: Beginning with Version 3.2, the following changes to the organization of the library took place:

- The information in the *Software User's Guide* titled **Understanding, Using and Configuring Features** was moved into a new manual, *Using and Configuring Features*.
- The chapters on using, configuring, and monitoring the DIALs feature were moved into the *Using and Configuring Features* book.

Information updates and corrections: To keep you informed of engineering changes, clarifications, and fixes that were implemented after the books were printed, refer to the IBM 2210 home pages at:

http://www.networking.ibm.com/220/220prod.html

The following list shows the books that support the IBM 2210.

Operations and Network Management

SC30-3681
Software User's Guide

This book explains how to:

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

SC30-3992
Using and Configuring Features

SC30-3680
Protocol Configuration and Monitoring Reference Volume 1

SC30-3865
Protocol Configuration and Monitoring Reference Volume 2

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

They include information about each of the protocols that the devices support.

SC30-3682
Event Logging System Messages Guide

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

Configuration

Online help
The help panels for the Configuration Program assist the user in understanding the program functions, panels, configuration parameters, and navigation keys.

GC30-3830
Configuration Program User's Guide for Nways Multiprotocol and Access Services Products

This book discusses how to use the Configuration Program.

GG24-4446
IBM 2210 Nways Multiprotocol Router Description and Configuration Scenarios

This book contains examples of how to configure protocols using IBM Nways Multiprotocol Routing Services.

Safety

SD21-0030
Caution: Safety Information - Read This First

This book provides translations of caution and danger notices applicable to the installation and maintenance of an IBM 2210.

The following list shows the books in the IBM 2210 Nways Multiprotocol Router library, arranged according to tasks.

Planning and Installation

GA27-4068
IBM 2210 Introduction and Planning Guide

GC30-3867
IBM 2210 Nways Multiprotocol Router Installation and Initial Configuration Guide

These books are shipped with the 2210. They explain how to prepare for installation, install the 2210, perform an initial configuration, and verify that the installation is successful.

These books provide translations of danger notices and other safety information.

Diagnostics and Maintenance

SY27-0345
IBM 2210 Nways Multiprotocol Router Service and Maintenance Manual

This book is shipped with the 2210. It provides instructions for diagnosing problems with and repairing the 2210.

Summary of Changes for the IBM 2210 Software Library

The following list applies to changes in the software that were made in Version 3.3. The changes consist of:

- **New functions:**
  - Encoding subsystem (ES)
  - Dynamic Host Configuration Protocol (DHCP) services
  - Virtual private network (VPN)
    - Directory services: Lightweight Directory Access Protocol (LDAP) support
    - ISAKMP/Oakley support
    - Layer 2 Forwarding (L2F)
    - Point to Point Tunneling protocol (PPTP)
    - Differentiated Services
  - Support of J2 6 Mbps for maximum for Frame Relay CIR, Bc, and Be
  - Frame Relay packet fragmentation
  - Voice packet forwarding over Frame Relay
- **Enhanced functions:**
  - IP enhancements
    - Generic IPv4 routing policy
    - IPv6 packet filters, dynamic reconfiguration, and DHCP relay agent support
  - SDLC enhancements
    - Primary group poll
    - Two-way simultaneous communication

- DLSw configuration parameters to allow control of the number of non-session messages queued in the router
- TN3270 enhancements
  - Host-initiated dynamic LU definition
  - Multiple PU SAs over DLSw
- Bridging enhancement
  - IPX SR-TB support
- X.25 dynamic reconfiguration support
- IPX enhancements
  - Configurable RIP ticks
  - IPXWAN over Frame Relay SVCs
- Command completion function of the command line interface

- **Clarifications and corrections**

  The technical changes and additions are indicated by a vertical line (|) to the left of the change.

Getting Help

At the command prompts, you can obtain help in the form of a listing of the commands available at that level. To do this, type **?** (the **help** command), and then press **Enter**. Use **?** to list the commands that are available from the current level. You can usually enter a **?** after a specific command name to list its options.

Exiting a Lower Level Environment

The multiple-level nature of the software places you in secondary, tertiary, and even lower level environments as you configure or operate the 2210. To return to the next higher level, enter the **exit** command. To get to the secondary level, continue entering **exit** until you receive the secondary level prompt (either `Config>` or `+`).

For example, to exit the ASRT protocol configuration process:

```
ASRT config> exit
Config>
```

If you need to get to the primary level (OPCON), enter the intercept character (**Ctrl-P** by default).

Summary of Changes

Chapter 1. Using Bandwidth Reservation and Priority Queuing

This chapter describes the Bandwidth Reservation System and priority queuing features currently available for Frame Relay and PPP interfaces. It includes the following sections:

Bandwidth Reservation System

The Bandwidth Reservation System (BRS) allows you to decide which packets to drop when demand (traffic) exceeds supply (throughput) on a network connection. When bandwidth utilization reaches 100%, BRS determines which traffic to drop based on your configuration.

Bandwidth reservation ″reserves″ transmission bandwidth for specified classes of traffic. Each class has an allocated minimum percentage of the connection's bandwidth. See Figure 1 on page 2 and Figure 2 on page 2.

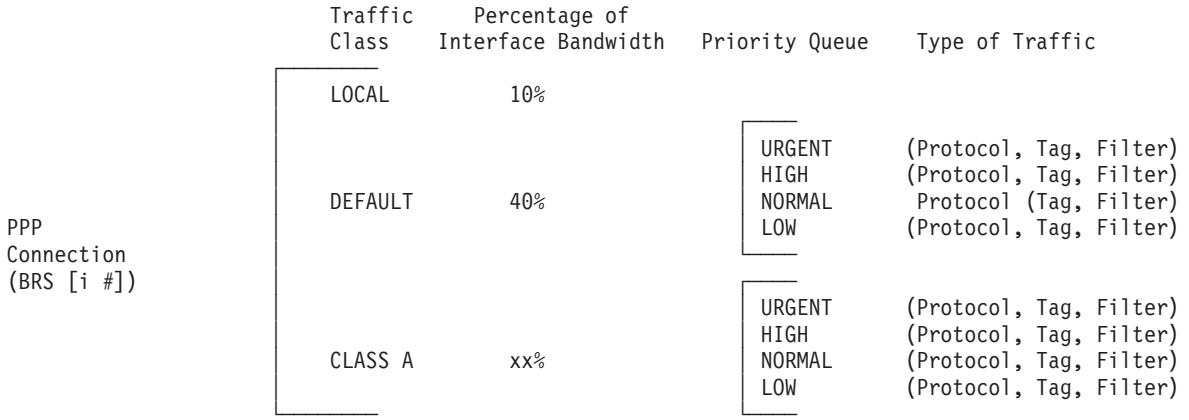
On PPP interfaces, you define traffic classes (t-classes) and each traffic class is allocated a percentage of the PPP interface's bandwidth. There are at least two traffic classes:

1. A LOCAL class which is allocated bandwidth for packets that are locally originated by the router (e.g. IP RIP packets)
2. A DEFAULT class to which all other traffic is initially assigned.

You can create additional traffic classes and assign protocols, filters and tags to the priority queues within a traffic class. See Figure 1 on page 2.

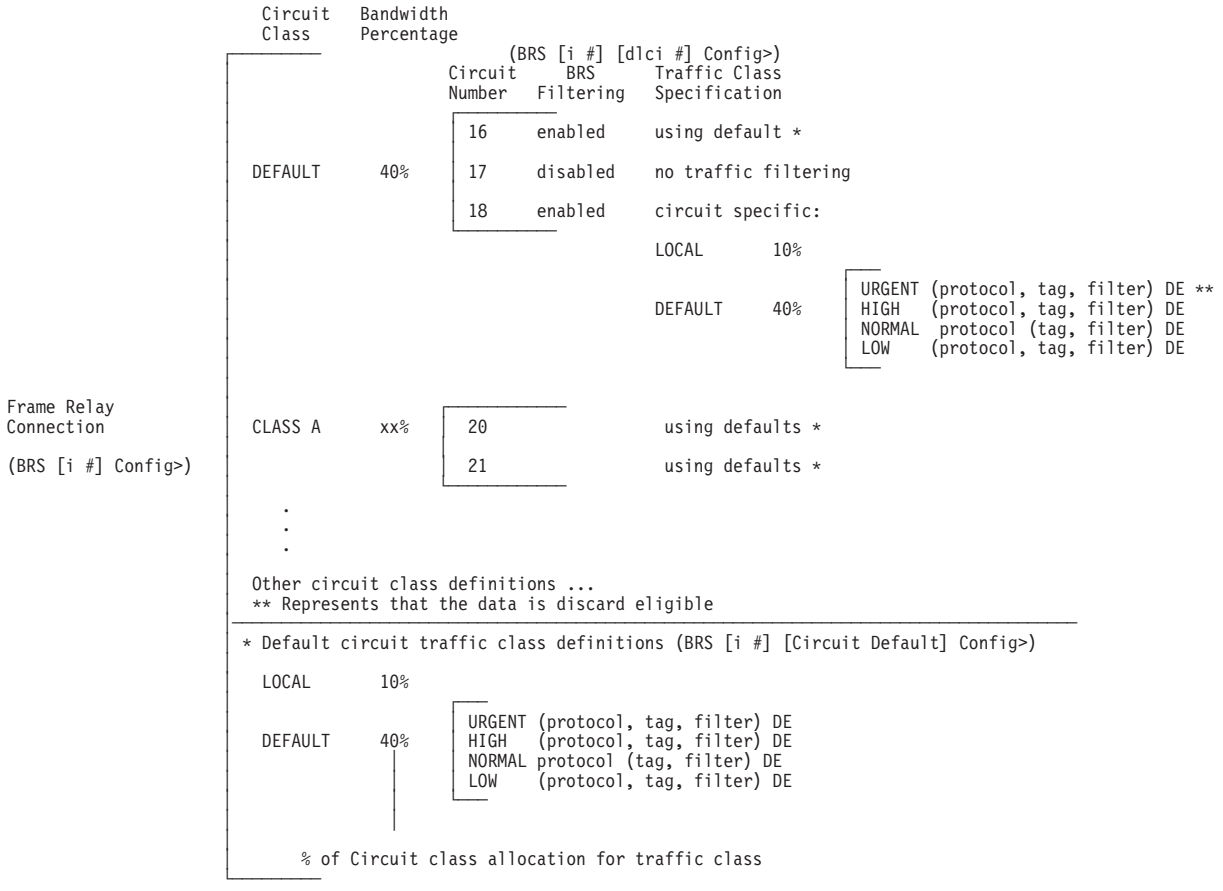
On Frame Relay interfaces, you define circuit classes (c-classes) and each circuit class is allocated a percentage of the Frame Relay interface's bandwidth. There is at least one circuit class: the DEFAULT circuit class to which all circuits are initially assigned. You can create additional circuit classes and assign circuits to these c-classes. On each Frame Relay circuit, you can define traffic classes (t-classes) and each traffic class is allocated a percentage of the Frame Relay circuit's bandwidth. The traffic class support for Frame Relay circuits is analogous to the traffic class support for PPP interfaces. See Figure 2 on page 2 for the Frame Relay Circuit Class and Traffic Class Relationships.

Using BRS and Priority Queuing



Note: All protocols are initially assigned to the NORMAL priority queue of the DEFAULT traffic class. You can assign a protocol, filter, or tag to any priority queue within a traffic class.

Figure 1. PPP BRS Traffic Class and Traffic Class Priority Queue Relationship



Note: All protocols are initially assigned to the NORMAL priority queue of the DEFAULT traffic class. You can assign a protocol, filter, or tag to any priority queue within a traffic class.

Figure 2. Frame Relay BRS Circuit Class and Traffic Class Relationship

These reserved percentages are a minimum *slice* of bandwidth for the network connection. If a network is operating to capacity, messages in any one class can be

transmitted only until they use the configured bandwidth allocated for the class. In this case, additional transmissions are held until other bandwidth transmissions have been satisfied. In the case of a light traffic path, a packet stream can use bandwidth exceeding its allowed minimum up to 100% if there is no other traffic.

Bandwidth reservation is really a *safeguard*. In general, a device should not attempt to use greater than 100% of its line speed. If it does, a faster line is probably needed. The “bursty” nature of traffic, however, can drive the requested transmission rate to exceed 100% for a short time. In these cases, bandwidth reservation is enabled and the higher priority traffic is ensured delivery (that is, is not discarded).

Bandwidth reservation runs over the following connection types:

- Frame Relay (serial line or dial circuit interface)
- PPP (serial line or dial circuit interface)

Bandwidth Reservation over Frame Relay

Bandwidth reservation allows you to reserve bandwidth at two levels:

- At the interface level, you can assign a percentage of the interface’s bandwidth to circuit classes (*c-classes*). Each circuit class contains one or more circuits.
- At the circuit level, you can define traffic classes (*t-classes*) and allocate a percentage of the circuit’s bandwidth. (A traffic class created by the **create-super-class** command is not associated with any bandwidth but always takes priority over all other t-classes defined for the circuit.)

When BRS receives a packet from Frame Relay, the configured c-classes and t-classes are used to determine when that packet will be transmitted. BRS queues the packet according to these criteria: c-class, circuit, t-class, and priority within the t-class. The c-class to which the circuit has been assigned is put onto a queue of c-classes and the queue of c-classes is sorted according to a fair weighted queuing algorithm. Within a c-class, circuits that have packets to be transmitted are serviced in a round robin fashion. The t-classes within each c-class are also sorted according to a fair weighted queuing algorithm. Within the t-class, packets are further queued according to their priority (urgent, high, normal, or low).

A packet is removed from the queue and transmitted when it meets all these criteria:

1. Is the next packet in the next c-class
2. Is the next packet in the next circuit within the c-class
3. Is one of the packets in the next t-class for that c-class
4. Is the next packet in the next priority group for that t-class

When you enable the interface and one or more circuits for BRS and do not configure any c-classes or t-classes, all the circuits are assigned to one c-class called *default*. With this configuration, there will be only the default c-class on the queue of c-classes and each of the circuits in the c-class with packets for transmission will be handled in a round robin order. If you want BRS to do this, leave all circuits in the default c-class and do not create any other circuit classes.

Orphaned circuits and circuits without BRS explicitly enabled use this default BRS queuing environment in all situations. BRS assigns them to the default c-class.

Using BRS and Priority Queuing

To configure BRS, you should follow this sequence:

1. Enable BRS on the interface.
2. Enable BRS on the circuits and add the c-classes.
3. Assign the circuits to the c-classes.
4. If desired, define t-classes for each of the c-classes.

You can use several bandwidth reservation monitoring commands to display reservation counters for the circuit classes for a given interface:

- clear-circuit-class
- counters-circuit-class
- last-circuit-class

See “Chapter 2. Configuring and Monitoring Bandwidth Reservation” on page 21 for more information on monitoring BRS.

The interface is the one shown at your prompt for the bandwidth monitoring commands. For example, `BRS [i 5]` is the prompt for interface 5.

Queuing Support

With bandwidth reservation over Frame Relay, each circuit can queue frames while in the congested state, even for interfaces and circuits that are not enabled for bandwidth reservation.

Discard Eligibility

The Frame Relay network may discard transmitted data exceeding CIR on a PVC. The DE bit can be set by the router to indicate that some traffic should be considered discard eligible. If appropriate, the Frame Relay network will discard frames marked as discard eligible, which may allow frames that are not marked discard eligible to make it through the network. When assigning a protocol, filter, or tag to a traffic class, you can specify whether or not the protocol, filter, or tag traffic is discard eligible. See “Assign” on page 27 for more information on how to configure traffic as discard eligible. Voice traffic (identified by the protocol VOFR) should always be configured as **not** discard-eligible.

Default Circuit Definitions for Traffic Class Handling

Frame Relay interfaces can have many circuits defined. Rather than having to fully configure traffic class definitions for each circuit, BRS allows you to define a default set of traffic classes and protocol, filter, and tag assignments called default circuit definitions that can be used by any circuit on the interface. When BRS is initially enabled on a circuit, the circuit is initialized to use default circuit definitions. If a circuit cannot use the default circuit definitions for traffic class handling then you can create circuit-specific definitions by using the **add-class**, **change-class**, **assign**, **deassign**, **tag**, and **untag** commands.

If a circuit is using circuit specific definitions and you want it to use the default circuit definitions instead, you can use the **use-circuit-defaults** command at the circuit’s BRS prompt.

The default circuit definitions for traffic class handling are defined by using the **set-circuit-defaults** at the BRS Frame Relay interface prompt. This command gets

you to a BRS circuit defaults prompt where you can add, change, and delete traffic classes, assign and deassign protocols, filters, and tags, and create BRS tags. Changes to the default circuit definitions for traffic classes result in dynamic updates to the traffic class handling for all circuits using the default circuit definitions.

Configuring BRS for Voice over Frame Relay

Voice frames can be transported over dedicated circuits. In this situation, enable BRS on the interface and on the circuits and accept the defaults on circuits associated with voice. You may want to create multiple c-classes and assign the circuits dedicated to voice to a c-class which is associated with a large bandwidth percentage and assign the circuits associated with data to a circuit class associated with a smaller bandwidth percentage.

If voice and other traffic are both transported over the same circuits, enable BRS on the interface and circuits. If you want all circuits serviced in a round robin fashion without favoring one or more circuits you may decide not to create additional c-classes beyond the default c-class. Then, for each circuit over which both voice and data will be transported, it is suggested that you create a t-class with the **create-super-class** command and assign your VOFR traffic to this class. Also create additional t-classes as needed and assign other types of traffic to these t-classes. This configuration will help to ensure that voice traffic gets priority over all other traffic and that unsegmented voice frames can be interleaved between fragmented data segments if fragmentation is enabled. It is recommended that you enable fragmentation on the Frame Relay interface if you will be sending voice and data over the same interface. Fragmentation will result in smaller frames and thus a smaller delay between consecutive voice frames.

Refer to the **enable fragmentation** command in the chapter “Configuring and Monitoring Frame Relay Interfaces” in the *Software User’s Guide* for more information about enabling fragmentation.

Priority Queuing

Bandwidth reservation allocates percentages of total connection bandwidth for specified traffic *classes*, or *t-classes*, defined by the user. Except for a t-class created by the **create-super-class**pubs command which has priority over all other t-classes, BRS t-classes are associated with a bandwidth percentage. Protocols and filter data can be assigned to t-classes and to specific priority queues within a t-class. With priority queuing, a protocol or filter can be assigned to a specific queue within a traffic class with settings: A BRS t-class is a group of packets identified by the same name; for example, a class called “ipx” to designate all IPX packets.

With priority queuing, each bandwidth t-class can be assigned one of the following priority level settings:

- Urgent
- High
- Normal (the default setting)
- Low

for specified traffic classes, or t-classes, defined by the user.

Also, you can set the number of packets that are waiting in the queue for each priority level in each bandwidth t-class. The BRS **queue-length** command sets the

maximum number of output buffers that can be queued in each BRS priority queue, and the maximum number of output buffers that can be queued in each BRS priority queue for when router input buffers are scarce. You can set up priority queue lengths for both PPP and Frame Relay.

Attention: If you set the values for queue length too high, you may seriously degrade the performance of your router.

For BRS, you can set priority queue lengths for PPP and Frame Relay WAN connections. See “Queue-length” on page 39 for a description of the **queue-length** command.

The priority settings in one bandwidth t-class have no effect on other bandwidth classes. No one bandwidth class has priority over the others.

Priority Queuing Without Bandwidth Reservation

When priority queuing is configured without bandwidth reservation, the highest priority traffic is delivered first. In instances of heavy high-priority traffic, lower priority levels can be overlooked. By combining priority queuing with bandwidth reservation, however, packet transmission can be allocated to all types of traffic.

Configuring Traffic Classes

You create a traffic class using the **add-class** command and then assign types of traffic to the class using the **assign** command. Traffic is assigned to a traffic class based on its *protocol type* or based on a filter that further identifies a specific type of *protocol traffic* (for example, SNMP IP packets).

Supported protocol types are:

- IP
- ARP
- DNA
- VINES
- IPX
- OSI
- VOFR
- AP2
- ASRT
- SNA/APPN-ISR
- APPN-HPR
- HPR/IP

BRS Filters

Using bandwidth reservation, you can treat specific protocol traffic differently from other traffic that is using the same protocol type. For example, you can assign SNMP IP traffic to a different traffic class and priority than other IP traffic. In this example, SNMP is a BRS filter because it *filters* (that is, uniquely identifies) specific protocol traffic. IP, ASRT (bridging) and APPN-HPR protocol traffic can be filtered by bandwidth reservation. The following filters are supported:

- IP tunneling

- SDLC tunneling over IP (SDLC Relay)
- BSC tunneling over IP (BSC Relay)
- Rlogin
- Telnet
- SNA/APPN-ISR
- APPN-HPR
- SNMP
- IP Multicast
- DLSw
- MAC Filter
- NetBIOS
- Network-HPR
- High-HPR
- Medium-HPR
- Low-HPR
- XTP
- TCP/UDP port numbers or sockets
- TOS byte
- precedence bit

BRS and Filtering

The following sections describe how to use BRS with various types of filtering.

MAC Address Filtering and Tags

MAC Address filtering is handled by a joint effort between bandwidth reservation and MAC filtering (MCF) using *tags*. For example, a user with bandwidth reservation is able to categorize bridge traffic by assigning a tag to it.

The tagging process is done by creating a filter item in the MAC filtering configuration console and then assigning a tag number to it. This tag number is used to set up a traffic class for all packets associated with this tag. Tag values must currently be in the range 1 through 64. See “Chapter 3. Using MAC Filtering” on page 47 for additional information about MAC filtering.

Note: Tags can be applied ***only*** to bridged packets. On a PPP or Frame Relay connection, up to five tagged MAC filters can be assigned as bandwidth reservation filters and are designated as TAG1 through TAG5. TAG1 is searched for first, then TAG2, and so on up to TAG5. A single MAC filter tag can consist of any number of MAC Addresses set in MCF.

Once you have created a tagged filter in the MAC filtering configuration process, you can use the BRS tag configuration command to assign a BRS tag name (TAG1, TAG2, TAG3, TAG4, or TAG5) to the MAC filter tag number. Then use the BRS tag name on the BRS assign command to assign the corresponding MAC filter to a bandwidth traffic class and priority.

Tags also can refer to “groups,” as in the example of IP Tunnel. IP Tunnel endpoints can belong to any number of groups. Packets are assigned to a particular group

through the tagging feature of MAC Address filtering. For additional information on MAC filtering, refer to “Chapter 3. Using MAC Filtering” on page 47 and “Chapter 4. Configuring and Monitoring MAC Filtering” on page 51.

To apply bandwidth reservation and queuing priority to tagged packets:

1. Use the MAC filtering configuration commands at the `filter config>` prompt to set up tags for packets passing through the bridge. Refer to “Chapter 3. Using MAC Filtering” on page 47 for more information.
2. Use the bandwidth reservation **tag** command to reference a tag for bandwidth reservation.
3. With the bandwidth reservation **assign** command, assign the BRS tag to a t-class. The **assign** command also prompts you for a queuing priority within that BRS t-class.

TCP/UDP Port Number Filtering

You can assign TCP/IP packets from a range of TCP or UDP ports to a BRS t-class and priority based on the packet’s UDP or TCP port number and, optionally, upon a socket. You can specify up to 5 UDP/TCP port number filters, where the filters specify either an individual TCP or UDP port number, a range of TCP or UDP port numbers, or a socket identifier (combination of port number and IP address). You can then assign that filter to a BRS traffic class and priority within the class.

If UDP/TCP port filtering is enabled, BRS looks at each TCP or UDP packet and checks to see if the destination or source port number matches one of the port numbers you have specified for filtering. Also, if you define an IP address as part of the BRS UDP/TCP filter and the destination or source IP address matches the filter address you define, BRS assigns the packet to the traffic class and priority for that port number filter.

For example, you can configure a UDP port number filter for UDP port numbers in the range 25 to 29 and assign the filter to traffic class ’A’ with a priority of ’normal’. BRS queues any UDP packets with a source or destination port number from 25 to 29 on the normal priority queue for traffic class ’A’.

You can also configure a TCP port number filter for TCP port number 50 for IP address 5.5.5.25 and assign the filter to traffic class ’B’ with priority ’urgent’. BRS queues any TCP packets whose source or destination port number is 50 and whose destination or source IP address is 5.5.5.25 on the urgent priority queue for traffic class ’B’.

IPv4 TOS Bit Filtering

You can create filters that will distinguish different types of IP traffic based upon the settings of the Type of Service (TOS) bits. These TOS filters can be used to assign IPv4 traffic that has particular settings of the TOS bits to a different class and priority than other types of IP traffic. Each filter allows IPv4 traffic whose TOS byte value matches the definition of a configured TOS filter to be assigned a unique traffic class and priority. Configuration of a TOS filter includes a mask value specification to define which bits within the TOS byte are to be matched as well as specification of low and high range values for bits that fall within the mask. The filtering mechanism is based solely on IPv4 TOS values; therefore, it does not rely on identification of IPv4 protocol type or port number information as do most of the other IP filters.

This filter is more expansive in its application than BRS IPv4 precedence filtering, which is concerned only with the high-order 3 bits of the TOS byte. When combined with IP access control support to set TOS bits, BRS TOS bit filter support enables you to perform filtering for traffic that is sent over a secure tunnel, that is fragmented, or that cannot be identified using the BRS UDP and TCP port number filter support. Also, IP access control support allows you to set the TOS bits to a user-defined value instead of having to use the hard-coded precedence bit values for APPN and DLSw that are associated with BRS IPv4 precedence bit filtering. Therefore, it is recommended that you use IP access control and BRS TOS filter support instead of BRS IPv4 precedence bit filtering.

As indicated in “Order of Filtering Precedence” on page 11, TOS filter matches are checked prior to IPv4 precedence bit filters and other IP-specific filters. Checks for the TOS1 to TOS5 filter matches are done sequentially, beginning with the TOS1 filter. Up to 5 TOS filters can be defined.

Important: Keep in mind that a packet with a particular TOS value is handled according the first TOS filter definition that the value matches. Be careful to set up your filters so that a particular TOS byte is filtered by the intended filter, not accidentally filtered by a lower-numbered filter. Refer to “Using IP” in *Using and Configuring Features* for more information.

Using IP Version 4 Precedence Bit Processing for SNA Traffic in IP Secure Tunnels and Secondary Fragments

BRS normally differentiates IP TCP and UDP traffic according to its port numbers. However, BRS cannot identify the ports after traffic has been encapsulated twice, such as IP traffic transported through an IP secure tunnel or in a secondary UDP or TCP fragment. IP version 4 precedence bit processing has been added to BRS to enable BRS to filter IP secure tunnel packets or TCP and UDP secondary fragment packets.

Note: It is recommended that you use BRS IPv4 TOS bit filtering instead of IPv4 precedence bit processing. See “IPv4 TOS Bit Filtering” on page 8 for more details.

When APPN/HPR traffic is being routed over IP, each transmission priority of APPN-HPR (network, high, medium, and low) is mapped to a particular value of the three IP version 4 precedence bits.

- The HPR network transmission priority maps to the IPv4 precedence value of '110'b.
- The HPR high transmission priority maps to the IPv4 precedence value of '100'b.
- The HPR medium transmission priority maps to the IPv4 precedence value of '010'b.
- The HPR low transmission priority maps to the IPv4 precedence value of '001'b.

When IPv4 precedence filtering is enabled for BRS and the precedence bits in an IP packet match one of the values used for APPN/HPR traffic, then the packet is queued on the priority queue of the BRS t-class to which the corresponding HPR transmission priority is assigned. For example, if an IP packet has a precedence value of '110'b and the BRS HPR-Network filter is assigned to t-class A and priority level normal, then the packet is queued on the normal priority queue of t-class A. If

a BRS HPR transmission priority filter is not configured, but the APPN-HPR filter is configured, then the packet is queued on the priority queue and t-class to which the APPN-HPR filter is assigned.

These three kinds of traffic map to the IPv4 precedence value '011'b:

- APPN/HPR XID traffic that is sent when APPN/HPR is routed over IP
- DLSw traffic
- TN3270 traffic

Because several types of traffic map to one value, BRS cannot distinguish between them when it is enabled to filter based on the IPv4 precedence bits. Therefore, when BRS encounters an IP packet with a precedence value of '011'b, it evaluates the BRS filters in the following order to determine whether or not the filter is enabled. When it finds a BRS filter that is configured, the packet is queued on the priority queue and t-class to which the BRS filter is assigned:

- SNA/APPN-ISR (used for APPN/HPR XID exchanges)
- DLSw
- Telnet

If a packet has one of the precedence values that are filtered by BRS, but none of the applicable BRS filter types are configured, the packet is queued on the priority queue and the BRS t-class to which the IP protocol is assigned.

When TN3270 traffic is sent by a client to the 2210 over a wide-area network where BRS is enabled, traffic from the client cannot be prioritized by BRS unless the client sets the precedence bits to '011'b.

You must configure IPv4 precedence bit handling in multiple places:

1. In BRS you configure whether or not BRS should filter based on the IPv4 precedence bits. It only performs this type of filtering for IP secure tunnel packets or TCP and UDP secondary fragment packets.
2. When you configure DLSw, HPR over IP, and TN3270, you specify whether or not the 2210 should set the IPv4 precedence bits for packets that it originates for each of these protocol types.

Perform these three steps to use IPv4 precedence bit filtering:

1. Activate IPv4 precedence filtering in BRS.
2. Configure BRS t-classes and assign protocols and filters for various categories of SNA traffic, as you would for SNA traffic that is not transported in an IP secure tunnel or is not fragmented.
3. Enable the setting of the IPv4 precedence bits when configuring the DLSw, HPR over IP, and TN3270 protocols.
4. Configure IPsec to create a secure tunnel over which the DLSw, HPR over IP, and TN3270 traffic will flow.

SNA and APPN Filtering for Bridged Traffic

The SNA/APPN-ISR filter allows you to assign SNA and APPN-ISR traffic that is being bridged to a BRS traffic class. SNA and APPN-ISR traffic is identified as any bridged packets with a destination or source SAP of 0x04, 0x08, or 0x0C and whose LLC (802.2) control field indicates that it is not an unnumbered information (UI) frame.

Note: Frame Relay BAN packets are in this category.

The APPN-HPR filters allow you to assign HPR traffic that is being bridged to a BRS t-class. HPR traffic is identified as any bridge packet with a destination or source SAP of X'04', X'08', X'0C', or X'C8' and whose LLC (802.2) control field indicates it is an unnumbered information (UI) frame.

The Network-HPR, High-HPR, Medium-HPR, and Low-HPR filters allow HPR bridge traffic to further be filtered according to the HPR transmission priority. For example, if you want to assign HPR traffic that uses the network transmission priority to one t-class and priority and all other HPR bridged traffic to a different t-class or priority, you would assign the Network-HPR filter to the appropriate t-class and priority and use the APPN-HPR filter to assign the rest of the HPR traffic to a different t-class or priority.

APPN-HPR traffic that is being routed over IP is filtered using the UDP port number assigned for network, high, medium and low HPR transmission priorities. An additional UDP port number is used for XID exchanges. All of the UDP port numbers used to support APPN-HPR over IP are configurable.

If APPN is not enabled in an intermediate router in the IP network, you can configure UDP port numbers for HPR over IP from the `BRS Config>` command prompt. If APPN is enabled in the device, BRS will use the values configured at the `APPN Config>` command prompt.

Other filters may help you to assign traffic. For example, the DLSw filter allows you to assign SNA-DLSw traffic that is being sent over a TCP connection to a BRS t-class.

For SNA/APPN-ISR and APPN-HPR filters, if you want to check for SAPs other than the ones above, create a sliding window filter using MAC filtering and tag that filter. Then assign the tagged MAC filter to a BRS t-class.

Order of Filtering Precedence

It is possible for a packet to match more than one BRS filter type. For example, an IP tunneled bridge packet containing SNA data could match the IP tunneling filter and the SNA/APPN-ISR filter. The order in which the filters are evaluated to determine whether or not a packet matches a BRS filter type is as follows:

1. TOS filters (IP)
2. IPv4 precedence handling
3. MAC filter tag match for bridging packets (IP/ASRT)
4. NetBIOS for bridging (IP/ASRT)
5. SNA/APPN-ISR for bridging (IP/ASRT)
6. HPR-Network (IP/ASRT/APPN-HPR)
7. HPR-High (IP/ASRT/APPN-HPR)
8. HPR-Medium (IP/ASRT/APPN-HPR)
9. HPR-Low (IP/ASRT/APPN-HPR)
10. APPN-HPR (IP/ASRT)
11. UDP/TCP port number filters (IP)
12. IP tunneling (IP)
13. SDLC/BSC relay (IP)
14. DLSw (IP)
15. Multicast (IP)

16. SNMP (IP)
17. Rlogin (IP)
18. Telnet (IP)
19. XTP (IP)

Note: The protocols for which a filter applies are shown in parentheses.

Sample Configurations

Using Default Circuit Definitions for Traffic Class Handling of Frame Relay Circuits

Notes:

1 Configure feature BRS.

2 Enable BRS on interface 1.

3 Enable BRS on circuits 16, 17, 18. Default circuit definitions for traffic class handling are used for these circuits.

4 Access the set-circuit-defaults menu to define default circuit definitions for traffic class handling.

5 Add traffic classes and assign protocols and filters to the traffic classes.

6 List and show the BRS definitions for circuit 16. Since circuit 16 is using default circuit definitions, the traffic classes and protocol and filter assignments defined by the default circuit definitions are displayed.

7 Change circuit 17 from using default circuit definitions to use circuit-specific definitions for traffic class handling by creating a unique class, CIRC171. This class can have protocols, filters, or tags assigned to it.

8 Change the default circuit definitions such that the DEF1 and DEF2 traffic classes each reserve 10% of the bandwidth and then show that these changes are picked up by circuit 16 but not by circuit 17, since circuit 17 is now using circuit-specific definitions.

9 Alter circuit 17 to use default circuit definitions for traffic class handling instead of circuit-specific definitions.

```
t 6
 Gateway user configuration
 Config>feature brs      1
 Bandwidth Reservation User Configuration
 BRS Config>interface 1     2
 BRS [i 1]Config>enable
 Please restart router for this command to take effect.
 BRS [i 1] Config>circuit 16    3
 BRS [i 1][dlci 16] Config>enable
 Defaults are in effect for this circuit.
 Please restart router for this command to take effect.
 BRS [i 1][dlci 16] Config>exit
 BRS [i 1]Config>circuit 17
 BRS [i 1][dlci 17] Config>enable
 Defaults are in effect for this circuit.
 Please restart router for this command to take effect.
 BRS [i 1][dlci 17] Config>exit
 BRS [i 1]Config>circuit 18
 BRS [i 1][dlci 18] Config>enable
 Defaults are in effect for this circuit.
 Please restart router for this command to take effect.
 BRS [i 1][dlci 18] Config>
 *restart
 Are you sure you want to restart the gateway? (Yes or [No]): yes
```

```
*t 6
Gateway user configuration
Config>feature brs
Bandwidth Reservation User Configuration
BRS Config>interface 1
BRS[i 1] Config>list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
interface number 1
maximum queue length 10, minimum queue length 3
total bandwidth allocated 10%
total circuit classes defined (counting one default) 1


class DEFAULT has 10% bandwidth allocated
  the following circuits are assigned:
      16  using defaults.
      17  using defaults.
      18  using defaults.

default class is DEFAULT


BRS [i 1] Config>?
 ENABLE
 DISABLE
 SET-CIRCUIT-DEFAULTS
 CIRCUIT
 ADD-CIRCUIT-CLASS
 DEL-CIRCUIT-CLASS
 CHANGE-CIRCUIT-CLASS
 DEFAULT-CIRCUIT-CLASS
 ASSIGN-CIRCUIT
 DEASSIGN-CIRCUIT
 QUEUE-LENGTH
 LIST
 SHOW
 CLEAR-BLOCK
 EXIT
 BRS [i 1] Config>set-circuit-defaults     4
 BRS [i 1] [circuit defaults] Config>?
 ADD-CLASS
 DEL-CLASS
 CHANGE-CLASS
 DEFAULT-CLASS
 TAG
 UNTAG
 ASSIGN
 DEASSIGN
 LIST
 EXIT
 BRS [i 1] [circuit defaults] Config>add    5
 Class name [DEFAULT]?DEF1
 Percent bandwidth to reserve [10]? 5
 BRS [i 1] [circuit defaults] Config>add
 Class name [DEFAULT]?DEF2
 Percent bandwidth to reserve [10]?5
 BRS [i 1] [circuit defaults] Config>assign ip
 Class name [DEFAULT]?DEF1
 Priority <URGENT/HIGH/NORMAL/LOW> [NORMAL]?
 Frame Relay Discard Eligible <NO/YES> [NO]?
 BRS [i 1] [circuit defaults] Config>assign asrt
 Class name [DEFAULT]? DEF2
 Priority <URGENT/HIGH/NORMAL/LOW> [NORMAL]?
 Frame Relay Discard Eligible <NO/YES> [NO]?
 BRS[i 1] [circuit defaults] Config>list

 BANDWIDTH RESERVATION listing from SRAM
 bandwidth reservation is enabled
 interface number 1, default circuit
 total bandwidth allocated 60%
 total classes defined (counting one local and one default) 4
```

```
class LOCAL has 10% bandwidth allocated
  protocols and filters cannot be assigned to this class.


class DEFAULT has 40% bandwidth allocated
  the following protocols and filters are assigned:
    protocol ARP with default priority is not discard eligible
    protocol DNA with default priority is not discard eligible
    protocol VINES with default priority is not discard eligible
    protocol IPX with default priority is not discard eligible
    protocol OSI with default priority is not discard eligible
    protocol VOFR with default priority is not discard eligible
    protocol AP2 with default priority is not discard eligible

class DEF1 has 5% bandwidth allocated
  the following protocols and filters are assigned:
    protocol IP with priority NORMAL is not discard eligible


class DEF2 has 5% bandwidth allocated
  the following protocols and filters are assigned:
    protocol ASRT with priority NORMAL is not discard eligible

assigned tags:

default class is DEFAULT with priority NORMAL

BRS  [i 1] [circuit defaults] Config>exit
BRS  [i 1] Config>circuit 16     6
BRS  [i 1][dlci 161] Config>list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
interface number 1, circuit number 16 using defaults.
maximum queue length 10, minimum queue length 3
total bandwidth allocated 60%
total classes defined (counting one local and one default) 4


class LOCAL has 10% bandwidth allocated
  protocols and filters cannot be assigned to this class.


class DEFAULT has 40% bandwidth allocated
  the following protocols and filters are assigned:
    protocol ARP with default priority is not discard eligible
    protocol DNA with default priority is not discard eligible
    protocol VINES with default priority is not discard eligible
    protocol IPX with default priority is not discard eligible
    protocol OSI with default priority is not discard eligible
    protocol VOFR with default priority is not discard eligible
    protocol AP2 with default priority is not discard eligible

class DEF1 has 5% bandwidth allocated
  the following protocols and filters are assigned:
    protocol IP with priority NORMAL is not discard eligible


class DEF2 has 5% bandwidth allocated
  the following protocols and filters are assigned:
    protocol ASRT with priority NORMAL is not discard eligible

assigned tags:

default class is DEFAULT with priority NORMAL


BRS  [i 1] [dlci 16] Config>show

BANDWIDTH RESERVATION currently in RAM
interface number 1, circuit number 16 using defaults.
maximum queue length 10, minimum queue length 3
4 current defined classes:
  class LOCAL has 10% bandwidth allocated
  class DEFAULT has 40% bandwidth allocated
  class DEF1 has 5% bandwidth allocated
  class DEF2 has 5% bandwidth allocated

protocol and filter assignments:
```

```
 Protocol/Filter      Class              Priority    Discard Eligible
 ---------------      -----              --------    ----------------
 IP                   DEF1                 NORMAL            NO
 ARP                  DEFAULT              NORMAL            NO
 DNA                  DEFAULT              NORMAL            NO
 VINES                DEFAULT              NORMAL            NO
 IPX                  DEFAULT              NORMAL            NO
 OSI                  DEFAULT              NORMAL            NO
 VOFR                 DEFAULT              NORMAL            NO
 AP2                  DEFAULT              NORMAL            NO
 ASRT                 DEF2                 NORMAL            NO

 BRS [i 1] [dlci 16] Config>exit


BRS [i 1] Config>circuit 17
 BRS [i 1] [dlci 17] Config>list

 BANDWIDTH RESERVATION listing from SRAM
 bandwidth reservation is enabled
 interface number 1, circuit number 17 using defaults.
 maximum queue length 10, minimum queue length 3
 total bandwidth allocated 60%
 total classes defined (counting one local and one default) 4


 class LOCAL has 10% bandwidth allocated
   protocols and filters cannot be assigned to this class.


 class DEFAULT has 40% bandwidth allocated
   the following protocols and filters are assigned:
     protocol ARP with default priority is not discard eligible
     protocol DNA with default priority is not discard eligible
     protocol VINES with default priority is not discard eligible
     protocol IPX with default priority is not discard eligible
     protocol OSI with default priority is not discard eligible
     protocol VOFR with default priority is not discard eligible
     protocol AP2 with default priority is not discard eligible

 class DEF1 has 5% bandwidth allocated
   the following protocols and filters are assigned:
     protocol IP with priority NORMAL is not discard eligible


 class DEF2 has 5% bandwidth allocated
   the following protocols and filters are assigned:
     protocol ASRT with priority NORMAL is not discard eligible

 assigned tags:

 default class is DEFAULT with priority NORMAL

 BRS [i 1] [dlci 17] Config>add-class   7
 This circuit is currently using circuit defaults...
 Are you sure you want to override the defaults ?(Yes or [No]): yes
 Class name [DEFAULT]? CIRC171
 Percent bandwidth to reserve [10]? 5
 BRS[i 1] [dlci 17] Config>assign vines
 Class name [DEFAULT]? CIRC171
 Priority <URGENT/HIGH/NORMAL/LOW> [NORMAL]?
 Frame Relay Discard Eligible <NO/YES>[NO]?


BRS [i 1] [dlci 17] Config>list

 BANDWIDTH RESERVATION listing from SRAM
 bandwidth reservation is enabled
 interface number 1, circuit number 17
 maximum queue length 10, minimum queue length 3
 total bandwidth allocated 65%
 total classes defined (counting one local and one default) 5


 class LOCAL has 10% bandwidth allocated
   protocols and filters cannot be assigned to this class.


 class DEFAULT has 40% bandwidth allocated
   the following protocols and filters are assigned:
```

```
    protocol ARP with default priority is not discard eligible
    protocol DNA with default priority is not discard eligible
    protocol IPX with default priority is not discard eligible
    protocol OSI with default priority is not discard eligible
    protocol VOFR with default priority is not discard eligible
    protocol AP2 with default priority is not discard eligible


class DEF1 has 5% bandwidth allocated
  the following protocols and filters assigned:
    protocol IP with priority NORMAL is not discard eligible


class DEF2 has 5% bandwidth allocated
  the following protocols and filters are assigned:
    protocol ASRT with priority NORMAL is not discard eligible


class CIRC171 has 5% bandwidth allocated
  the following protocols and filters are assigned:
    protocol VINES with priority NORMAL is not discard eligible

assigned tags:

default class is DEFAULT with priority NORMAL

BRS [i 1] [dlci 17] Config>show

BANDWIDTH RESERVATION currently in RAM
interface number 1, circuit number 17
maximum queue length 10, minimum queue length 3
5 current defined classes:
  class LOCAL has 10% bandwidth allocated
  class DEFAULT has 40% bandwidth allocated
  class DEF1 has 5% bandwidth allocated
  class DEF2 has 5% bandwidth allocated
  class CIRC171 has 5% bandwidth allocated


protocol and filter assignments:

Protocol/Filter      Class              Priority    Discard Eligible
---------------      -----              --------    ----------------
IP                   DEF1                 NORMAL             NO
ARP                  DEFAULT              NORMAL             NO
DNA                  DEFAULT              NORMAL             NO
VINES                CIRC171              NORMAL             NO
IPX                  DEFAULT              NORMAL             NO
OSI                  DEFAULT              NORMAL             NO
VOFR                 DEFAULT              NORMAL             NO
AP2                  DEFAULT              NORMAL             NO
ASRT                 DEF2                 NORMAL             NO

BRS [i 1] [dlci 17] Config>exit
BRS [i 1] Config>set-circuit-defaults
BRS [i 1] [circuit defaults] Config>change DEF1     8
Percent bandwidth to reserve [ 5]? 10
BRS [i 1] [circuit defaults] Config>change DEF2
Percent bandwidth to reserve [5]? 10
BRS [i 1] [circuit defaults] Config>list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
interface number 1, default circuit
total bandwidth allocated 70%
total classes defined (counting one local and one default) 4


class LOCAL has 10% bandwidth allocated
  protocols and filters cannot be assigned to this class.


class DEFAULT has 40% bandwidth allocated
  the following protocols and filters are assigned:
    protocol ARP with default priority is not discard eligible
    protocol DNA with default priority is not discard eligible
    protocol VINES with default priority is not discard eligible
    protocol IPX with default priority is not discard eligible
    protocol OSI with default priority is not discard eligible
```

```
      protocol VOFR with default priority is not discard eligible
      protocol AP2 with default priority is not discard eligible

  class DEF1 has 10% bandwidth allocated
    the following protocols and filters are assigned:
      protocol IP with priority NORMAL is not discard eligible


  class DEF2 has 10% bandwidth allocated
    the following protocols and filters are assigned:
      protocol ASRT with priority NORMAL is not discard eligible

  assigned tags:

  default class is DEFAULT with priority NORMAL

  BRS [i 1] [circuit defaults] Config>exit

BRS [i 1] Config>circuit 16
  BRS [i 1] [dlci 16] Config>list

  BANDWIDTH RESERVATION listing from SRAM
  bandwidth reservation is enabled
  interface number 1, circuit number 16 using defaults.
  maximum queue length 10, minimum queue length 3
  total bandwidth allocated 70%
  total classes defined (counting one local and one default) 4


  class LOCAL has 10% bandwidth allocated
    protocols and filters cannot be assigned to this class.


  class DEFAULT has 40% bandwidth allocated
    the following protocols and filters are assigned:
      protocol ARP with default priority is not discard eligible
      protocol DNA with default priority is not discard eligible
      protocol VINES with default priority is not discard eligible
      protocol IPX with default priority is not discard eligible
      protocol OSI with default priority is not discard eligible
      protocol VOFR with default priority is not discard eligible
      protocol AP2 with default priority is not discard eligible

  class DEF1 has 10% bandwidth allocated
    the following protocols and filters are assigned:
      protocol IP with priority NORMAL is not discard eligible


  class DEF2 has 10% bandwidth allocated
    the following protocols and filters are assigned:
      protocol ASRT with priority NORMAL is not discard eligible

  assigned tags:

  default class is DEFAULT with priority NORMAL

  BRS [i 1] [dlci 16] Config>exit

BRS [i 1] Config>circuit 17
  BRS [i 1[ [dlci 17] Config>list

  BANDWIDTH RESERVATION listing from SRAM
  bandwidth reservation is enabled
  interface number 1, circuit number 17
  maximum queue length 10, minimum queue length 3
  total bandwidth allocated 65%
  total classes defined (counting one local and one default) 5


  class LOCAL has 10% bandwidth allocated
    protocols and filters cannot be assigned to this class.


  class DEFAULT has 40% bandwidth allocated
    the following protocols and filters are assigned:
      protocol ARP with default priority is not discard eligible
      protocol DNA with default priority is not discard eligible
      protocol IPX with default priority is not discard eligible
      protocol OSI with default priority is not discard eligible
```

```
    protocol VOFR with default priority is not discard eligible
    protocol AP2 with default priority is not discard eligible


class DEF1 has 5% bandwidth allocated
  the following protocols and filters are assigned:
    protocol IP with priority NORMAL is not discard eligible


class DEF2 has 5% bandwidth allocated
  the following protocols and filters are assigned:
    protocol ASRT with priority NORMAL is not discard eligible


class CIRC171 has 5% bandwidth allocated
  the following protocols and filters are assigned:
    protocol VINES with priority NORMAL is not discard eligible

assigned tags:

default class is DEFAULT with priority NORMAL

BRS [i 1] [dlci 17] Config>use-circuit-defaults     9
This circuit is currently NOT using circuit defaults...
Are you sure you want to delete current definitions and use defaults ? (Yes or
[No]): yes
Defaults are in effect for this circuit.
Please restart router for this command to take effect.
BRS [i 1] [dlci 17] Config>
*restart
Are you sure you want to restart the gateway? (Yes or [No] ):yes


*t 6
Gateway user configuration
Config>feature brs
Bandwidth Reservation User Configuration
BRS Config>interface 1
BRS [i 1] Config>circuit 17
BRS [i 1] [dlci 17] Config>list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
interface number 1, circuit number 17 using defaults.
maximum queue length 10, minimum queue length 3
total bandwidth allocated 70%
total classes defined (counting one local and one default) 4


class LOCAL has 10% bandwidth allocated
  protocols and filters cannot be assigned to this class.


 class DEFAULT has 40% bandwidth allocated
  the following protocols and filters are assigned:
    protocol ARP with default priority is not discard eligible
    protocol DNA with default priority is not discard eligible
    protocol VINES with default priority is not discard eligible
    protocol IPX with default priority is not discard eligible
    protocol OSI with default priority is not discard eligible
    protocol VOFR with default priority is not discard eligible
    protocol AP2 with default priority is not discard eligible

class DEF1 has 10% bandwidth allocated
  the following protocols and filters are assigned:
    protocol IP with priority NORMAL is not discard eligible


class DEF2 has 10% bandwidth allocated
  the following protocols and filters are assigned:
    protocol ASRT with priority NORMAL is not discard eligible

assigned tags:

default class is DEFAULT with priority NORMAL


BRS [i 1] [dlci 17] Config>show

BANDWIDTH RESERVATION currently in RAM
```

```
interface number 1, circuit number 17 using defaults.
maximum queue length 10, minimum queue length 3
4 current defined classes:
  class LOCAL has 10% bandwidth allocated
  class DEFAULT has 40% bandwidth allocated
  class DEF1 has 10% bandwidth allocated
  class DEF2 has 10% bandwidth allocated

protocol and filter assignments:

Protocol/Filter      Class              Priority    Discard Eligible
---------------      -----              --------    ----------------
IP                   DEF1                 NORMAL             NO
ARP                  DEFAULT              NORMAL             NO
DNA                  DEFAULT              NORMAL             NO
VINES                DEFAULT              NORMAL             NO
IPX                  DEFAULT              NORMAL             NO
OSI                  DEFAULT              NORMAL             NO
VOFR                 DEFAULT              NORMAL             NO
AP2                  DEFAULT              NORMAL             NO
ASRT                 DEF2                 NORMAL             NO

BRS [i 1] [dlci 17] Config>exit
```

Chapter 2. Configuring and Monitoring Bandwidth Reservation

This chapter describes the Bandwidth Reservation System (BRS) configuration and operational commands.

This chapter includes the following sections:

Bandwidth Reservation Configuration Overview

To access bandwidth reservation configuration commands and configure bandwidth reservation on your router:

1. At the `OPCON (*)` prompt, enter **talk 6**.
2. At the `Config>` prompt, enter **feature brs**.
3. At the `BRS Config>` prompt, enter **interface #**.
4. At the `BRS [i 0] Config>` prompt, enter **enable**.

   This is the interface prompt level, and the interface number is zero in this instance. You need to repeat step 3 and step 4 for each interface you are configuring.

   If you are configuring BRS on a Frame Relay interface, continue with step 4a:

   If you are configuring BRS on any other interface, go directly to step 5.

   a. At the `BRS [i 0] Config>` prompt, enter **circuit** *#*, where *#* is the number of the circuit you want to configure.

   b. At the `BRS [i 0] [dlci 16] Config>` prompt, enter **enable**. This is the circuit prompt level and the circuit (DLCI) number is 16 in this instance.

   c. At the `BRS [i 0] [dlci 16] Config>` prompt, enter **exit** to return to the interface level prompt.

   d. Repeat steps 4a through 4c for each circuit for which you want to define BRS t-classes.
5. Restart your router.
6. Repeat steps 1 through 3 to configure bandwidth reservation for the particular interface that you have enabled.
7. If you are configuring BRS on a PPP interface, at the `BRS[i 0]Config>` prompt, configure traffic classes and assign protocols, filters, and tags to the traffic classes using the configuration commands listed in Table 3 on page 24. If you are configuring BRS on a FR interface, follow steps 8 through 10.
8. If you are configuring BRS on a FR interface, you can configure circuit classes and assign circuits to circuit classes using the commands listed in Table 2 on page 23
9. If you want to use default circuit definitions then enter the **set-circuit-defaults** command at the `BRS[i 0]Config>` prompt. This gets you to the `BRS[i 0][circuit defaults]` prompt where you can use the appropriate commands from Table 3 on page 24 to configure traffic classes and assign protocols, filters,

and tags to the traffic classes. Once you are through defining default circuit definitions for traffic class handling, enter ″exit″ to return to the `BRS[i 0] Config>` prompt.

10. If you have FR circuits that cannot use default circuit definitions for traffic class handling, enter **circuit** *permanent-virtual-circuit circuit_number*. This will access the circuit prompt where you can use the commands listed in Table 3 on page 24 to create circuit-specific definitions for traffic class handling.

Note: You do not need to restart the router for t-class and c-class configuration changes to take effect.

The **talk 6 (t 6)** command lets you access the configuration process.

The **feature brs** command lets you access the BRS configuration process. You can enter this command by using either the feature name (brs) or number (1).

The **interface #** command selects the particular interface that you want to configure for bandwidth reservation. Before configuring any BRS classes, you must use the **enable** command to enable BRS on the interface. In Step 4 on page 21, the prompt indicates that the selected interface's number is zero.

The **circuit #** command selects the circuit on the FR interface on which you want to configure BRS traffic classes. Before configuring any BRS t-classes for the circuit, you must use the **enable** command to enable BRS on the circuit. In step 4.b on page 21, the prompt indicates that circuit 16 on interface 0 has been selected.

You must enable bandwidth reservation for the selected interface and circuit and then restart your router before configuring circuit classes (Frame Relay only), and traffic classes.

To return to the `Config>` prompt at any time, enter the **exit** command at the different levels of BRS prompts until you are at the `Config>` prompt.

Bandwidth Reservation Configuration Commands

This section describes the Bandwidth Reservation configuration commands. The commands that can be used differ depending on the BRS configuration prompt that is displayed (`BRS Config>`, `BRS [i x] Config>`, or `BRS [i x] [dlci y] Config>`, or `BRS [i x] [circuit defaults] Config>`).

Table 1. Bandwidth Reservation Configuration Command Summary (Available from BRS Config> prompt)

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page xxix.
Activate-IP-precedence-filtering	Activates BRS IPv4 precedence filtering of APPN and SNA packets that are sent over a secure IP tunnel or that are in secondary TCP or UDP fragments. You also must configure the setting of the IPv4 precedence bits when you configure DLSw, HPR over IP or TN3270.

Table 1. Bandwidth Reservation Configuration Command Summary (Available from BRS Config> prompt) (continued)

Command	Function
Deactivate-IP-precedence-filtering	Deactivates IPv4 precedence filtering processing.
Enable-hpr-over-ip-port-numbers	Enables the use of BRS filtering for APPN-HPR over IP traffic and allows the configuration of the UDP port numbers used to identify HPR over IP packets. **Note:** If APPN is in the load image, this command is not supported since BRS learns from APPN if HPR over IP has been configured and, if it has been configured, learns the UDP port numbers that will be used for HPR over IP packets from the APPN support.
Disable-hpr-over-ip-port-numbers	Disables BRS filtering of APPN-HPR over IP traffic. **Note:** If APPN is in the load image, this command is not supported since BRS learns from APPN whether or not HPR over IP has been configured.
Interface	Selects an interface on which to configure bandwidth reservation. **Note:** This command must be entered before using any other configuration commands. See Table 2 and Table 3 on page 24.
List	Lists the interfaces that can support bandwidth reservation and, for each interface, indicates if bandwidth reservation is enabled or disabled.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Table 2. BRS Interface Configuration Commands Available from BRS [i #] Config> prompt for Frame Relay Interfaces

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Add-circuit-class	Sets the name of a bandwidth c-class and its percentage of bandwidth.
Assign-circuit	Assigns a specified circuit to the specified bandwidth c-class.
Change-circuit-class	Changes the amount of bandwidth configured for a bandwidth c-class.
Circuit	Accesses the BRS circuit-level prompt (`BRS [i x][dlci y] Config>`) prompt where you can use the commands listed in Table 3 on page 24 to configure Bandwidth Reservation on the Frame Relay circuit.
Clear-block	Clears the configuration data associated with the current interface from SRAM. Circuit class configuration data and default circuit definitions for traffic class handling are cleared.
Deassign-circuit	Restores the specified circuit to the default c-class
Default-circuit-class	Assigns the name of a default bandwidth c-class and its percentage of the interface’s bandwidth.
Del-circuit-class	Deletes the specified bandwidth c-class.

Table 2. BRS Interface Configuration Commands Available from BRS [i #] Config> prompt for Frame Relay Interfaces (continued)

Command	Function
Disable	Disables bandwidth reservation on the interface .
Enable	Enables bandwidth reservation on the interface.
List	Displays the c-classes and assigned circuit definitions from SRAM.
Queue-length	Sets the maximum and minimum values for the number of packets in a priority queue.
Set-circuit-defaults	Accesses the `BRS [i x] [circuit defaults] Config>` command prompt so that you can use the appropriate commands from Table 3 to create default circuit definitions for traffic class handling.
Show	Displays the currently defined c-classes and assigned circuits from SRAM.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

The following table lists BRS circuit commands Available from `BRS [i x] Config>` for PPP interfaces, `BRS [i x] dlci [y] Config>` prompt for Frame Relay circuits, and from the `BRS [i x] [circuit defaults] Config>` prompt.

Table 3. BRS Traffic Class Handling Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Add-class	Allocates a designated amount of bandwidth to a user-defined traffic class.
Create-super-class	Defines the t-class called *super-class*.
Assign	Assigns a protocol or filter to a configured traffic class.
Change-class	Changes the amount of bandwidth configured for a bandwidth t-class.
Clear-block	Clears the traffic class and protocol, filter, and tag assignment configuration data from SRAM for the PPP interface or Frame Relay circuit. **Note:** This command cannot be used from the `BRS [i x] [circuit defaults] Config>` prompt.
Deassign	Restores the queuing of the specified packet or filter to the default t-class and priority.
Default-class	Sets the default t-class and priority to a desired value and assigns all unassigned protocols to the new default t-class.
Del-class	Deletes a previously configured bandwidth t-class.
Disable	Disables bandwidth reservation on the PPP interface or Frame Relay circuit. **Note:** BRS cannot be enabled or disabled from the `BRS [i x] [circuit defaults] Config>` prompt.
Enable	Enables bandwidth reservation on the PPP interface or Frame Relay circuit. **Note:** BRS cannot be enabled or disabled from the `BRS [i x] [circuit defaults] Config>` prompt.
List	Lists the configured t-classes and protocol, filter and tag assignments stored in SRAM.
Queue-length	Sets the maximum and minimum values for the number of packets in a priority queue. **Note:** This command is not supported at the `BRS [i x] [circuit defaults] Config>` prompt.

Table 3. BRS Traffic Class Handling Commands (continued)

Command	Function
Show	Displays the currently defined t-classes and protocol, filter, and tag assignments stored in RAM. **Note:** This command is not supported at the `BRS [i x] [circuit defaults] Config>` prompt.
Tag	Assigns a BRS tag name (TAG1 - TAG5) to a MAC filter that has been tagged during the configuration of the MAC Filtering feature.
Untag	Removes the relationship between a BRS tag name (TAG1 - TAG5) and a MAC filter that has been tagged during configuration of the MAC filtering feature.
Use-circuit-defaults	Allows the user to delete the circuit-specific definitions and use the circuit-defaults definitions for the traffic-class handling. This command is valid at the `BRS [i x] dlci [y] Config>` prompt for Frame Relay only. **Note:** The router must be restarted in order for the defaults to become operational.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Use the appropriate commands to configure bandwidth reservation for the Point-to-Point protocol (PPP) and Frame Relay. For Frame Relay, you need to configure the circuit and the network interface. For PPP, you only need to configure the network interface.

Notes:

1. When the **clear-block, disable**, **enable**, **list**, and **show** commands are issued from within the BRS interface menu, they affect or list the bandwidth reservation information configured for the selected interface. When these commands are issued from within the BRS circuit menu, only the Frame Relay bandwidth reservation information configured for the permanent virtual circuit (PVC) is affected or listed.
2. Before using the bandwidth reservation commands, keep the following in mind:
   - You must use the **interface** command to select an interface before you use any other configuration commands. (BRS configuration enforces this.)
   - The *Class-name* parameter is case-sensitive.
   - To view the current *class-names*, use the **list** or **show** command.
   - After you enable bandwidth reservation on an interface or circuit, you can add/delete/change circuit and traffic classes and assign circuits or protocols dynamically. The only commands that require a router restart before taking effect are the enable, disable, use-circuit-defaults, and clear-block commands.
3. You do not need to restart the router for t-class and c-class configuration changes to take effect.

Activate-IP-precedence-filtering

Use the **activate-ip-precedence-filtering** command to activate BRS IPv4 precedence filtering of APPN and SNA packets that are sent over a secure IP tunnel or that are in secondary TCP or UDP fragments. You also must configure the setting of the IPv4 precedence bits when you configure DLSw, HPR over IP or TN3270. See “Using IP Version 4 Precedence Bit Processing for SNA Traffic in IP Secure Tunnels and Secondary Fragments” on page 9 for more information.

Syntax:

activate-ip-precedence-filtering

Add-circuit-class

Note: Used only when configuring Frame Relay.

Use the **add-circuit-class** command at the interface level to allocate a designated amount of bandwidth to be used by the group of circuits assigned to the user-defined bandwidth c-class.

Syntax:

add-circuit-class *class-name %*

Add-class

Use the **add-class** command to allocate a designated amount of bandwidth to a user-defined bandwidth t-class.

Note: If this command is used for a Frame Relay circuit that is currently using default circuit definitions for traffic class handling, you will be asked whether or not you want to override the default circuit definitions. If you answer “Yes”, the circuit will be changed to use circuit-specific definitions for traffic class handling and the command will be allowed. If you answer “No”, the command is aborted and default circuit definitions will continue to be used for the circuit. If you want to change the default circuit definitions, you should go to the `BRS [i x][circuit defaults]Config>` command prompt.

Syntax:

add-class [*class-name* or *class#*] *%*

Example 1: Adding one class named CIRC17 on a Frame Relay circuit

```
BRS [i 1] [dlci 17] Config>add-class
This circuit is currently using circuit defaults...
Are you sure you want to override the defaults ?(Yes or [No]):y
Class name [DEFAULT]? CIRC17
Percent bandwidth to reserve [10]?5
BRS [i 1] [dlci 17] Config>list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
interface number 1, circuit number 17
maximum queue length 10, minimum queue length 3
total bandwidth allocated 65%
total classes defined (counting one local and one default) 5


class LOCAL has 10% bandwidth allocated
  protocols and filters cannot be assigned to this class.


class DEFAULT has 40% bandwidth allocated
  the following protocols and filters are assigned:
    protocol DNA with default priority is not discard eligible
    protocol VINES with default priority is not discard eligible
    protocol IPX with default priority is not discard eligible
    protocol OSI with default priority is not discard eligible
    protocol VOFR with default priority is not discard eligible
    protocol AP2 with default priority is not discard eligible
    protocol ASRT with default priority is not discard eligible


class DEF1 has 5% bandwidth allocated
  protocol IP with priority NORMAL is not discard eligible.
```

```
 class DEF2 has 5% bandwidth allocated
  protocol ARP with priority NORMAL is not discard eligible.


 class CIRC171 has 5% bandwidth allocated
   no protocols or filters are assigned to this class.

 assigned tags:

 default class is DEFAULT with priority NORMAL
```

Example 2: Adding one class named class1 on a Frame Relay circuit

```
BRS [i 2] [dlci 128]>add
This circuit is currently using circuit defaults...
Are you sure you want to override the defaults ?(Yes or [No]): y
Class name [DEFAULT]?
Class is already allocated.
BRS [i 2] [dlci 128]>add class1
Percent bandwidth to reserve [10]?
BRS [i 2] [dlci 128]>

BRS [i 2] [dlci 128]>list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
interface number 2, circuit number 128
maximum queue length 10, minimum queue length 3
total bandwidth allocated 60%
total classes defined (counting one local and one default) 3


class LOCAL has 10% bandwidth allocated
  protocols and filters cannot be assigned to this class.


class DEFAULT has 40% bandwidth allocated
  the following protocols and filters are assigned:
    protocol IP with default priority is not discard eligible
    protocol ARP with default priority is not discard eligible
    protocol DNA with default priority is not discard eligible
    protocol VINES with default priority is not discard eligible
    protocol IPX with default priority is not discard eligible
    protocol OSI with default priority is not discard eligible
    protocol VOFR with default priority is not discard eligible
    protocol AP2 with default priority is not discard eligible
    protocol ASRT with default priority is not discard eligible

class class1 has 10% bandwidth allocated
  no protocols or filters are assigned to this class.

assigned tags:

default class is DEFAULT with priority NORMAL

BRS [i 2] [dlci 128]>
```

Assign

Use the **assign** command to assign specified tags, protocol packets, or filters to a given t-class and priority within that class. The four priority types include:

- Urgent
- High
- Normal (the default priority)
- Low.

Note: The protocol Voice over Frame Relay (VOFR) is used when voice packets are sent over a Frame Relay interface. If a circuit will carry voice packets only, assign only one t-class on the circuit and specify the protocol as VOFR. Only one t-class is allowed because one t-class does not have priority over

another. If there is more than one t-class, a t-class that does not carry voice can get control of the bandwidth and interfere with the transmission of the voice traffic. To ensure that voice traffic will receive immediate transmission, VOFR traffic and VOFR traffic only should be given the priority type *Urgent*.

Fragmentation over Frame Relay as described in the **enable fragmentation** command in the chapter "Configuring and Monitoring Frame Relay Interfaces" in *Software User's Guide* must be configured over the circuit if it will carry data traffic as well as voice. This is necessary so that large data packets will not use up the bandwidth and prevent the voice packets from getting through quickly enough.

Syntax:

assign [*protocol-class* or TAG or *filter-class*] [*class-name* or *class#*]

The **assign** command also allows you to set the Discard-eligible (DE) bit for Frame Relay frames.

Note: If this command is used for a Frame Relay circuit that is currently using default circuit definitions for traffic class handling, you will be asked whether or not you want to override the default circuit definitions. If you answer "Yes", the circuit will be changed to use circuit-specific definitions for traffic class handling and the command will be allowed. If you answer "No", the command is aborted and default circuit definitions will continue to be used for the circuit. If you want to change the default circuit definitions, you should go to the `BRS [i x][circuit defaults]Config>` command prompt.

Example 1:

```
assign IPX test
priority <URGENT/HIGH/NORMAL/LOW>: [NORMAL]? low
protocol IPX maps to class test with priority LOW Discard eligible <yes/no> [N]?
```

Example 2: Assigning a TOS filter to class1; class1 has previously been added to the configuration using the *add class* command.

```
BRS [i 2] [dlci 128]>assign ?
IP
ARP
DNA
VINES
IPX
OSI
VOFR
AP2
ASRT
TUNNELING-IP
SDLC/BSC-IP
RLOGIN-IP
TELNET-IP
NETBIOS
SNA/APPN-ISR
SNMP-IP
MULTICAST-IP
DLSW-IP
TAG1
TAG2
TAG3
TAG4
TAG5
APPN-HPR
NETWORK-HPR
HIGH-HPR
MEDIUM-HPR
LOW-HPR
```

```
XTP-IP
UDP_TCP1
UDP_TCP2
UDP_TCP3
UDP_TCP4
UDP_TCP5
TOS1
TOS2
TOS3
TOS4
TOS5
Protocol or filter name [IP]? TOS1  1
Class name [DEFAULT]? class1  2
Priority  [NORMAL]?
Frame Relay Discard Eligible  [NO]?
TOS Mask [1-FF] [FF]?
TOS Range (Low) [0-FF] [0]? 1
TOS Range (High) [1]? 3
BRS [i 2] [dlci 128]> list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
interface number 2, circuit number 128
maximum queue length 10, minimum queue length 3
total bandwidth allocated 60%
total classes defined (counting one local and one default) 3


class LOCAL has 10% bandwidth allocated
  protocols and filters cannot be assigned to this class.


class DEFAULT has 40% bandwidth allocated
  the following protocols and filters are assigned:
    protocol IP with default priority is not discard eligible
    protocol ARP with default priority is not discard eligible
    protocol DNA with default priority is not discard eligible
    protocol VINES with default priority is not discard eligible
    protocol IPX with default priority is not discard eligible
    protocol OSI with default priority is not discard eligible
    protocol VOFR with default priority is not discard eligible
    protocol AP2 with default priority is not discard eligible
    protocol ASRT with default priority is not discard eligible


class class1 has 10% bandwidth allocated
  the following protocols and filters are assigned:
    filter TOS1 with priority NORMAL is not discard eligible
       with TOS range x1 - x3 and TOS mask xFF

assigned tags:

default class is DEFAULT with priority NORMAL

BRS [i 2] [dlci 128]>show

BANDWIDTH RESERVATION currently in RAM
interface number 2, circuit number 128
maximum queue length 10, minimum queue length 3
3 current defined classes:
  class LOCAL has 10% bandwidth allocated
  class DEFAULT has 40% bandwidth allocated
  class class1 has 10% bandwidth allocated

protocol and filter assignments:

Protocol/Filter      Class              Priority    Discard Eligible
---------------      -----              --------    ----------------
IP                   DEFAULT              NORMAL            NO
ARP                  DEFAULT              NORMAL            NO
DNA                  DEFAULT              NORMAL            NO
VINES                DEFAULT              NORMAL            NO
IPX                  DEFAULT              NORMAL            NO
OSI                  DEFAULT              NORMAL            NO
VOFR                 DEFAULT              NORMAL            NO
AP2                  DEFAULT              NORMAL            NO
ASRT                 DEFAULT              NORMAL            NO
TOS1                 class1               NORMAL            NO
   with TOS range x1 - x3
```

```
    and TOS mask xFF

BRS [i 2] [dlci 128]>
```

1 Using the TOS filter requires you to enter three parameters: TOS mask, TOS range-low, and TOS range- high. Refer to the command “Add” in the chapter “Configuring and Monitoring IP” in the *Protocol Configuration and Monitoring Reference Volume 1* for a description of these parameters.

Assign-circuit

Note: Used only when configuring Frame Relay.

Use the **assign-circuit** command at the interface level to assign the specified circuit to the specified bandwidth c-class. Use the DLCI when assigning a PVC to a circuit class and the circuit name when assigning an SVC to a circuit class.

Note: You must use the **circuit** command to enable BRS on the virtual circuit and restart or reload the router before you can use this command to assign the circuit to a circuit class.

Syntax:

assign-circuit *# class name*

Change-circuit-class

Note: Used only when configuring Frame Relay.

Use the **change-circuit-class** command at the interface level to change the percentage of the bandwidth to be used by the group of circuits assigned to the specified c-class.

Syntax:

change-circuit-class *class-name %*

Change-class

Use the **change-class** command to change the amount of bandwidth configured for a bandwidth t-class.

Note: If this command is used for a Frame Relay circuit that is currently using default circuit definitions for traffic class handling, you will be asked whether or not you want to override the default circuit definitions. If you answer “Yes”, the circuit will be changed to use circuit-specific definitions for traffic class handling and the command will be allowed. If you answer “No”, the command is aborted and default circuit definitions will continue to be used for the circuit. If you want to change the default circuit definitions, you should go to the `BRS [i x][circuit defaults]Config>` command prompt.

Syntax:

change-class [*class-name* or *class#*] *%*

Circuit

Note: Used only when configuring Frame Relay.

Use the **circuit** command to configure a Frame Relay permanent virtual circuit (PVC) or switched virtual circuit (SVC). This command can only be issued from the BRS interface configuration prompt (`BRS [i #] Config>`).

Syntax:

***ci*rcuit**

Before you can use the **add-class**, **assign**, **default-class**, **del-class**, **deassign**, or **change-class** commands, you must enable BRS on the circuit and restart or reload the router.

PVC example:

```
BRS [i 1] Config> circuit
Circuit (PVC number or SVC name) to reserve bandwidth:  [16]

BRS [i 1 ] [dlci 16] Config> enable
```

SVC example:

```
BRS [i 1] Config> circuit
Circuit (PVC number or SVC name) to reserve bandwidth:  [16] svc01

BRS [i 1 ] [svc svc01] Config> enable
```

After the **enable** command is issued for the Frame-Relay circuit and the router is restarted or reloaded, the following configuration commands are available for the circuit:

add-class	deassign	enable	tag
assign	default-class	exit	untag
change-class	del-class	list	clear-block
disable	show	use-circuit-defaults	

Clear-block

Use the **clear-block** command to clear the current bandwidth reservation configuration data from SRAM.

Syntax:

clear-block

- If you enter this command from the interface prompt for PPP, all BRS configuration data is cleared for the interface.
- If you enter this command from the interface prompt for Frame Relay, BRS is no longer enabled on the interface or on any circuits of the interface, and all circuit-class configuration data and default circuit definitions for traffic class handling are cleared. However, the traffic-class configuration data for each individual circuit is not cleared and is available if you re-enable BRS on the interface.
- To clear a circuit's traffic-class configuration data, you first enter the **circuit** command from the interface-level prompt and then the **clear-block** command from the circuit-level prompt. After you have cleared the traffic-class configuration

data for each circuit, enter the **clear-block** command from the interface-level prompt to clear the circuit-class configuration data. The changes do not take effect until the router is restarted or reloaded.

Example:

```
clear-block
You are about to clear BRS configuration information for this interface
Are you sure you want to do this (Yes or No): y
BRS [i 1] Config>
```

Create-super-class

Use the **create-super-class** command to configure a t-class called *super-class* on the PPP interface or Frame Relay circuit. Only one super-class can be configured for each PPP interface or Frame Relay circuit. No bandwidth percentage is associated with the super-class. Any protocol or filter data that is assigned to a super-class will be transmitted prior to protocol or filter data assigned to any other t-classes on the PPP interface or the Frame Relay circuit. A super-class for the Voice over Frame Relay (VOFR) protocol should be configured for a circuit that transports both voice and data packets. In this environment, configuring the super-class to carry voice helps to insure that voice packets get priority.

Syntax:

create-super-class

Deactivate-IP-precedence-filtering

Use the **deactivate-ip-precedence-filtering** command to deactivate IPv4 precedence filtering processing.

Syntax:

deactivate-ip-precedence-filtering

Deassign

Use the **deassign** command to restore the queuing of the specified protocol packet or filter to the default t-class and priority.

Note: If this command is used for a Frame Relay circuit that is currently using default circuit definitions for traffic class handling, you will be asked whether or not you want to override the default circuit definitions. If you answer “Yes”, the circuit will be changed to use circuit-specific definitions for traffic class handling and the command will be allowed. If you answer “No”, the command is aborted and default circuit definitions will continue to be used for the circuit. If you want to change the default circuit definitions, you should go to the `BRS [i x][circuit defaults]Config>` command prompt.

Syntax:

deassign [*prot-class* or *filter-class*]

Deassign-circuit

Note: Used only when configuring Frame Relay.

Use the **deassign-circuit** command at the interface level to restore the queuing of the specified circuit to the default c-class.

Syntax:

deassign-c #

Default-circuit-class

Note: Used only when configuring Frame Relay.

Use the **default-circuit-class** command at the interface level to set the user-defined name of the default bandwidth c-class and the percentage of the bandwidth allocated to that class of circuits, including orphans, that are not assigned to a bandwidth c-class.

Syntax:

default-circuit-class *class-name %*

Del-circuit-class

Note: Used only when configuring Frame Relay.

Use the **del-circuit-class** command at the interface level to delete the specified bandwidth c-class.

Syntax:

del-circuit-class *class-name*

Default-class

Use the **default-class** command to set the default t-class and priority to a desired value. If no value has been previously assigned, system default values are used. Otherwise, the last previously assigned value is used.

Note: If this command is used for a Frame Relay circuit that is currently using default circuit definitions for traffic class handling, you will be asked whether or not you want to override the default circuit definitions. If you answer “Yes”, the circuit will be changed to use circuit-specific definitions for traffic class handling and the command will be allowed. If you answer “No”, the command is aborted and default circuit definitions will continue to be used for the circuit. If you want to change the default circuit definitions, you should go to the `BRS [i x][circuit defaults]Config>` command prompt.

Syntax:

default-cl [*class-name* or *class#*] *priority*

Del-class

Use the **del-class** command to delete a previously configured bandwidth t-class from the specified interface or circuit.

Note: If this command is used for a Frame Relay circuit that is currently using default circuit definitions for traffic class handling, you will be asked whether

or not you want to override the default circuit definitions. If you answer “Yes”, the circuit will be changed to use circuit-specific definitions for traffic class handling and the command will be allowed. If you answer “No”, the command is aborted and default circuit definitions will continue to be used for the circuit. If you want to change the default circuit definitions, you should go to the `BRS [i x][circuit defaults]Config>` command prompt.

Syntax:

del-class [*class-name* or *class#*]

Disable

Use the **disable** command to disable bandwidth reservation on the interface (if entered from the interface prompt) or on the circuit (if entered from the circuit prompt). The changes do not take effect until the router is restarted or reloaded.

To verify that bandwidth reservation is disabled, enter the **list** command.

Syntax:

disable

Disable-hpr-over-ip-port-numbers

Use the **disable-hpr-over-ip-port-numbers** command to disable BRS filtering of HPR over IP traffic.

Syntax:

disable-hpr-over-ip-port-numbers

To verify that BRS filtering of HPR over IP traffic is disabled, enter the **list** command.

Note: If APPN is included in the load image, you configure whether or not HPR over IP traffic will be used at the `APPN Config>` command prompt.

Enable

Use the **enable** command to enable bandwidth reservation on the interface (if entered from the interface prompt) or the circuit (if entered from the circuit prompt). The changes do not take effect until the router is restarted or reloaded.

Syntax:

enable

Note:

- When configuring BRS on a PPP interface, issue the **enable** command at the interface prompt, and then restart or reload the router before configuring any traffic classes and assigning protocols and filters to traffic classes.
- When BRS is initially enabled on a Frame Relay circuit, the circuit is initialized to use default circuit definitions for traffic class handling. Issue

the **enable** command at the interface prompt and at the circuit prompt of each circuit for which you want to define traffic classes. Then restart or reload the router before configuring circuit classes for the interface and traffic classes for each circuit. For example:

```
t 6
Gateway user configuration
Config>f brs
Bandwidth Reservation User Configuration
BRS Config>interface 1
BRS [i 1] Config>enable
Please  restart router for this command to take effect
BRS [i 1] Config>list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
interface number 1
maximum queue length 10, minimum queue length 3
total bandwidth allocated 10%
total circuit classes defined (counting one default) 1


class DEFAULT has 10% bandwidth allocated
  no circuits are assigned to this class.

default class is DEFAULT

BRS [i 1] Config>circ 16
BRS [i 1] [dlci 16] Config>enable
Defaults are in effect for this circuit.
Please  restart router for this command to take effect.
BRS [i 1] [dlci 16] Config>ex
Please  restart router for this command to take effect.
BRS [i 1] [dlci 16] Config>
*rest
Are you sure you want to  restart the gateway? (Yes or [No]): y
```

Enable-hpr-over-ip-port-numbers

Use the **enable-hpr-over-ip-port-numbers** command to enable BRS filtering of APPN-HPR over IP traffic and to configure UDP port numbers used to identify HPR over IP packets.

Note: If APPN is included in the load image, you enable HPR over IP and specify the UDP port numbers used for HPR over IP traffic at the `APPN Config>` command prompt.

Syntax:

enable-hpr-over-ip-port-numbers

Example:

```
BRS Config> enable-hpr-over-ip-port-numbers
XID exchange port number [12000]?
HPR net  trans prio port number [12001]?
HPR high trans prio port number [12002]?
HPR medium trans prio port number [12003]?
HPR low    trans prio port number [12004]?
```

XID exchange port number

This parameter specifies the UDP port number to be used for XID exchange. This port number must be the same as the one defined on other devices in the network.

Valid Values: 1024 - 65535

Default Value: 12000

Network priority port number
This parameter specifies the UDP port number to be used for network priority traffic. This port number must be the same as the one defined on other devices in the network.

Valid Values: 1024 - 65535

Default Value:12001

High exchange port number
This parameter specifies the UDP port number to be used for high priority traffic. This port number must be the same as the one defined on other devices in the network.

Valid Values: 1024 - 65535

Default Value:12002

Medium exchange port number
This parameter specifies the UDP port number to be used for medium priority traffic. This port number must be the same as the one defined on other devices in the network.

Valid Values: 1024 - 65535

Default Value:12003

Low exchange port number
This parameter specifies the UDP port number to be used for low priority traffic. This port number must be the same as the one defined on other devices in the network.

Valid Values: 1024 - 65535

Default Value:12004

Interface

Use the **interface** command to select the serial interface to which bandwidth reservation configuration commands will be applied. *Bandwidth reservation is supported on routers running PPP (Point-to-Point Protocol) and Frame Relay interfaces.*

Syntax:

interface *interface#*

Notes:

1. To enter bandwidth reservation commands for a new interface, this command must be entered ***before*** using any other bandwidth reservation configuration commands. If you have exited the bandwidth reservation prompt and wish to return to make bandwidth reservation changes to a previously configured interface, this command must again be entered first.
2. If WAN Restoral is used and BRS is configured on a primary interface, BRS should also be configured on the secondary interface. Typically when WAN Restoral is used, the secondary interface takes on the identity of the primary interface. This is not true for BRS; therefore, BRS needs to be configured on both the primary and secondary interfaces.

To enable Bandwidth Reservation on a particular interface, at the `BRS Config>` prompt, enter the number of the interface that supports the particular protocol or feature. You can then use the BRS **enable** configuration command as described in

this chapter. After enabling the interface number, you must restart or reload the 2210 for the command to take effect before you can make any other configuration changes to the interface.

Notes:

1. If you are configuring BRS on a Frame Relay interface, you can use the **circuit** command to select circuits and enable bandwidth reservation on those circuits before you restart or reload the router.

List

Use the **list** command to display currently defined bandwidth classes and their guaranteed percentage rates.

The **list** command and **show** command are similar. The **list** command displays the current SRAM definitions and the **show** command displays the current RAM definitions.

Syntax:

list *interface#*

Depending on the prompt at which you issue the **list** command, various outputs are displayed. You can issue the **list** command from the following prompts:

- `BRS [i 1] [dlci 16] Config>`
- `BRS [i 1] Config>`
- `BRS Config>`
- `BRS [i 1] [circuit defaults] Config>`

Note: When you use this command from a Frame Relay circuit prompt (`BRS [i x] [dlci y] Config>`) it indicates if the circuit is using default circuit definitions or circuit-specific definitions for traffic class handling. If the circuit is using default circuit definitions, the traffic class, protocol, filter, and tag assignments currently defined for default circuit definitions are displayed. However, if you want to alter the default circuit definitions, you need to get to the `BRS[i x] [circuit defaults] Config>` prompt to make changes.

At the BRS interface level prompt (`BRS [i 0]`) for PPP interfaces and at the BRS circuit level prompt (`BRS [i 0] [dlci 16] Config>`) for Frame Relay interfaces, the **list** command lists the traffic classes, their configured bandwidth percentages, and the assigned protocols and filters.

At the BRS interface level prompt for Frame Relay, the **list** command lists the circuit classes, their configured bandwidth percentages, and the assigned circuits.

Example 1

```
BRS Config>list
 Bandwidth Reservation is available for 2 interfaces.

 Interface   Type            State
 ---------   ----            -----
         1   FR              Enabled
         2   PPP             Enabled

 The use of HPR over IP port numbers is disabled

 BRS Config>interface 1
```

```
BRS [i 1] Config>list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
interface number 1
maximum queue length 10, minimum queue length 3
total bandwidth allocated 10%
total circuit classes defined (counting one default) 1


class DEFAULT has 10% bandwidth allocated
  the following circuits are assigned:
      17
      16  using defaults.
      18  using defaults.

default class is DEFAULT

BRS [i 2] Config>exit
BRS Config>interface 2
BRS [i 2] Config>list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
interface number 2
maximum queue length 10, minimum queue length 3
total bandwidth allocated 50%
total classes defined (counting one local and one default) 2


class LOCAL has 10% bandwidth allocated
  protocols and filters cannot be assigned to this class.


class DEFAULT has 40% bandwidth allocated
  the following protocols and filters are assigned:
    protocol IP with default priority
    protocol ARP with default priority
    protocol DNA with default priority
    protocol VINES with default priority
    protocol IPX with default priority
    protocol OSI with default priority
    protocol VOFR with default priority
    protocol AP2 with default priority
    protocol ASRT with default priority

assigned tags:

default class is DEFAULT with priority NORMAL

BRS [i 2] Config>
```

Example 2

```
BRS [i 1] [dlci 17] Config>list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
maximum queue length 10, minimum queue length 3
total bandwidth allocated 60%
total classes defined (counting one local and one default) 3


class LOCAL has 10% bandwidth allocated
  protocols and filters cannot be assigned to this class.


class DEFAULT has 40% bandwidth allocated
  the following protocols and filters are assigned:
   protocol ASRT with priority NORMAL is not discard eligible
   filter NETBIOS with priority NORMAL is not discard eligible


class CLASS1 has 10% bandwidth allocated
  the following protocols and filters are assigned:
   protocol IP with priority NORMAL is not discard eligible
   protocol ARP with priority NORMAL is not discard eligible
   protocol DNA with priority NORMAL is not discard eligible
   protocol VINES with priority NORMAL is not discard eligible
   protocol IPX with priority NORMAL is discard eligible
   protocol OSI with priority NORMAL is not discard eligible
   protocol VOFR with priority NORMAL is not discard eligible
   protocol AP2 with priority NORMAL is not discard eligible
```

Example 3

```
BRS [i 1] [circuit defaults] Config>list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
interface number 1, default circuit
maximum queue length 10, minimum queue length 3
total bandwidth allocated 70%
total classes defined (counting one local and one default) 4


class LOCAL has 10% bandwidth allocated
  protocols and filters cannot be assigned to this class.



class DEFAULT has 40% bandwidth allocated
  the following protocols and filters are assigned:
    protocol DNA with default priority is not discard eligible
    protocol VINES with default priority is not discard eligible
    protocol IPX with default priority is not discard eligible
    protocol OSI with default priority is not discard eligible
    protocol VOFR with default priority is not discard eligible
    protocol AP2 with default priority is not discard eligible
    protocol ASRT with default priority is not discard eligible


class DEF1 has 10% bandwidth allocated
  protocol IP with priority NORMAL is not discard eligible.


class DEF2 has 10% bandwidth allocated
 protocol ARP with priority NORMAL is not discard eligible.

assigned tags:

default class is DEFAULT with priority NORMAL

 BRS [i 1] [circuit defaults] Config>
```

Example 4

```
BRS Config>list
 Bandwidth Reservation is available for 2 interfaces.

 Interface   Type            State
 ---------   ----            -----
         1   FR              Enabled
         2   PPP             Enabled

 The use of HPR over IP port numbers is enabled.

    Transmission Type   Port Number
    -----------------   -----------
        XID exchange        12000
        HPR network         12001
        HPR high            12002
        HPR medium          12003
        HPR low             12004
```

Queue-length

Use the **queue-length** command to set the number of packets that can be queued in each BRS priority queue. Each BRS class has a priority value assigned to its protocols, filters, and tags, and each priority queue can store the number of packets that you specify with this command.

Syntax:

queue-length *maximum-length minimum-length*

This command sets the maximum number of buffers that can be queued in each BRS priority queue as well as the maximum number that can be queued in each BRS priority queue when there is a shortage of router input buffers.

If you issue **queue-length** for a PPP interface, the command sets the queue-length values for each priority queue of each BRS t-class that is defined for the interface.

Configuring BRS and Priority Queuing

If you issue **queue-length** for a Frame Relay interface (at the prompt: `BRS [i 0] Config>`), the command sets the default queue-length values for each priority queue of each BRS t-class that is defined for each permanent virtual circuit of the interface.

If you issue **queue-length** for a Frame-Relay PVC (at a prompt like this: `BRS [i 0] [dlci 16] Config>`) the command sets the queue length values for each priority queue of each BRS t-class that is defined for the PVC. These values override the default queue length values set for the Frame Relay interface.

Attention: Do not use this command unless it is essential to do so. The default values for queue length are the recommended values for most users. If you set the values for queue length too high, you may seriously degrade the performance of your router.

Set-circuit-defaults

Use the **set-circuit-defaults** command to access the commands used to define default circuit definitions for traffic class handling. These default circuit definitions can then be used by any Frame Relay circuits on the interface that can use the same traffic classes and protocol, filter, and tag assignments.

Syntax:

set-circuit-defaults

Show

Use the **show** command to display currently defined bandwidth classes stored in RAM.

Syntax:

show *interface#*

Depending on the prompt at which you issue the **show** command, various outputs are displayed. You can issue the **show** command from the following prompts:

- `BRS [i x] Config>` - interface level prompt for interface number *x*.
- `BRS [i x] [dlci y] Config>` - circuit level prompt for circuit *y* on Frame Relay interface number *x*. The following example shows the output of the show command from the circuit level prompt.

```
BRS [i 1] [dlci 17] Config>show

Protocol/Filter      Class             Priority    Discard Eligible
---------------      -----             --------    ----------------
IP                   CLASS1              NORMAL            NO
ARP                  CLASS1              NORMAL            NO
DNA                  CLASS1              NORMAL            NO
VINES                CLASS1              NORMAL            NO
IPX                  CLASS1              NORMAL            YES
OSI                  CLASS1              NORMAL            NO
VOFR                 CLASS1              NORMAL            NO
AP2                  CLASS1              NORMAL            NO
ASRT                 DEFAULT             NORMAL            NO
NETBIOS              DEFAULT             NORMAL            NO
```

At the interface prompt for PPP and the circuit prompt for Frame Relay, traffic class information is displayed. At the interface prompt for Frame Relay, circuit class information is displayed.

Notes:

1. When you use this command from a Frame Relay circuit prompt (`BRS [i x] [dlci y] Config>`) it indicates if the circuit is using default circuit definitions or circuit-specific definitions for traffic class handling. If the circuit is using default circuit definitions, the traffic class, protocol, filter, and tag assignments currently defined for default circuit definitions are displayed. However, if you want to alter the default circuit definitions, you need to get to the `BRS [i x] [circuit defaults] Config>` prompt to make changes.
2. This command cannot be used from the `BRS [i x] [circuit defaults] Config>` prompt.

Tag

Use the **tag** command to assign a MAC filter item that has been tagged during the configuration of the MAC filtering feature to the next available BRS tag name. The BRS tag names are TAG1, TAG2, TAG3, TAG4, and TAG5. You use the BRS tag name on the assign command to assign the tag to a BRS traffic class.

Syntax:

tag *mac_filter_tag#*

Use the **list** command to list which MAC filter tags have been assigned to a BRS tag name and which BRS tag names have been assigned to a bandwidth traffic class.

Note: If this command is used for a Frame Relay circuit that is currently using default circuit definitions for traffic class handling, you will be asked whether or not you want to override the default circuit definitions. If you answer “Yes”, the circuit will be changed to use circuit-specific definitions for traffic class handling and the command will be allowed. If you answer “No,” the command is aborted and default circuit definitions will continue to be used for the circuit. If you want to change the default circuit definitions, you should go to the `BRS [i x][circuit defaults]Config>` command prompt.

Untag

Use the **untag** command to remove the MAC filter tag number and BRS tag name relationship. A tag can be removed only if its corresponding BRS tag name is not assigned to a bandwidth traffic class.

Syntax:

untag *mac_filter_tag#*

Use the **list** command to show which MAC filter tags are assigned to a BRS tag name and which BRS tag names are assigned to a traffic class.

Note: If this command is used for a Frame Relay circuit that is currently using default circuit definitions for traffic class handling, you will be asked whether or not you want to override the default circuit definitions. If you answer “Yes”, the circuit will be changed to use circuit-specific definitions for traffic class handling and the command will be allowed. If you answer “No”, the command is aborted and default circuit definitions will continue to be used for the circuit. If you want to change the default circuit definitions, you should go to the `BRS [i x][circuit defaults]Config>` command prompt.

Use-circuit-defaults

Use the **use-circuit-defaults** command at the circuit level to delete the circuit-specific definitions and use the circuit default definitions for traffic-class handling. You will be prompted to confirm that you want to use the circuit defaults.

Syntax:

use-circuit-defaults

Notes:

1. This command is used only when configuring Frame Relay
2. The router must be restarted or reloaded for the defaults to become operational.

Example:

```
BRS [i 1] [dlci 17] Config>use-circuit-defaults
This circuit is currently NOT using circuit defaults...
 Are you sure you want to delete current definitions and use defaults ? (Yes or
 [No]): y
 Defaults are in effect for this circuit.
 Please  restart router for this command to take effect.
 BRS [i 1] [dlci 17] Config>
 *rest
 Are you sure you want to  restart the gateway? (Yes or [No]): y
```

Accessing the Bandwidth Reservation Monitoring Prompt

To access bandwidth reservation monitoring commands and to monitor bandwidth reservation on your router, do the following:

1. At the OPCON prompt (*), type **talk 5**.
2. At the GWCON prompt (+), type **feature brs.**
3. At the BRS> prompt, type **interface #**, where # is the number of the interface that you want to monitor. This takes you to the BRS interface-level prompt, BRS [i x]>, where x is the interface number.
4. For Frame Relay only, type **circuit #** at the interface prompt to specify the circuit on this interface that you want to monitor.

   This takes you to the circuit-level prompt BRS [i x] [dlci y]>, where x is the interface number and y is the circuit number.
5. At the prompt, type the appropriate monitoring command. (Refer to “Bandwidth Reservation Monitoring Commands” on page 43.)

   The **talk 5 (t 5)** command lets you access the monitoring process.

   The **feature brs** command lets you access the BRS monitoring process. You can enter this command by using either the feature name (brs) or number (1).

   The **interface #** command selects the particular interface that you want to monitor for bandwidth reservation.

   The **circuit #** command selects the DLCI of a Frame Relay permanent virtual circuit (PVC).

   To return to the GWCON prompt at any time, type the **exit** command at the BRS> prompt.

   Once you access the bandwidth reservation monitoring prompt (BRS>), you can enter any of the specific monitoring commands described in Table 4 on page 43.

Bandwidth Reservation Monitoring Commands

This section summarizes and explains the Bandwidth Reservation monitoring commands. 4 shows the Bandwidth Reservation monitoring commands. The commands that can be used differ depending on the BRS monitoring prompt (`BRS>`, `BRS [i x]>`, or `BRS [i x] [dlci y]>`).

Table 4. Bandwidth Reservation Monitoring Command Summary

Command	Used Only With FR	Function
? (Help)		Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix
Circuit	yes	Selects the DLCI of a Frame Relay permanent virtual circuit (PVC). To monitor Frame Relay bandwidth reservation traffic, you must be at the circuit prompt level.
Clear		Clears the current t-class counters and stores them as **last** t-class counters. Counters are listed by class.
Clear-circuit-class	yes	Clears the current c-class counters and stores them as **last** c-class counters. Counters are listed by class.
Counters		Displays the current t-class counters.
Counters-circuit-class	yes	Displays the current c-class counters.
Interface		Selects the interface to monitor. **Note:** This command must be entered before using any other bandwidth reservation monitoring commands.
Last		Displays the last saved t-class counters.
Last-circuit-class	yes	Displays the last saved c-class counters.
Exit		Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix

Circuit

Note: Used only when monitoring Frame Relay.

Use the **circuit** command to select the DLCI of a Frame Relay PVC for monitoring. This command can be issued only from the BRS interface monitoring prompt (`BRS [i #]>`).

Syntax:

circuit *permanent-virtual-circuit-#*

After the Frame Relay circuit has been selected, the following commands can be used at the circuit prompt:

```
CLEAR
COUNTERS
LAST
EXIT
```

Clear

Use the **clear** command to save the current bandwidth reservation t-class counters so that they can be retrieved using the **last** command and clear the values. The counters are kept on a bandwidth traffic class basis.

Syntax:

clear

Clear-Circuit-Class

Note: Used only when monitoring Frame Relay.

Use the **clear-circuit-class** command to save the current bandwidth reservation c-class counters so that they can be retrieved using the **last-circuit-class** command and clear the values. The counters are kept on a circuit class basis.

Syntax:

clear-circuit-class

Counters

Use the **counters** command to display statistics describing bandwidth reservation traffic for the traffic classes configured for a PPP interface or Frame Relay circuit.

Syntax:

counters

Example:

counters

```
Bandwidth Reservation Counters
interface number 1
Class            Pkt Xmit     Bytes Xmit     Bytes Ovfl     Pkt Ovfl    Q_len

LOCAL               10            914             0            0
   LOW               0              0             0            0        0
   NORMAL           10            914             0            0        0
   HIGH              0              0             0            0        0
   URGENT            0              0             0            0        0
DEFAULT             55           5555             0            0
   LOW               0              0             0            0        0
   NORMAL           20           5020             0            0        0
   HIGH              0              0             0            0        0
   URGENT           35            535             0            0        0
CLASS_1              5            910             0            0
   LOW               0              0             0            0        0
   NORMAL            5            910             0            0        0
   HIGH              0              0             0            0        0
   URGENT            0              0             0            0        0
CLASS_2             70           4123             0            0
   LOW              10            617             0            0        0
   NORMAL           55           3117             0            0        0
   HIGH              0              0             0            0        0
   URGENT            5            389             0            0        0

TOTAL              140          11502             0            0
```

Bytes Ovfl

Lists the number of bytes for packets that could not be transmitted because either the maximum queue-length was reached for a priority queue or the packet could not be queued because the priority queue was at the minimum queue length threshold and the packet came from an interface that was running low on receive buffers.

Pkt Ovfl
Lists the number of packets that could not be transmitted because either the maximum queue-length was reached for a priority queue or the packet could not be queued because the priority queue was at the minimum queue length threshold and the packet came from an interface that was running low on receive buffers.

Q_len The current number of packets waiting for transmit on each of the priority queues within each traffic class.

Counters-circuit-class

Note: Used only when monitoring Frame Relay.

Use the **counters-circuit-class** command to display statistics for the traffic classes configured for a Frame Relay circuit.

Syntax:

counters-circuit-class

Example:

```
counters-circuit-class

Bandwidth Reservation Circuit Class Counters
Interface 1


Class         Pkt Xmit     Bytes Xmit     Bytes Ovfl
DEFAULT             25           3402             26
CIRCLASS1            1             56              0
CIRCLASS2            0              0              0

TOTAL               26           3458             26
```

Interface

Use the **interface** command to select the serial interface to which bandwidth reservation monitoring commands will be applied. *Bandwidth reservation is supported on routers running the PPP (Point-to-Point Protocol) and Frame Relay interfaces.*

Syntax:

interface *interface#*

Note: To enter bandwidth reservation commands for a new interface, this command must be entered before using any other bandwidth reservation monitoring commands. If you have exited the bandwidth reservation monitoring prompt (BRS>) and want to return to monitor bandwidth reservation, you must again enter this command first.

To monitor Bandwidth Reservation on a particular interface, at the BRS> monitoring prompt, type the number of the interface. You can then use bandwidth reservation monitoring commands as described in this chapter.

Last

Use the **last** command to display the last saved t-class statistics. The t-class statistics are displayed in the same format as they are for the **counters** command.

Syntax:

last

Last-circuit-class

Note: Used only when monitoring Frame Relay.

Use the **last-circuit-class** command to display the last saved circuit class statistics. The c-class statistics are displayed in the same format as they are for the **counters-circuit-class** command.

Syntax:

last-circuit-class

Chapter 3. Using MAC Filtering

This chapter describes how to use medium access control (MAC) for specifying packet filters to be applied to packets during processing. It includes the following sections:

Filters are a set of rules applied to a packet to determine how the packet should be handled during bridging. MAC filtering affects only bridged traffic.

Note: MAC Filtering is allowed on tunnel traffic.

During the filtering process, packets are processed, filtered, or tagged during bridging. The actions are:

- **Processed** – Packets are permitted to pass unaffected through the bridge.
- **Filtered** – Packets are not permitted to pass through the bridge.
- **Tagged** – Packets are allowed to pass through the bridge, but are marked with a number in the range 1 through 64 based on a configurable parameter.

A MAC filter consists of the following three objects:

1. Filter-item – which is a single rule that is applied to the address field or an arbitrary window of data within a packet. The result of applying the rule is either a true (successful match) or false (no match) condition.
2. Filter-list – which contains a list of one or more filter-items.
3. Filter – which contains a set of filter-lists.

MAC Filtering and DLSw Traffic

You can filter incoming LLC traffic for the DLSw network by implementing MAC Filtering.

To set up a filter for LLC, use the *Bridge Net* number as the interface number for the filter. Determine the Bridge Net number by adding two to the number of interfaces configured for your router. Enter the **list devices** command at the `Config>` prompt, or enter **configuration** at the + prompt to see a list of interfaces.

In the following example, the Bridge Net number is 7.

```
Ifc 0 Ethernet                    CSR  81600, CSR2  80C00, vector 94
Ifc 1 WAN X.25                    CSR  81620, CSR2  80D00, vector 93
Ifc 2 WAN X.25                    CSR  81640, CSR2  80E00, vector 92
Ifc 3 WAN PPP                     CSR 381620, CSR2 380D00, vector 125
Ifc 4 WAN Frame Relay             CSR 381640, CSR2 380E00, vector 124
Ifc 5 Token Ring                  CSR 600000, vector 95
```

When you set up a filter for the Bridge Net, for example, the router does not drop frames that match exclusive filters. Instead, it forwards those frames to the bridge.

MAC Filtering Parameters

You can specify some or all of the following parameters to create a filter:

- Source MAC address or destination MAC address
- Data to be matched within the packet
- Mask to be applied to the packet's fields to be filtered
- Interface number
- Input/Output designation
- Include/Exclude/Tag designation
- Tag value (if the tag designation is given)

Filter-Item Parameters

The following parameters are used to construct an address-filter-item:

- Address Type: SOURCE or DESTINATION
- Tag: a *tag-value*
- Address Mask: a *hex-mask*

Each filter-item specifies an address type (either SOURCE or DESTINATION) to match against the type in the packet.

The address mask is a string of numbers entered in hex, which is used in comparing the packet's addresses. The mask is applied to the SOURCE or DESTINATION MAC address of the packet before comparing it against the specified MAC address.

The address mask must be of equal length to the MAC address and specifies the bytes that are to be logically ANDed with the bytes in the MAC address before the equality comparison to the specified MAC address is made. If no mask is specified, it is assumed to be all 1s.

Filter-List Parameters

The following parameters are used to construct a filter-list:

- Name: an *ASCII-string*
- Filter-item list: *filter-item 1* . . . *filter-item n*
- Action: INCLUDE, EXCLUDE, TAG(*n*)

A filter-list is built from one or more filter-items. Each filter-list is given a unique name.

Applying a filter-list to a packet consists of comparing each filter-item in the order in which the filter-items were added to the list. If any filter-item in the list returns a TRUE condition then the filter-list will return its designated action.

Filter Parameters

The following parameters are used to construct a filter:

- Filter-list names: *ASCII-string 1* . . . *ASCII-string n*
- Interface number: an *IFC-number*

- Port direction: INPUT or OUTPUT
- Default action: INCLUDE, EXCLUDE, or TAG
- Default tag: a *tag-value*

A filter is constructed by associating a group of filter-list names with an interface number and assigning an INPUT or OUTPUT designation. The application of a filter to a packet means that each of the associated filter-lists should be applied to packets being received (INPUT) or sent (OUTPUT) on the specified numbered interface.

When a filter evaluates a packet to an INCLUDE condition, the packet is forwarded. When a filter evaluates a packet to an EXCLUDE condition, the packet is dropped. When a filter evaluates to a TAG condition, the packet being considered is forwarded with a tag.

An additional parameter of each filter is the default action, which is the result of non-match for all of its filter-lists. This default action is INCLUDE. It can be set to INCLUDE, EXCLUDE, or TAG. In addition, if the default action is TAG, a tag value is also given.

Using MAC Filtering Tags

The following list includes some uses of MAC filtering tags

- MAC Address filtering is handled jointly by bandwidth reservation and the MAC Filtering feature (MCF) using tags. A user with bandwidth reservation is able to categorize bridge traffic, for example, by assigning a tag to it.
- The tagging process is done by creating a filter-item in the MAC Filtering configuration console and then assigning a tag to it. This tag is then used to set up a bandwidth class for all packets associated with this tag. Tag values must currently be in the range 1 to 64.
- Once a tagged filter has been created in the MAC Filtering configuration process, the Bandwidth Reservation (BRS) **tag** configuration command is used to assign a BRS tag name (TAG1, TAG2, TAG3, TAG4, or TAG5) to the MAC filter tag number. The BRS tag name is then used on the BRS **assign** configuration command to assign the corresponding MAC filter to a bandwidth traffic class and priority.
- Up to 5 tagged MAC addresses can be set from 1 to 5. TAG1 will be searched for first, then TAG2, all the way to TAG5.

Tags can also refer to “groups” in IP Tunnel. IP Tunnel end-points can belong to any number of groups, with packets assigned to a particular group through the tagging feature of MAC address filtering.

Chapter 4. Configuring and Monitoring MAC Filtering

This chapter describes how to access the MAC Filtering configuration and monitoring prompts and how to use the available commands. It includes the following sections:

Accessing the MAC Filtering Configuration Prompt

Use the **feature** command from the CONFIG process to access the MAC filtering configuration commands. The **feature** command lets you access configuration commands for specific features outside the protocol and network interface configuration processes.

Enter a question mark after the **feature** command to obtain a listing of the features available for your software release. For example:

```
Config> feature ?
WRS
BRS
MCF
Feature name or number [MCF]?
```

To access the MAC filtering configuration prompt, enter the **feature** command followed by the *feature number* (3) or *short name* (MCF). For example:

```
Config> feature mcf
MAC Filtering user configuration
Filter config>
```

Once you access the MAC filtering configuration prompt, you can begin entering specific configuration commands. To return to the CONFIG prompt at any time, enter the **exit** command at the MAC filtering configuration prompt.

MAC Filtering Configuration Commands

This section summarizes the MAC filtering configuration commands. Enter these commands at the `Filter config>` prompt.

Use the following commands to configure the MAC filtering feature.

Table 5. MAC Filtering Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page xxix.
Attach	Adds a filter list to a filter.
Create	Creates a filter list or an INPUT or OUTPUT filter.
Default	Sets the default action for the specified filter to EXCLUDE, INCLUDE, or TAG.
Delete	Removes all information associated with a filter list. Also deletes a filter that was created using the create filter command.
Detach	Removes a filter list from a filter.
Disable	Disables MAC Filtering entirely or disables a particular filter.
Enable	Enables MAC Filtering entirely or enables a particular filter.

Table 5. MAC Filtering Configuration Command Summary (continued)

Command	Function
List	Lists a summary of all the filter lists and filters configured by the user. Also generates a list of attached filter lists for this filter and all subsequent information for the filter.
Move	Reorders the filter lists attached to a specified filter.
Reinit	Re-initializes the entire MAC Filtering system from an updated configuration, without affecting the rest of the router.
Set-Cache	Changes the cache size for a filter.
Update	Adds or deletes information from a specific filter list. Brings you to a menu of appropriate subcommands.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Attach

Use the **attach** command to add a filter-list to a filter.

A filter is constructed by associating a group of filter-lists with an interface number. A filter-list is built from one or more filter-items.

Syntax:

attach *filter-list-name filter-number*

Create

Use the **create** command to create a filter-list or an INPUT or OUTPUT filter.

Syntax:

create list *filter-list-name*

filter [input or output] *interface-number*

list *filter-list-name*
Creates a filter-list. Lists are named by a unique string (Filter-list-name) of up to 16 characters of the user’s choice. This name is used to identify a filter-list that is being built. This name is also used with other commands associated with the filter-list.

filter [input or output] *interface-number*
Creates a filter and places it on the network associated with the INPUT or OUTPUT direction on the interface given by an interface number. By default this filter is created with no attached filter-lists, has a default action of INCLUDE and is ENABLED.

Default

Use the **default** command to set the default action for the filter with a specified filter number to exclude, include, or tag.

Syntax:

default exclude *filter-number*

include *filter-number*

tag *tag-number filter-number*

exclude *filter-number*
Sets the default action for the filter with a specified filter number to exclude.

include *filter-number*
Sets the default action for the filter with a specified filter number to include.

tag *tag-number filter-number*
Sets the default action for the filter with the specified filter number to TAG and sets the associated tag value to tag number.

Delete

Use the **delete** command to remove all information associated with a filter-list and to free an assigned string as a name for a new filter-list. If filter-list is attached to a filter that has already been created by the user, then this command will display an error message on the console without deleting anything. In addition all filter-items belonging to this list are also deleted

This command also deletes a filter that was created using the **create filter** command.

Syntax:

delete list *filter-list*

filter *filter-number*

list *filter-list*
Removes all information associated with a filter-list and frees an assigned string as a name for a new filter-list. The filter-list must be a string entered by a previous **create list** command.

If the filter-list is attached to a filter that has already been created by the user, then this command will display an error message on the console without deleting anything. All filter-items belonging to this list are also deleted when this command is used.

filter *filter-number*
Deletes a filter that was created using the **create filter** command.

Detach

Use the **detach** command to delete a filter-list name (filter-list parameter) from a filter (filter-number parameter).

Syntax:

detach *filter-list-name filter-number*

Disable

Use the **disable** command to disable MAC Filtering entirely or to disable a particular filter.

Syntax:

disable all

filter *filter-number*

all Disables MAC Filtering entirely. Filters are still set as ENABLED, however, if they were enabled previously.

filter *filter-number*
Disables a particular filter. The filter-number parameter corresponds to the numbers displayed in the **list filters** command.

Enable

Use the **enable** command to enable MAC Filtering entirely or to enable a particular filter.

Syntax:

enable all

filter *filter-number*

all Enables MAC Filtering entirely, although filters themselves may still be set to DISABLED.

filter *filter-number*
Enables a particular filter. The filter-number parameter corresponds to the numbers displayed in the **list filters** command.

List

Use the **list** command to list a summary of all the filter-lists and filters configured by the user. A list of all the filter-lists attached to a filter is not given. Other information displayed includes:

- A list containing the state of the filtering system (ENABLE, DISABLE)
- The set of configured filter-list records
- Each of the configured filter records.

In addition, the following information is displayed for each filter:

- Filter number
- Interface number
- Filter direction (INPUT, OUTPUT)
- Filter state (ENABLE, DISABLE)
- Filter default action (TAG, INCLUDE, EXCLUDE).

This command also generates a list of attached filter-lists for this filter and all subsequent information for the filter.

Syntax:

list all

filter *filter-number*

all Displays a summary of all the configured filter-lists and filters.

filter *filter-number*
Generates a list of attached filter-lists for the specified filter and all subsequent information for the filter.

Move

Use the **move** command to reorder the filter-lists attached to a specified filter (given by filter-number parameter). The list given by Filter-list-name1 is moved immediately before the list given by Filter-list-name2.

Syntax:

move *filter-list-name1 filter-list-name2 filter-number*

Reinit

Use the **reinit** command to re-initialize the entire MAC Filtering system from an updated configuration, without affecting the rest of the router.

Syntax:

reinit

Set-Cache

Use the **set-cache** command to change the default cache size (16) to a number in the range 4 to 32768.

Syntax:

set-cache *cache-size filter-number*

Update

Use the **update** command to add information to or delete information from a specific filter-list. Using this command with the desired filter-list-name brings you to the `Filter filter-list-name Config>` prompt for that specific filter-list. From this new prompt you can then change information in the specified list.

The new prompt level is used to add or delete filter-items from filter-lists. The order in which the filter-items are specified for a given filter-list is important as it determines the order in which the filter-items are applied to a packet.

Syntax:

update *filter-list-name*

Update Subcommands

This section summarizes the MAC filtering configuration subcommands. Enter these subcommands at the `Filter filter-list-name config>` prompt.

Table 6. Update Subcommands Summary

Subcommand	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Add	Adds source or destination MAC address filters or a window filter. Adds filter-items to a filter-list.
Delete	Removes filter-items from a filter-list.

Table 6. Update Subcommands Summary (continued)

Subcommand	Function
List	Lists a summary of all the filter-lists and filters configured by the user. Also generates a list of attached filter-lists for this filter and all subsequent information for the filter.
Move	Reorders the filter-lists attached to a specified filter.
Set-Action	Sets a filter-item to evaluate the INCLUDE, EXCLUDE or TAG (with a tag-number option) condition.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Use the following subcommands to update a filter-list.

Add

Use the **add** subcommand to add filter-items to a filter-list. This subcommand specifically lets you add a hexadecimal number to compare against the source or destination MAC address, or a sequence of window data with a mask to compare against a packet data.

The order in which the filter-items are added to a given filter-list is important because it determines the order in which the filter-items are applied to a packet.

Each use of the **add** subcommand creates a filter-item within the filter-list. The first filter-item created is assigned filter-item-number 1, the next one is assigned number 2, and so on. After you enter a successful **add** subcommand, the router displays the number of the filter-item just added.

The first match that occurs stops the application of filter-items, and the filter-list evaluates to INCLUDE, EXCLUDE, or TAG, depending on the designated action of the filter-list. If none of the filter-items of a filter-list produces a match, then the default action (INCLUDE, EXCLUDE or TAG) of the filter is returned.

Syntax: add source *hex-MAC-addr hex-Mask*

destination *hex-MAC-addr hex-Mask*

window MAC *offset-value hex-data hex-mask*

window INFO *offset-value hex-data hex-mask*

source *hex-MAC-addr hex-Mask*
Adds a hexadecimal number to compare against the source MAC address. ***hex-MAC-addr*** must be an even number of hex digits with a maximum of 16 digits and should be entered without a 0x in front.

The hex-mask parameter must be the same length as hex-MAC-address and is logically ANDed with the designated MAC address in the packet. The default hex-mask argument is to be all binary 1s.

The hex-MAC-addr parameter can be specified in canonical or noncanonical bit order. A canonical bit order is specified as just a hex number (for example, 000003001234). It may also be represented as a series of hex digits with a hyphen (-) between every two digits (for example, 00-00-03-00-12-34).

A noncanonical bit order is specified as a series of hex digits with a colon (:) between every two digits (for example, 00:00:C9:09:66:49). MAC addresses of filter-items will always be displayed using either a hyphen (-) or a colon (:) to distinguish canonical from noncanonical representations.

destination *hex-MAC-addr hex-Mask*
Acts identically to the add source subcommand, with the exception that the match is made against the destination rather than the source MAC address of the packet.

window MAC *offset-value hex-data hex-mask*
Adds a sliding window filter-item using the specified offset (computed from the beginning of the frame) that matches the hex data with the mask against packet data.

window INFO *offset-value hex-data hex-mask*
Similar to the **add window mac** command, except that the offset is computed with respect to the beginning of the information field.

Delete

Use the **delete** subcommand to remove filter-items from a filter-list. You delete filter-items by specifying the filter-item-number assigned to the item when it was added.

When the **delete** subcommand is used, any gap created in the number sequence is filled in. For example, if filter-items 1, 2, 3, and 4 exist and filter-item 3 is deleted, then filter-item 4 will be renumbered to 3.

Syntax:

delete	*filter-item-number*

List

Use the **list** subcommand to print out a listing of all the filter-item records. The following information about each MAC-Address filter-item is displayed:

- MAC address and address mask in canonical or noncanonical form.
- filter-item numbers
- address type (source or destination)
- filter-list action

Syntax:

list	canonical
	noncanonical
	mac-address canonical
	mac-address noncanonical
	window

canonical
Prints out a listing of all the filter-item records within a filter-list, giving the item numbers, the address type (SRC, DST), the MAC address in canonical form, and the address mask in canonical form. It also gives the filter-list action.

mac-address canonical
Prints out a listing of all the filter-item records within a filter-list, giving the

item numbers, the address type (SRC, DST), the MAC address in canonical form, and the address mask in canonical form. In addition the filter-list action is given.

noncanonical
Prints out a listing of all the filter-item records within a filter-list, giving the item numbers, the address type (SRC, DST), the MAC address in noncanonical form, and the address mask in noncanonical form. It also gives the filter-list action.

mac-address noncanonical
Prints out a listing of all the filter-item records within a filter-list, giving the item numbers, the address type (SRC, DST), the MAC address in noncanonical form, and the address mask in noncanonical form. It also gives the filter-list action.

window
Prints out a listing of all the sliding window filter-item records within a filter-list, giving the item numbers, base, offset, data, and mask. It also gives the filter-list action.

Move

The **move** subcommand reorders filter-items within the filter-list. The filter-item whose number is specified by *filter-item-name1* is moved and renumbered to be just before *filter-item-name2*.

Syntax:

move *filter-item-name1 filter-item-name2*

Set-Action

The **set-action** subcommand lets you set a filter-item to evaluate the INCLUDE, EXCLUDE, or TAG (with a tag-number option) condition. If one of the filter-items of the filter-list matches the contents of the packet being considered for filtering, the filter-list will evaluate to the specified condition. The default setting is INCLUDE.

Syntax:

set-action [INCLUDE or EXCLUDE or TAG] *tag-number*

Accessing the MAC Filtering Monitoring Prompt

Use the **feature** command from the GWCON process to access the MAC filtering monitoring commands. The **feature** command lets you access monitoring commands for specific router features outside of the protocol and network interface monitoring processes.

Enter a question mark after the **feature** command to obtain a listing of the features available for your software release. For example:

```
+ feature ?
WRS
BRS
MCF
```

To access the MAC filtering monitoring prompt, enter the **feature** command followed by the feature number (3) or short name (MCF). For example:

```
+ feature mcf
MAC Filtering user monitoring
Filter>
```

Once you access the MAC filtering monitoring prompt, you can begin entering specific monitoring commands. To return to the GWCON prompt at any time, enter the **exit** command at the MAC Filtering monitoring prompt.

MAC Filtering Monitoring Commands

This section summarizes the MAC filtering monitoring commands. Enter these commands at the `Filter>` prompt.

Table 7. MAC Filtering Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Clear	Clears the "per filter" statistics listed in the list filter command.
Disable	Disables MAC Filtering globally or on a "per filter" basis.
Enable	Enables MAC Filtering globally or on a "per filter" basis.
List	Lists a summary of statistics and settings for each filter currently running in the router.
Reinit	Re-initializes the entire MAC Filtering system from an updated configuration, without affecting the rest of the router.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Use the following commands to monitor the MAC filtering feature.

Clear

Use the **clear** command to clear filter statistics.

Syntax:

clear all

filter *filter-number*

all Clears the statistics listed by the **list all** command.

filter *filter-number*
Clears the statistics listed by the **list filter** command.

Disable

Use the **disable** command to disable MAC filtering globally. This command does not individually disable each filter.

The command also disables a filter as specified by filter-number. This filter is disabled without modifying configuration records. If no argument is given, MAC filtering is globally disabled.

Syntax:

disable all

filter *filter-number*

all Disables MAC filtering globally. This command does not individually disable each filter.

filter ***filter-number***
Disables the filter that is specified by the filter number. This filter is disabled without modifying configuration records. If no filter number is given, MAC filtering is globally disabled.

Enable

Use the **enable** command to enable MAC filtering globally. This command does not individually enable each filter.

The command also enables a filter as specified by filter-number. This filter is enabled without modifying configuration records. If no argument is given, MAC filtering is globally enabled.

Syntax:

enable all

filter *filter-number*

all Enables MAC filtering globally. This command does not individually enable each filter.

filter *filter-number*
Enables the filter that is specified by the filter number. This filter is enabled without modifying configuration records. If no filter number is given, MAC filtering is globally enabled.

List

Use the **list** command to list a summary of statistics and settings for each filter currently running in the router. The following information is displayed for each filter when the **list all** command is used:

- Default action
- Cache size
- Default tag
- State (enabled/disabled)
- Number of packets which have been filtered as INCLUDE, EXCLUDE or TAG.

In addition, the following information is also displayed by the **list filter** command for a specified filter:

- All information displayed by the list all command
- All the filter-lists currently running in this filter including:
  - List name
  - List action
  - List tag
  - Number of packets which have been filtered by each filter-list.

Syntax:

list all

filter *filter-number*

all Lists statistics and settings for each filter currently running in the router.

filter *filter-number*
Generates statistics and settings for each filter plus all the filter-lists currently running in this filter.

Reinit

Use the **reinit** command to re-initialize the entire MAC Filtering system from an updated configuration, without affecting the rest of the router.

Syntax:

reinit

Configuring MAC Filtering

Chapter 5. Using WAN Restoral

This chapter includes the following sections:

Overview for WAN Restoral, WAN Reroute, and Dial-on-Overflow

The WAN Restoral, WAN Reroute, and Dial-on-overflow features have similar functions and might be confused. This overview is intended to help you decide which of these functions will be useful to you and to help you find the information you need to configure them.

The configuration commands for all three features are included in the ″Configuring WAN Restoral″ chapter. For additional information about WAN Reroute and Dial-on-overflow see "Chapter 7. The WAN Reroute Feature" on page 87.

WAN Restoral

WAN Restoral is the most basic function. When you use WAN Restoral, you configure a primary and a secondary link. In case the primary link fails, the secondary link is started and assumes the characteristics of the primary. You don't configure any protocol definitions on the secondary link because it uses the protocol definitions from the primary link.

For WAN Restoral:

- There is a pairing between a primary and a secondary link.
- You can configure only one primary to use a specific secondary link.
- You don't configure protocol definitions (for example: protocol addresses) on the secondary link.
- The primary link can be a PPP serial interface or a multilink PPP interface. It can not be a PPP dial circuit interface.
- The secondary link must be a PPP dial circuit or a multilink PPP interface.
- You must enable the WRS feature using the **enable wrs** command.
- You must enable the primary/secondary pair using the **enable secondary-circuit** command.

Note: When BRS is configured on a primary link and the primary link is part of a primary-secondary pair for WAN Restoral, you must configure BRS on the secondary link. Typically when WAN Restoral is configured, the secondary link takes the identify of the primary link. However, this is not true for BRS; therefore, BRS needs to be configured on both the primary and secondary link.

WAN Reroute

WAN Reroute is a more advanced function. When you use WAN Reroute, you configure a primary and an alternate link. In case the primary link fails, the alternate link is started. The routing protocols (for example, RIP or OSPF) detect the newly available link and adjust the routes that are used for forwarding packets.

For WAN Reroute:

- There is a pairing between a primary and an alternate link.
- You may configure multiple primary links to use the same alternate link.
- You must configure protocol definitions on the alternate link.
- The primary link may be any link on which you can configure routable protocols (e.g. IP, IPX). For example, the primary link may be a LAN interface, a PPP, Frame Relay, or X.25 serial interface, or a PPP or Frame Relay dial circuit. The following are examples of interface types that cannot be primary links: SDLC serial interfaces, SRLY serial interfaces, and base nets like V.25bis and ISDN.
- The alternate link may be any link on which you can configure routable protocols (e.g. IP, IPX) and the datalink type of the alternate link need not match the datalink type of the primary link. For example, the alternate link may be a LAN interface, a PPP, Frame Relay, or X.25 serial interface, or a PPP or Frame Relay dial circuit. The following are examples of interface types that cannot be alternate links: SDLC serial interfaces, SRLY serial interfaces, and base nets like V.25bis and ISDN.
- If the primary link is a dial circuit, then it cannot be a dial-on-demand dial circuit. To configure the dial circuit so that it is not a dial-on-demand circuit, you must configure it with **set idle 0** at that dial `Circuit Config>` prompt. Refer to "Configuring and Monitoring Dial Circuits" in the *Software User's Guide* for more information.

  I.430, I.431 and Channelized T1/E1 dial circuits are implicitly fixed, and therefore can be used as a WRS primary.

  Note: I.430/I.431 and Channelized T1/E1 dial circuits can be used as WRS primary without any explicit configuration.
- The alternate link cannot be a dial-on-demand dial circuit (you must configure **set idle 0** on the dial circuit).
- You must enable the WRS feature using the **enable wrs** command.
- You must enable the primary/alternate pair using the **enable alternate-circuit** command.
- You may optionally configure stabilization times, routing-stabilization times, and start-and stop-time-of-day-revert-back times to control the switching back to the primary link.
- If the alternate link is X.25, you should use the **national-personality set disconnect-procedure active** command when configuring the X.25 interface of the router that has WAN Reroute enabled and use the **national-personality set disconnect-procedure passive** command when configuring the X.25 interface of the other router.

Dial-on-overflow

Dial-on-overflow is similar to WAN Reroute, but does not require failure of the primary to start the alternate link. Instead, the utilization of the primary link is monitored, and if a threshold is exceeded, the alternate link is started. Also, not all

protocols are brought up on the alternate link. Only IP is brought up on the alternate link, and other protocols continue to use the primary link unless the primary link goes down.

If the primary link goes down, WAN Reroute takes over and any protocols configured on the alternate interface can start detecting and using routes on the alternate interface.

For Dial-on-overflow:

- Dial-on-overflow uses the primary/alternate pairing of a WAN Reroute pair.
- You must configure a WAN reroute pair to use Dial-on-overflow, and all the restrictions of WAN Reroute configuration apply.
- The primary link of a WAN Reroute pair that will be used for Dial-on-overflow must be Frame Relay.
- You must use the OSPF routing protocol to use Dial-on-overflow.
- You must use the **enable dial-on-overflow** command to configure add-threshold and drop-threshold, the bandwidth monitoring interval, and the minimum alternate up time.
- Stabilization times, routing-stabilization times, and start-time-of-day-revert-back and stop-time-of-day-revert-back times do not affect the operation of dial-on-overflow.

For more information about WAN Reroute see “Chapter 7. The WAN Reroute Feature” on page 87.

Before You Begin

Before you configure WAN Restoral, you must have the following:

1. A primary serial interface (leased line) configured for PPP. You can use any serial interface on the router.
2. An interface with the associated dial circuits configured on the router. You can use an ISDN interface, a V.25bis interface, or V.34 interface as the base net.
3. A secondary dial circuit configured to dial when the primary interface goes down. To configure a dial circuit to do this, set the idle timer to zero using the **set idle** command at that dial `Circuit Config>`prompt. This command prevents the dial circuit from being dial-on-demand.
4. A secondary dial circuit at one end of the link configured to send calls only. Use the **set calls outbound** command at the `Circuit Config>` prompt.

   Note: Do not configure any protocol addresses on the secondary interface. The protocol assignments for the primary interface are used on the secondary link (dial circuit) when it is active.
5. A secondary dial circuit at the other end of the link configured to receive calls only. Use the **set calls inbound** command at the `Circuit Config>` prompt.

Configuration Procedure for WAN Restoral

This section describes the steps required to configure WAN Restoral. Before you begin, use the **list device** command at the Config> prompt to list the interface numbers of different devices.

Follow these steps to configure WAN Restoral on the router:

1. Display the WRS Config> prompt by entering the **feature wrs** command at the Config> prompt. For example:

```
Config>feature wrs
WAN Restoral user configuration
WRS Config>
```

2. Assign a secondary dial circuit to the primary interface. This dial circuit will back up the primary interface. For example:

```
WRS Config>add secondary-circuit
Secondary interface number  [0]? 3
Primary interface number  [0]? 1
```

3. Enable WAN Restoral on the secondary dial circuit that you added. For example:

```
WRS Config>enable secondary-circuit
Secondary interface number  [0]? 3
```

4. Globally enable WAN Restoral on the router. For example:

```
WRS Config>enable wrs
```

5. Restart the router for configuration changes to take effect.

Secondary Dial Circuit Configuration

To configure a dial circuit:

1. Determine the dial-circuit interface number: To do this, type:

```
Config> list device
```

   If no PPP dial-circuit interface is listed, add a dial-circuit interface by typing:

```
Config> add device dial-circuit

Adding device as interface 3
Defaulting Data-link protocol to PPP
Use "net 3" command to configure circuit parameters
```

2. Configure the secondary interface (dial circuit) to have the same datalink type as the primary interface (PPP) from the Config> prompt as follows:

```
Config> set data PPP
Interface Number [0]? 3
```

3. Access the dial circuit configuration prompt (Circuit Config>) by entering **network** *interface#*.

```
Config> network 3
```

4. Select the base net interface for the dial circuit. The base net can be V.25bis, ISDN, or V.34.

```
Circuit Config> set net 2
```

5. Set the dial circuit idle timer to 0 (0=fixed) as follows:

```
Circuit Config> set idle 0
```

6. Set one end of the backup connection to receive calls (for example, router A) as follows:

```
Circuit Config> set calls inbound
```

7. Set the other end of the backup connection to initiate calls (for example, router B) as follows:

```
Circuit Config> set calls outbound
```

Notes:

1. Do not use the **set calls both** command. Setting these individually will help prevent the collisions of incoming and outgoing connection attempts.
2. Do not configure any forwarder (for example, IP, IPX, etc.) addresses on the dial circuit. The protocol assignments for the primary interface are used on the secondary interface (dial circuit) when it is active.
3. For ISDN configuration instructions, see “Using the ISDN Interface” in *Software User’s Guide*.
4. For V.25bis configuration instructions, see “Using the V.25bis Interface” in *Software User’s Guide*.
5. For V.34 configuration instructions, see “Using the V.34 Interface” in *Software User’s Guide*.

Using WAN Restoral

Chapter 6. Configuring and Monitoring WAN Restoral

This chapter describes the WAN Restoral configuration and operational commands. It includes the following sections:

Note: Refer to "Configuring and Monitoring Dial Circuits" in the *Software User's Guide* for information about configuring dial circuits. A dial circuit can be used as an interface when configuring WAN Reroute.

WAN Restoral, WAN Reroute, and Dial-on-Overflow Configuration Commands

The WAN Restoral configuration commands allow you to create or modify the WAN Restoral interface configuration. This section summarizes and explains the WAN Restoral configuration commands.

Table 8 lists the WAN Restoral configuration commands and their function. Enter these commands at the `WRS Config>` prompt. To access `WRS Config>`, enter **feature wrs** at the `Config>` prompt.

Table 8. WAN Restoral Configuration Commands Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page xxix.
Add	Adds a mapping of primary-to-secondary (for WAN Restoral) or primary-to-alternate (for WAN Reroute).
Disable	Disables WRS, an individual secondary-circuit mapping, or alternate-circuit mapping.
Enable	Enables WRS, an individual secondary-circuit mapping, or alternate-circuit mapping.
List	Displays the current Restoral configuration.
Remove	Removes a primary to secondary mapping or a primary to alternate mapping created by add.
Set	Sets the values for the stabilization, route-stabilization, and time-of-day-revert-back timers.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page xxix.

Add

Use the **add** command to identify a secondary or an alternate dial-circuit or leased link interface for a primary serial link.

Syntax:

add alternate-circuit
secondary-circuit

alternate-circuit

The **add alternate-circuit** command binds an alternate interface to a

primary interface for WAN Reroute purposes. You can assign multiple primaries to a single alternate interface. The alternate link type need not be the same as the primary link type (for example, the alternate link type can be a PPP dial circuit and the primary link type can be a Frame Relay leased line).

Example:

```
WRS Config>add alt
Alternate interface number  [0]? 6
Primary interface number  [0]? 1
```

Alternate interface number

This is the interface number previously assigned to the alternate interface. Any LAN interface, PPP, Frame Relay, or X.25 serial interface, or a PPP or Frame Relay dial circuit is an eligible alternate interface. The default is 0.

Primary interface number

This is the interface number of the primary interface previously assigned when the device was added. A primary interface can be any previously defined LAN interface, PPP, Frame Relay, or X.25 serial interface, or a PPP or Frame Relay dial circuit. The default is 0.

secondary-circuit

The **add secondary-circuit** command binds a secondary interface to a primary interface for WAN Restoral purposes. Both interfaces must have previously been configured. You can only assign one secondary interface to a primary and vice-versa.

Example:

```
WRS Config>add secondary-circuit
Secondary interface number  [0]? 4
Primary interface number  [0]? 1
```

Secondary interface number

This is the dial circuit interface number previously assigned to the secondary interface when the device was added. Any PPP dial circuit or Multilink PPP interface can be a secondary interface. The default is 0.

Primary interface number

This is the interface number of the primary interface previously assigned when the device was added. A primary interface can be any previously defined leased-line running PPP. The default is 0.

Disable

Use the **disable** command to disable the WAN Restoral function, or to disable a primary/secondary pairing for WAN Restoral, or to disable a primary/alternate pairing for WAN Reroute, or to disable Dial-on-overflow for a primary/alternate pairing.

Syntax:

disable alternate-circuit
dial-on-overflow
secondary-circuit
wrs

alternate-circuit *interface#*
Disables the primary/alternate pairing for WAN Reroute.

Example:

```
WRS Config> disable alternate-circuit
Alternate interface number  [0]? 6
```

Alternate interface number
This is the number of the alternate interface previously configured with the **add alternate-circuit** command. The default is 0.

dial-on-overflow *alt-intfc#*
Disables dial-on-overflow for all primary/alternate pairings using a specified alternate.

Example:

```
WRS Config> disable dial-on-overflow
alternate interface number  [0]? 6
```

Alternate interface number
This is the number of the alternate interface previously configured with the **add alternate-circuit** command. The default is 0.

secondary-circuit *interface#*
Disables the restoral of a particular primary interface by its associated secondary interface until the next **enable secondary-circuit** command at the WRS console. Both interfaces must have been previously configured and bound together in the WRS configuration.

Example:

```
WRS Config> disable secondary-circuit
Secondary interface number  [0]? 3
```

Secondary interface number
This is the number of the secondary interface previously configured with the **add secondary-circuit** command. The default is 0.

wrs Disables the WAN Restoral feature globally on the router. This means that WAN Reroute and Dial-on-overflow are also disabled.

Enable

Use the **enable** command to enable the WAN Restoral function, to enable a primary/secondary pairing for WAN Restoral, to enabled a primary/alternate pairing for WAN Reroute, or to enable dial-on-overflow for a primary/alternate pairing.

Syntax:

enable alternate-circuit
dial-on-overflow
secondary-circuit
wrs

alternate-circuit *interface#*
Enables an alternate circuit

Example:

```
WRS Config>enable alternate-circuit
Alternate interface number  [0]? 6
```

Alternate interface number
This is the number of the alternate interface previously configured with the **add alternate-circuit** command. The default is 0.

dial-on-overflow
Enables dial-on-overflow and allows you to set parameters that control how dial-on-overflow works.

Example:

```
WRS>enable dial-on-overflow

For dial-on-overflow, only IP traffic can overflow to the alternate
interface.
Primary interface number  ]0]? 1
add-threshold (1-100% utilization) [90]?
drop-threshold(0-99% utilization) [60]?
bandwidth test interval(10-200 seconds) [15]?
minimum time to keep the alternate up (20-21600 sec.) [300]?
Dial-on overflow is enabled.
Remember to configure the primary interface's line speed!
```

Primary interface number
This is the interface number of the primary interface for which you are enabling dial-on-overflow. The default is 0.

add-threshold
Determines when an alternate interface will be brought up for additional bandwidth. This value must expressed as a percentage of the primary interface's configured line speed. The default is 90%.

drop-threshold
Determines when an alternate interface is no longer needed for additional bandwidth. This value must expressed as a percentage of the primary interface's configured line speed. The default is 60%.

bandwidth monitoring interval
Determines how often the primary interface's bandwidth is monitored for the *add-threshold* and *drop-threshold.* The default is 15 seconds.

Minimum time to keep alternate up
This time period needs to include enough time for the routers to establish the new route when IP traffic on the local router is rerouted to the alternate interface. The default is 5 minutes.

secondary-circuit *interface#*
Enables the restoral of a primary link by the indicated secondary link.

Example:

```
WRS Config>enable secondary-circuit
Secondary interface number  [0]? 3
```

Secondary interface number
This is the number of the secondary interface previously configured with the **add secondary-circuit** command. The default is 0.

wrs Enables the function of the WAN Restoral feature on the router. This means that if WAN Reroute and Dial-on-overflow are configured they are also enabled.

List

Use the **list** command to display global configuration information for the feature and display configuration information for WAN Restoral primary-secondary pairs, WAN Reroute primary-alternate pairs, and Dial-on-Overflow.

Syntax:

list

Example:

```
WRS Config>list all
WAN Restoral is enabled.
Default Stabilization Time:       0 seconds
Default First Stabilization Time: 0 seconds

                                                  Secondary
   Primary Interface       Secondary Interface    Enabled
----------------------  ----------------------  --------
 4 - WAN PPP             7 - PPP Dial Circuit       No
                                          Alt.     1st  Subseq TOD Revert Back  Stab
Primary Interface  Alternate Interface    Enabled  Stab  Stab  Start    Stop
-----------------  ---------------------  -------  ----  ----  -------  ------- ----

 1 - WAN Frame Re   2 - WAN Frame Relay      Yes    dflt  dflt  Not Set  Not Set 15


Dial-on-overflow is enabled.
Primary    add-       drop-      test       minimum
Interface  threshold  threshold  interval   alt up time
---------  ---------  ---------  ---------  -----------
     1         29%        20%      15 sec.   300 sec.
```

Remove

Use the **remove** command to delete the mapping of an alternate interface or secondary (backup) interface to the primary interface.

Syntax:

remove alternate-circuit

secondary-circuit

alternate-circuit *alternate-interface# primary-interface#*

Removes the mapping of a alternate (backup) interface to the primary interface for WAN Reroute. Both interfaces must have been previously assigned and bound together using the **add alternate-circuit** command.

Alternate-interface#

This is the number of the alternate interface previously configured with the **add alternate-circuit** command. The default is 0.

Primary-interface#

This is the interface number of the primary interface previously bound to the alternate being removed. The default is 0.

Example:

```
WRS Config> remove alternate-circuit
Alternate interface number  [0]? 3
Primary interface number  [0]? 1
```

secondary-circuit *secondary-interface# primary-interface#*

Removes the mapping of a secondary (backup) interface to the primary interface for WAN Restoral. Both interfaces must have been previously assigned and bound together using the **add secondary-circuit** command.

Secondary-interface#

This is the number of the secondary interface previously configured with the **add secondary-circuit** command. The default is 0.

Primary-interface#

This is the interface number of the primary interface previously bound to the secondary being removed. The default is 0.

Example:

```
WRS Config> remove secondary-circuit
Secondary interface number  [0]? 3
Primary interface number  [0]? 1
```

Set

Use the **set** command to set the parameters for WAN Reroute.

Syntax:

set ?

default

first-stabilization

routing-stabilization

stabilization

start-time-of-day-revert-back

stop-time-of-day-revert-back

default

Use the **set default** command to set the defaults to be used by links that do not have configured stabilization and first-stabilization times.

first-stabilization

Sets the default first-stabilization value to be used for links for which a first-stabilization time was not configured.

```
WRS Config>set default first
Default first primary stabilization time (0 - 3600 seconds)  [0]? 20
```

stabilization

Sets the default stabilization value to be used for links for which a stabilization time was not configured.

```
WRS Config>set default stab
 Default primary stabilization time (0 - 3600 seconds)  [0]? 30
```

first-stabilization

Sets the number of seconds at router initialization before routing for this primary link is switched to the alternate link if the primary link is not up.

Example:

```
WRS Config>set first
Primary interface number  [0]? 1
First primary stabilization time (0 - 3600 seconds -1 = default)  [-1]?
```

Primary interface number

This is the primary interface number of the primary interface for which you are setting first-stabilization. The default is 0.

First primary stabilization time

The stabilization time for this primary interface. The default is 1.

routing-stabilization

Sets the routing-stabilization value. This parameter defines the number of seconds that both the primary link and the alternate link remain up after the primary link has been found to be up and the stabilization timer, if any, has expired. The routing-stabilization time is provided so that routing protocols such as OSPF or RIP have enough time to recognize the availability of the

new route. Without the routing-stabilization timer, traffic can be interrupted for several seconds while the alternate route has been disabled and the primary route has not yet been discovered.

If the alternate link was up prior to the reroute, the alternate link remains up and the routing-stabilization timer is ignored. If the alternate link went down prior to the reroute or during the reroute, the alternate link remains down and the routing-stabilization timer and the stabilization timer are both ignored.

```
WRS Config>set routing-stabilization
Primary interface number [0]? 1
Routing stabilization timer (0 - 3600 seconds) [0]?
```

Primary interface number

Valid Values: 0 to the number of interfaces configured on the router

Default Value: 0

Routing-stabilization timer

Valid Values: 1 to 3600 seconds

Default Value: 0

stabilization

Sets the number of seconds required after the primary link is first detected to be up before the process of re-initializing routing on the primary link begins. When the stabilization timer expires, the alternate link will be brought down unless the routing-stabilization timer has been configured. The routing-stabilization timer will start as soon as the stabilization timer expires and will keep both the primary and the alternate links up long enough to maintain the traffic on the alternate link while the routing protocols such as OSPF and RIP reestablish the route over the primary link.

Example:

```
WRS Config>set first
Primary interface number  [0]? 1
Primary stabilization time (0 - 3600 seconds -1 = default)  [-1]?
```

Primary interface number

This is the primary interface number of the primary interface for which you are setting stabilization. The default is 0.

Primary stabilization time

The stabilization time for the primary interface. The default is 1.

start-time-of-day-revert-back

The earliest time of the day the router can switch back to the primary route. The router can revert back to the primary any time between the start-time-of-day-revert-back time and the stop-time-of-day-revert-back time. Reverting back to the primary will only occur if the primary is up and the stabilization parameters are met. The default is 0.

Example:

```
WRS Config>set start
Primary interface number  [0]? 1
Time-of-Day revert back window start (1 - 24 hours, 0 = not configured) [0] 3
Start time-of-day revert back configured. Remember to configure stop time-of-day
```

Primary interface number

This is the primary interface number of the primary interface for which you are setting first-stabilization. The default is 0.

Time-of-day-revert-back-window start
This time marks the beginning time for the revert back window. The router can revert back to the primary interface any time between the start-time-of-day-revert-back time and the stop-time-of-day-revert-back time. Reverting back to the primary interface will only occur if the primary interface is up and the stabilization parameters are met. The default is 1.

stop-time-of-day-revert-back
This time marks the ending time for the revert back window. The router can revert back to the primary interface any time between the start-time-of-day-revert-back time and the stop-time-of-day-revert-back time. Reverting back to the primary interface will only occur if the primary interface is up and the stabilization parameters are met. The default is 1.

Example:

```
WRS Config>set stop
Primary interface number  [0]? 1
Time-of-Day revert back window stop (1 - 24 hours, 0 = not configured) [0]?5
```

Primary interface number
This is the primary interface number of the primary interface for which you are setting first-stabilization. The default is 0.

Time-of-day-revert-back-window stop
This time marks the ending time for the revert back window. The router can revert back to the primary interface any time between the start-time-of-day-revert-back time and the stop-time-of-day-revert-back time. Reverting back to the primary interface will only occur if the primary interface is up and the stabilization parameters are met. The default is 1.

Accessing the WAN Restoral Interface Monitoring Process

To access the WAN Restoral interface monitoring process, enter the following command at the GWCON (+) prompt:

```
+ feature wrs
```

WAN Restoral Monitoring Commands

The WAN Restoral (WRS) monitoring commands allow you to monitor the state of WAN Restoral primary-secondary pairs, WAN Reroute primary-alternate pairs, and Dial-on-Overflow. Any modifications to the operational state of WAN Restoral, WAN Reroute, and Dial-on-Overflow made through the monitoring interface are not maintained across router restarts.

Access the WRS prompt by entering **feature wrs** at the GWCON (+) prompt. Table 9 lists the WRS commands and their functions, and the following sections explain the commands.

Table 9. WAN Restoral Monitoring Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Clear	Clears the monitoring statistics displayed using the **list** command.

Table 9. WAN Restoral Monitoring Commands (continued)

Command	Function
Disable	Disables the WRS, or an individual secondary, or alternate, or dial-on-overflow.
Enable	Enables the WRS, or an individual secondary, or alternate, or dial-on-overflow.
List	Displays the monitoring information on one or all alternate or secondary circuits.
Set	Sets the values for the stabilization, route-stabilization, and time-of-day-revert-back-timers.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Clear

Use the **clear** command to clear WAN Restoral, WAN Reroute, and dial-on-overflow statistics that are displayed using the **list** command.

Syntax:

clear

Note: This command clears *Longest restoral period*, but does not clear the *Most recent restoral period*. For the screen display, refer to the example in the **list** command.

Disable

Use the **disable** command to disable the WAN Restoral feature completely, disable the restoral of a particular primary interface by its associated secondary interface, disable an alternate interface or disable dial-on-overflow.

Syntax:

disable alternate-circuit
dial-on-overflow
secondary-circuit
wrs

alternate-circuit

Disables a primary/alternate pairing for WAN Reroute. There can be multiple pairings using the same alternate. This command disables all the pairings using the specified alternate-circuit.

Example:

```
WRS>disable alternate-circuit
Alternate circuit number  [0]? 6
```

Alternate circuit number

This is the number of the alternate circuit. The default is 0.

dial-on-overflow

Disables dial-on-overflow for the specified primary/alternate pairing, without changing the enabled/disabled state of WAN Reroute for that pairing. If dial-on-overflow is actively routing, it is terminated at the expiration of the next monitor interval.

secondary-circuit
Disables the restoral of a particular primary interface by its associated secondary interface until the next **restart**, **reload**, or **enable secondary-circuit** command. Both interfaces must have been previously configured and bound together in the WRS configuration.

Normally, in **talk 5** (GWCON), the **disable** command causes the interface to be inactive and stay inactive. For WAN Restoral secondary, however, this is not the case. The **disable** command applied to the secondary interface does not disable the interface itself. It disables only the current call (that is, causes any active call to be disconnected.) To disable use of the secondary circuit, you need to **disable secondary-circuit** at the WAN Restoral monitoring prompt and disable the secondary interface at the top level GWCON prompt.**Example:**

```
WRS>disable secondary-circuit
Secondary interface number  [0]? 3
```

Secondary interface number
This is the number of the secondary interface previously configured with the **add secondary-circuit** command. The default is 0.

wrs Disabling WRS disables WAN Restoral, WAN Reroute, and Dial-on-overflow on the router until the next **restart**, **reload**, or **enable WRS** command.

Enable

Use the **enable** command to enable the WAN Restoral interface, enable the restoral of a primary link by a secondary circuit, enable an alternate circuit, or enable dial-on-overflow.

Syntax:

enable alternate-circuit
dial-on-overflow
secondary-circuit
wrs

alternate-circuit
Enables the primary/alternate pairings for WAN Reroute for all pairings using the specified alternate.

Example:

```
WRS> enable alternate-circuit
Alternate circuit number  [0]? 3
```

Alternate circuit number
This is the interface number of the alternate circuit. The default is 0.

dial-on-overflow
Enables dial-on-overflow and allows you to set parameters that control dial-on-overflow. Optionally, allows you to cause the IP protocol to be switched immediately to the alternate, as if the add threshold had been crossed.

Example:

```
WRS> dial-on-overflow

For dial-on-overflow, only IP traffic can overflow to the alternate interface.
Primary interface number  [0]? 1
add-threshold (1-100% utilization) [90]?
drop-threshold(0-99% utilization) [60]?
```

```
bandwidth test interval(10-200 seconds) [15]?
minimum time to keep the alternate up (20-21600 sec.) [300]?
Dial-on overflow is enabled.
Remember to configure the primary interface's line speed!

Do you want to switch IP traffic to the alternate now?(Yes or [No]):
WRS>
```

secondary-circuit
Enables the restoral of a primary link by the indicated secondary link.

Example:

```
WRS> enable secondary-circuit
Secondary interface number  [0]? 3
```

Secondary interface number
This is the number of the secondary interface previously configured with the **add secondary-circuit** command. The default is 0.

wrs Enables the function of the WAN Restoral feature on the router. This feature needs to be enabled in order to do WAN Restoral, WAN Reroute, or Dial-on-overflow.

Set

Use the **set** command to set the parameters for WAN Reroute.

Syntax:

set ?

default

first-stabilization

routing-stabilization

stabilization

start-time-of-day-revert-back

stop-time-of-day-revert-back

default
Use the **set default** command to set the defaults to be used by links that do not have configured stabilization and first-stabilization times.

Example:

```
WRS Config>set default ?
FIRST-STABILIZATION
STABILIZATION
```

first-stabilization
Sets the default first-stabilization value to be used for links for which a first-stabilization time was not configured.

```
WRS Config>set default first

Default first primary stabilization time (0 - 3600 seconds)  [0]? 20
```

stabilization
Sets the default stabilization value to be used for links for which a stabilization time was not configured.

```
WRS Config>set default stab
Default primary stabilization time (0 - 3600 seconds)  [0]? 30
```

first-stabilization
Sets the number of seconds at router initialization before routing for this primary link is switched to the alternate link if the primary link is not up.

Example:

```
WRS Config>set first
Primary interface number  [0]? 1
First primary stabilization time (0 - 3600 seconds -1 = default)  [-1]?
```

Primary interface number
This is the primary interface number of the primary interface for which you are setting first-stabilization. The default is 0.

First primary stabilization time
The stabilization time for this primary interface. The default is 1.

routing-stabilization
Sets the routing-stabilization value. This parameter defines the number of seconds that both the primary link and the alternate link remain up after the primary link has been found to be up and the stabilization timer, if any, has expired. The routing-stabilization time is provided so that routing protocols such as OSPF or RIP have enough time to recognize the availability of the new route. Without the routing-stabilization timer, traffic can be interrupted for several seconds while the alternate route has been disabled and the primary route has not yet been discovered.

If the alternate link was up prior to the reroute, the alternate link remains up and the routing-stabilization timer is ignored. If the alternate link went down prior to the reroute or during the reroute, the alternate link remains down and the routing-stabilization timer and the stabilization timer are both ignored.

```
WRS Config>set routing-stabilization
Primary interface number [0]? 1
Routing stabilization timer (0 - 3600 seconds) [15]?
```

Primary interface number

Valid Values: 0 to the number of interfaces configured on the router

Default Value: 0

Routing-stabilization timer

Valid Values: 1 to 3600 seconds

Default Value: 0

stabilization
Sets the number of seconds required after the primary link is first detected to be up before the process of re-initializing routing on the primary link begins. When the stabilization timer expires, the alternate link will be brought down unless the routing-stabilization timer has been configured. The routing-stabilization timer will start as soon as the stabilization timer expires and will keep both the primary and the alternate links up long enough to maintain the traffic on the alternate link while the routing protocols such as OSPF and RIP reestablish the route over the primary link.

Example:

```
WRS Config>set first
Primary interface number  [0]? 1
Primary stabilization time (0 - 3600 seconds -1 = default)  [-1]?
```

Primary interface number
This is the primary interface number of the primary interface for which you are setting stabilization. The default is 0.

Primary stabilization time
The stabilization time for the primary interface. The default is 1.

start-time-of-day-revert-back

Sets the earliest time of the day that the router can switch back to the primary route. The router can revert back to the primary any time between the start-time-of-day-revert-back time and the stop-time-of-day-revert-back time. Reverting back to the primary will only occur if the primary is up and the stabilization parameters are met. The default is 0.

Example:

```
WRS Config>set start
Primary interface number [0]? 1
Time-of-Day revert back window start (1 - 24 hours, 0 = not configured) [0] 3
Start time-of-day revert back configured. Remember to configure stop time-of-day
```

Primary interface number

This is the primary interface number of the primary interface for which you are setting first-stabilization. The default is 0.

Time-of-day-revert-back-window start

This time marks the beginning time for the revert back window. The router can revert back to the primary interface any time between the start-time-of-day-revert-back time and the stop-time-of-day-revert-back time. Reverting back to the primary interface will only occur if the primary interface is up and the stabilization parameters are met. The default is 1.

stop-time-of-day-revert-back

This time marks the ending time for the revert back window. The router can revert back to the primary interface any time between the start-time-of-day-revert-back time and the stop-time-of-day-revert-back time. Reverting back to the primary interface will only occur if the primary interface is up and the stabilization parameters are met. The default is 1.

Example:

```
WRS Config>set stop
Primary interface number  [0]? 1
Time-of-Day revert back window start (1 - 24 hours, 0 = not configured) [0]?
5
```

Primary interface number

This is the primary interface number of the primary interface for which you are setting first-stabilization. The default is 0.

Time-of-day-revert-back-window stop

This time marks the ending time for the revert back window. The router can revert back to the primary interface any time between the start-time-of-day-revert-back time and the stop-time-of-day-revert-back time. Reverting back to the primary interface will only occur if the primary interface is up and the stabilization parameters are met. The default is 1.

List

Use the **list** command to display monitoring information on one or all WAN Restoral primary-secondary pairs or one or all WAN Reroute primary-alternate pairs.

Syntax:

list all

alternate-circuit

secondary-circuit

summary

all Provides summary information, followed by the specific information, for each secondary interface.

Example:

```
list all
WAN Restoral/Re-route is enabled with 2 circuits configured
    Total restoral attempts =          7 completions   =          7
    Total packets forwarded =         39
    Longest completed restoral period in hrs:min:sec     0:03:27

    Total overflow attempts =         20 completions   =         19
    Longest completed overflow period in hrs:min:sec     0:05:00

   Primary       Secondary    Restoral  Restoral   Current/Longest
Net Interface  Net Interface  Enabled    Active       Duration
-------------  -------------  --------  --------   --------------------
  4  PPP/0       7  PPP/1       No         No       00:03:27/ 00.06.00

                               Re-route/ Re-route/         Recent
   Primary       Alternate    Overflow  Overflow     Reroute/Overflow
Net Interface  Net Interface  Enabled    Active          Duration
-------------  -------------  --------  --------   --------------------
  1   FR/0       2   FR/1     Yes/Yes    No /No     00:00:56/ 00:05:00
```

Total restoral attempts
The number of times the primary link failed, causing the router to try to bring up a secondary link.

Completions
The number of successful restoral attempts when the secondary link came up and was used.

Total packets forwarded
The total number of packets forwarded across the secondary interface. It is the sum of both directions, and is cumulative over all successful restores, until the restart or clear restoral-statistics command is issued.

Longest Completed Restoral Period
This field displays in hours, minutes, and seconds the longest amount of time a restoral was in operation, not counting any current usage.

Total Overflow Attempts
The number of attempts due to an overflow.

Completions
The number of successful overflow attempts when the secondary link came up and was used.

Longest Completed Overflow Period
Displays in hours, minutes , and seconds the longest amount of time an overflow was in operation, not counting any current usage.

Primary Net Interface
The interface that is being backed up by its associated secondary interface.

Secondary Net Interface
The dial circuit that is being used to back up the associated primary interface.

Restoral Enabled
Indicates that restoral of this primary interface is currently enabled.

Restoral Active
Indicates whether restoral is active (Yes or No).

Current/Longest Duration
Indicates in hours, minutes, and seconds the current and longest duration the secondary net interface was up.

Primary Net Interface
The interface that is being backed up by its associated alternate interface.

Alternate Net Interface
The interface that is being used as an alternate back up the associated primary interface.

Re-route/Overflow Enabled
Indicates whether reroute and overflow are enabled (Yes or No).

Re-route/Overflow Active
Indicates whether reroute and overflow are active (Yes or No).

Recent Re-route Overflow Duration
Indicates in hours, minutes, and seconds the recent reroute and overflow duration of the alternate net interface.

Alternate-circuit
Provides totals for an alternate circuit. Allows the monitoring operator to retrieve the WAN Reroute state and associated statistics for each alternate interface and its associated primary mapping.

Example:

```
WRS>li alt 7
Primary    1:FR/0 Frame Relay SCC Serial Line
Alternate  7:PPP/1 Point to Point V.25bis Dial Circuit
reroute Enabled, currently inactive
overflow Enabled, currently inactive
Primary first stabilization time: default (0 seconds)
Primary stabilization time: default (0 seconds)
Routing-stabilization time: 15 seconds
Time-of-day revert back not configured: start = 0, stop = 0
Restored 0 times (0 attempts)
Overflow 0 times (0 attempts)
```

Primary Interface
The interface that is being backed up by this associated alternate interface.

Alternate Interface
The dial circuit that is being used to back up the associated primary interface.

Reroute Enabled
Indicates whether reroute of this primary interface is currently enabled.

Overflow Enabled
Indicates whether overflow of this primary interface is currently enabled.

Primary first stabilization
The number of seconds at router initialization before routing for this primary link is switched to the alternate link if the primary link is not up.

First stabilization
The number of seconds required after the primary link is first detected to be up before routing is switched back from the alternate link to the primary link. Routing over the alternate link continues until the primary link remains up for this number of seconds.

Routing stabilization
The number of seconds required after routing is switched back to the primary link before the alternate link is taken down. During this time both the primary and the alternate links remain up. This interval is provided to allow routing protocols such as OSPF and RIP time to recognize the availability of the route over the primary interface.

Time-of-day revert back
The time of the day the router can switch back to the primary route. The router can revert back to the primary any time between the start-time-of-day-revert-back time and the stop-time-of-day-revert-back time. Reverting back to the primary will only occur if the primary is up and the stabilization parameters are met. The default is 0.

Restored times
The number of attempts to reroute the primary interface.

Overflow times
The number of dial-on-overflow attempts.

secondary-circuit
Provides totals for each secondary circuit. Allows the monitoring operator to retrieve the WAN Restoral state and associated statistics for each secondary interface and its associated primary mapping.

Example:

```
list secondary-circuit
Secondary interface number  [0]? 1

                                                    Secondary
Primary Interface          Secondary Interface      Enabled
-------------------        -------------------      ---------
1  PPP/0  Point to Poi     3  PPP/1  Point to Poi     Yes

Router primary interface state = Up
Router secondary interface state = Available
Restoral Statistics:

  Primary restoral attempts =       6   completions =    5
  Restoral packets forwarded =    346
    Most recent restoral period in hrs:min:sec          00:08:20
```

Primary Interface
The interface that is being backed up by this associated secondary interface.

Secondary Interface
The dial circuit that is being used to back up the associated primary interface.

Secondary Enabled
Indicates whether restoral of this primary interface is currently enabled.

Router Primary Interface State
Indicates that the primary interface state is one of the following:

Up - Indicates that the link is up.

Down - Indicates that the link is down.

Disabled - Indicates that the operator has disabled the link.

Not present - Indicates that the link is configured but there is a hardware problem.

Router Secondary Interface State
Indicates that the associated secondary interface state is one of the following:

Up - Indicates that the link is up.

Down - Indicates that the link is down. This also occurs when the base network for the secondary is disabled either at the `Config>` prompt or at the operator console.

Available - Indicates that the link is in the waiting mode.

Testing - Indicates that the link is in the process of establishing a connection.

Restoral Statistics:

Primary Restoral Attempts
The number of times the primary failed, causing the router to try to bring up a secondary link.

Restoral Packets forwarded
This field indicates the total number of packets forwarded.

Most Recent Restoral Period
This indicates how long the secondary was up, the last time it was used or during the current restoral use.

summary
Provides totals for each secondary circuit.

Example:

```
list summary
WAN Restoral is enabled with 3 circuit(s) configured

  Total restoral attempts =      3 completions =      2
  Total packets forwarded =    346
  Longest restoral period in hrs:min:sec   00:08:20

Primary Interface and State      Secondary Interface and State
---------------------------   ----------------------------------
 1  PPP/0  - Up                 3  PPP/1  - Available
```

Total restoral attempts
The number of times the primary failed, causing the router to try to bring up a secondary link.

Completions
The number of successful restoral attempts when the secondary came up and was used.

Total packets forwarded
The total number of packets forwarded across the secondary interface. It is the sum of both directions, and is cumulative over all restoral periods until the restart or clear restoral-statistics command is used.

Longest restoral period
This field displays in hours, minutes, seconds the longest amount of time restoral was in use, not counting the current usage.

Primary Interface and State
The interface that is being backed up by its associated secondary. Valid states are:

Up - Indicates that the link is up.

Down - Indicates that the link is down.

Disabled - Indicates that the operator has disabled the link.

Not present - Indicates that the link is configured but there is a hardware problem.

Secondary Interface and State

The dial circuit that is being used to back up the associated primary. Valid states are:

Up - Indicates that the link is up.

Down - Indicates that the link is down. This also occurs when the base network for the secondary is disabled either at the `Config>` prompt or at the operator console.

Testing - Indicates that the link is in the process of establishing a connection.

Available - Indicates that the link is in the waiting mode.

Chapter 7. The WAN Reroute Feature

This chapter describes the WAN reroute feature. It includes the following sections:

- "WAN Reroute Overview"
- "Configuring WAN Reroute" on page 89

Important

For the 1Sx and 1Ux models, WAN Reroute is available only if the router has both a WAN port and an ISDN B-channel active.

WAN Reroute Overview

WAN Reroute lets you set up an alternate route so that if a primary link fails, the router automatically initiates a new connection to the destination through the alternate route. See "Overview for WAN Restoral, WAN Reroute, and Dial-on-Overflow" on page 63 for an explanation of WAN Restoral, and how WAN Reroute and Dial-on-overflow work together.

The WAN Reroute process involves:

1. Detecting the primary link failure
2. Switching to the alternate link
3. Detecting the primary link recovery
4. Switching back to the primary link

The alternate link can be any link on which you can configure routable protocols (for example, IP, IPX) and the datalink type of the alternate link need not match the datalink type of the primary link. For example, the alternate link can be a LAN interface, a PPP, Frame Relay, or X.25 serial interface, or a PPP or Frame Relay dial circuit. The following are examples of interface types that cannot be alternate links: SDLC serial interfaces, SRLY serial interfaces, and base nets like V.25bis and ISDN.

Note: If the primary link or alternate link is a dial circuit, that dial circuit cannot be configured for dial-on-demand. Use the **set idle 0** command at the `Circuit Config>` prompt to configure the dial circuit so that it cannot perform dial-on-demand. Refer to "Configuring and Monitoring Dial Circuits" in the *Software User's Guide* for more information.

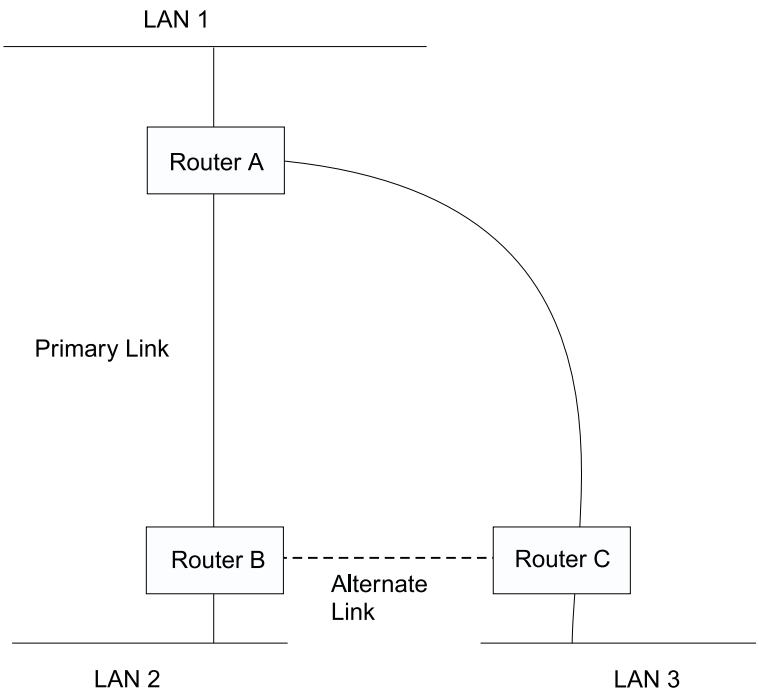


Figure 3. WAN Reroute. Normally, there is a connection between Routers A and B and Routers A and C. If the primary link between routers A and B fails, WAN reroute establishes an alternate link between routers B and C. Routers A and B can then communicate through router C.

Dial-on-Overflow

Dial-on-overflow allows you to use an alternate interface for IP traffic when the traffic rate on the primary link reaches a specified threshold. This means that the primary interface does not have to be down before the alternate link is brought up. When the primary interface's traffic reaches the specified threshold the router brings up the alternate link. To use dial-on-overflow, WAN Reroute must be configured and the primary interface must be Frame Relay. IP is the only protocol that can be switched over to the alternate interface by dial-on-overflow. Also, OSPF should be used as the IP routing protocol instead of RIP when dial-on-overflow is used.

For information about configuring dial-on-overflow, see "WAN Restoral, WAN Reroute, and Dial-on-Overflow Configuration Commands" on page 69.

Bandwidth Monitoring

The interval for bandwidth monitoring can be specified for dial-on-overflow during WAN Reroute configuration. The primary interface's receive and transmit bandwidth utilization are monitored. When the primary interface's bandwidth reaches the *add* threshold, a WAN Reroute request is generated to bring up the alternate interface. If WAN Reroute is successful bringing up the alternate interface, IP stops routing over the primary interface and starts routing over the alternate interface.

If WAN Reroute is not successful in bringing up the alternate route it periodically attempts to bring up the alternate interface until the primary interface's bandwidth utilization drops below the *drop* threshold.

When the primary interface's receive and transmit bandwidth utilization reaches the *drop* threshold and the minimum configured up time has expired the alternate interface is dropped. This causes IP to stop routing over the alternate interface and start using the primary interface.

The add-threshold and the drop-threshold are specified as a percentage of the configured line speed for the primary link. The configured line speed does not always match the actual speed of the link. The amount of traffic on the link in each direction is calculated separately. The threshold is exceeded if the traffic in either direction is greater then the specified percentage.

Configuring WAN Reroute

Following are the steps required to configure WAN reroute. The next section shows an example of how to perform these tasks.

To configure WAN Reroute, you need to:

1. Configure the primary link.
2. Configure the alternate link.
3. Assign the alternate link to the primary link. You can also specify a stabilization period for the primary link.

   You can specify a time-of-day revert-back to the primary link which will happen after the stabilization period is over (if configured). This allows the secondary to stay up until such time that the user desires and revert back to the primary during off-peak hours.

   Note: The primary and alternate links can be different datalink types. The primary and alternate links can be:
   - A LAN interface.
   - A PPP serial interface.
   - A Frame Relay serial interface.
   - An X.25 serial interface.
   - A PPP dial circuit.
   - A Frame Relay dial circuit.

Sample WAN Reroute Configuration

Figure 4 on page 90 shows WAN reroute using a Frame Relay dial circuit over ISDN as the alternate link. If the Frame Relay DLCI between router A and router C fails, WAN reroute uses the dial circuit to establish an alternate connection through router D. If one of the primary links from a branch to headquarters fails, WAN reroute establishes an alternate route to headquarters through another branch.

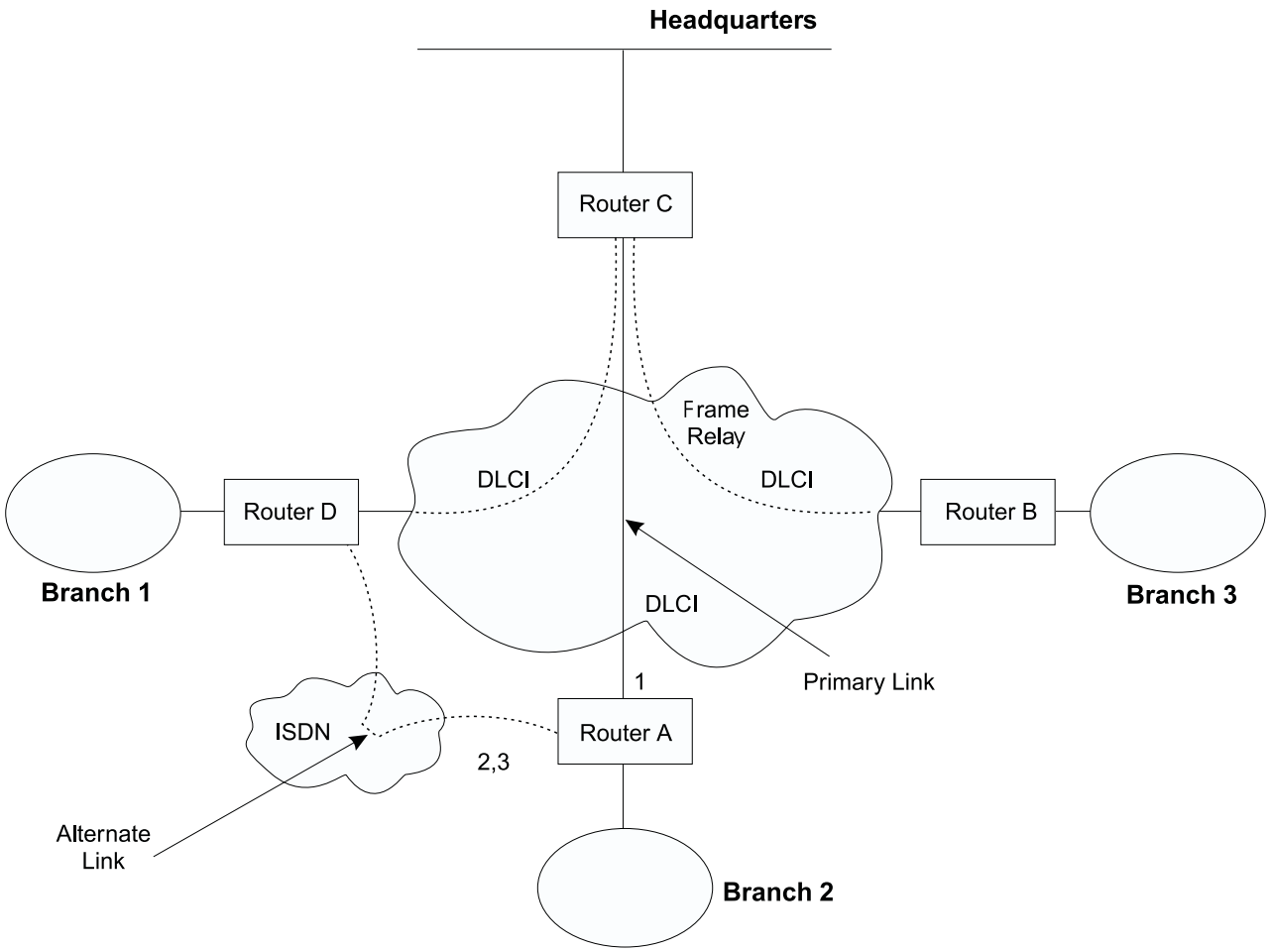


Figure 4. Sample WAN Reroute Configuration. Branch offices use frame relay to connect to headquarters.

The following sections describe how to set up WAN reroute on Router A in Figure 4. You will need to:

- Configure the primary frame relay interface (1) to have a Required PVC or Required PVC Group or enable the No-PVC feature on the frame relay interface.
- Configure the ISDN interface (2) and its frame relay dial circuit (3).
- Assign the dial circuit to be the alternate link for the primary frame relay interface and issue the **set idle 0** command at the dial `Circuit Config>` prompt to disable dial-on-demand for this circuit.
- Optionally, you can assign:
  - Stabilization period for the primary link,
  - Time-of-day revert-back window for the primary link.

These tasks are described in detail below.

Configuring the Frame Relay Interface

To configure the frame relay interface for WAN reroute, on Router A, add a PVC between Routers A and C on the primary Frame Relay interface.

To cause the primary FR interface to declare itself down when the connection to other router(s) is lost, you have three options:

1. Enable the No-PVC feature. When this feature is enabled, the FR interface goes down when there are no active PVCs.
2. Configure a PVC as required but don't include the PVC in a required PVC group. In this case, the FR interface goes down when the PVC becomes inactive.
3. Configure a set of PVCs as required and as part of a required PVC group. In this case, the FR interface goes down when all of the PVCs of a required PVC group become inactive.

Follow these steps to configure the primary Frame Relay interface:

1. If you have not yet done so, set the data link on the ISDN interface to Frame Relay.

```
Config>set data-link frame relay
Interface Number [0]? 2
```

2. Enter the Frame Relay configuration process.

```
Config>network
What is the network number [0]?2
Frame Relay user configuration
FR Config>
```

Note: Complete only *one* of the two remaining steps for configuring the primary frame relay interface.

3. Add a PVC using the **add permanent-virtual-circuit** command.

   To configure the PVC as Required:

   Enter **y** to the question "Is circuit required for interface operation ?".

   To configure the PVC as a member of a required PVC group:

   a. Enter **y** to the question "Does circuit belong to a Required PVC group ?".
   b. Enter a group name in response to the question "What is the group name ?".

   If you have already added PVCs, use the **change permanent-virtual-circuit** command to configure the PVC as Required and to assign it to a Required PVC Group, as appropriate. Refer to Using Frame Relay Interfaces in *Software User's Guide* for more information.

```
FR Config>add permanent-virtual-circuit
Circuit number [16]?
Committed Information Rate (CIR) in bps [64000]?
Committed Burst Size (Bc) in bits [64000]?
Excess Burst Size (Be) in bits [0]?
Assign circuit name []?
Is circuit required for interface operation [N]?y
Does the circuit belong to a required PVC group [N]? y
What is the group name []?group1
```

4. If desired, enable the No-PVC feature.

   Note: Complete this step *only* if you bypassed the previous step.

```
FR Config>enable no-pvc
```

There are additional parameters that you can set for frame relay. For more information, see 'Using Frame Relay' in *Software User's Guide*.

Configuring the ISDN Interface and Dial Circuit

Configure the ISDN interface and dial circuit between Router A and Router D. See 'Using the ISDN Interface' in *Software User's Guide* for information on how to configure ISDN interfaces and dial circuits.

Unlike WAN Restoral, you must configure routable protocols on the dial circuit that will be used as the alternate link. If those routable protocols cannot be prevented from sending maintenance packets, the alternate link will establish a connection even if rerouting is not necessary. In this case if you want to use the alternate link only for rerouting, disable the dial circuit. To disable the dial circuit, enter the **disable interface** command at the `Config>` prompt.

If you have multiple dial circuits assigned to the ISDN interface, you can set a priority for the dial circuits. If all the B channels have active dial circuits on the physical interface and a circuit with a higher priority receives a packet, the lowest priority connection is terminated and the high priority circuit establishes a connection.

You can set the priority to between 0 and 15, where 15 is the highest priority circuit and 0 is the lowest priority circuit. The default priority for new dial circuits is 8. Enter **set priority** at the `Circuit Config>` prompt to change the priority.

Assigning and Configuring the Alternate Link

Enter the WAN reroute configuration process to assign the dial circuit as the alternate link for a LAN interface, a PPP, Frame Relay, or X.25 serial interface, or a PPP or Frame Relay dial circuit, and if desired, to specify the stabilization periods and/or the time-of-day revert-back window.

There are three types of stabilization periods:

- *First stabilization period* is the amount of time the router waits for the primary interface to become active when the router first attempts to bring it up. If, after the first stabilization period, the primary has not come up, WAN reroute brings up the alternate link.
- *Stabilization period* is the amount of time the router waits to be sure the primary link is reliable before it switches from the alternate link back to the primary link.
- *Routing stabilization period* is the amount of time that the router keeps both the primary link and the alternate link active after it switches from the alternate link back to the primary link. This time is used by routing protocols such as OSPF or RIP to recognize the availability of the new route over the primary link before the alternate link goes down.

The time-of-day revert-back window is the specific time of day when the user desires the switch back to the primary after it is up and any configured stability time has passed.

Using a 24-hour clock, the user specifies the start and stop hours of the revert back window. The secondary stays up and is not taken down until the start hour is reached. If the time of day when the primary comes up is between the start and stop hours (in the window) then the switch to the primary link is immediate after the stability time is up.

Follow these steps to assign and configure the alternate link:

1. Enter the WAN Restoral configuration process.

```
Config>feature wrs
WAN Restoral user configuration
```

2. Assign the dial circuit as the alternate link for the primary frame relay interface.

```
WRS Config>add alternate-circuit
Alternate interface number  [0]? 4
Primary interface number  [0]? 1
```

3. Enable the alternate circuit.

```
WRS Config>enable alternate-circuit
Alternate interface number  [0]? 4
```

4. Optionally, specify a first stabilization period.

   To set the first stabilization period for a specific primary interface, use the **set first-stabilization-period** command. To set a default first stabilization period for all interfaces that do not have specific periods set, use the **set default first-stabilization-period** command.

```
WRS Config>set first-stabilization-period
Primary interface number [0]?
First primary stabilization time (0 - 3600 seconds -1=default) [-1]?

WRS Config>set default first-stabilization-period
Default first primary stabilization time (0 - 3600 seconds) [0]?
```

5. Optionally, specify a stabilization period. To set a stabilization period for specific interfaces use the **set stabilization-period** command. To set a default stabilization period for all interfaces that do not have specific periods set, use the **set default stabilization-period** command.

```
WRS Config>set stabilization-period
Primary interface number [0]?
First primary stabilization time (0 - 3600 seconds -1=default) [-1]?
WRS Config>set default stabilization-period
Default first primary stabilization time (0 - 3600 seconds) [0]?
```

6. Optionally, specify a routing stabilization period. To set a routing stabilization period for specific interfaces use the **set routing-stabilization** command.

```
WRS Config>set routing-stabilization
Primary interface number [0]? 1
Routing stabilization time (0 - 3600 seconds) [15]?
```

7. Optionally, specify a time-of-day-revert-back window.

   To set the start and stop times for specific interface windows use the set start-time-of-day-revert-back and stop-time-of-day-revert-back commands. The default value of zero means no window is configured. The 24-hour clock starts at 1 a.m. and ends at 24 midnight. If the start and stop times are the same (but not zero) then the revert back will happen at exactly that hour.

   Following are two examples of setting the revert-back window:

   a. A start time of 23 and a stop time of 3 will give a revert-back window from 11 p.m. until 3 a.m.

   b. A start time of 1 and a stop time of 5 will give a revert-back window from 1 a.m. to 5 a.m.

```
WRS Config> set start-time-of-day-revert-back
Primary interface number [0]?
Time-of-Day revert back window start (1 - 24 hours, 0 = not configured) [0]?
WRS Config> set stop-time-of-day-revert-back
Primary interface number [0]?
Time-of-Day revert back window stop (1 - 24 hours, 0 = not configured) [0]?
```

Configuring WAN Reroute

Chapter 8. Using the Network Dispatcher Feature

This chapter describes how to use the Network Dispatcher Feature and contains the following sections:

Network Dispatcher uses load-balancing technology from IBM Research Division to determine the most appropriate server to receive each new connection. This is the same technology used in IBM's eNetwork Dispatcher product for Solaris, Windows NT and AIX.

Overview of Network Dispatcher

Network Dispatcher is a feature that boosts the performance of servers by forwarding TCP/IP session requests to different servers within a group of servers, thus load balancing the requests among all servers. The forwarding is transparent to the users and to applications. Network Dispatcher is useful for server applications such as e-mail, World Wide Web servers, distributed parallel database queries, and other TCP/IP applications.

Network Dispatcher can also be used for load balancing stateless UDP application traffic to a group of servers.

Network Dispatcher can help maximize the potential of your site by providing a powerful, flexible, and scalable solution to peak-demand problems. During peak demand periods, Network Dispatcher can automatically find the optimal server to handle incoming requests.

The Network Dispatcher function does not use a domain name server for load balancing. It balances traffic among your servers through a unique combination of load balancing and management software. Network Dispatcher can also detect a failed server and forward traffic to other available servers.

All client requests sent to the Network Dispatcher machine are forwarded to the server that is selected by the Network Dispatcher as the optimal server according to certain dynamically set weights. You can use the default values for those weights or change the values during the configuration process.

The server sends a response back to the client without any involvement of Network Dispatcher. No additional software is required on your servers to communicate with Network Dispatcher.

The Network Dispatcher function is the key to stable, efficient management of a large, scalable network of servers. With Network Dispatcher, you can link many individual servers into what appears to be a single, virtual server. Your site thus

appears as a single IP address to the world. Network Dispatcher functions independently of a domain name server; all requests are sent to the IP address of the Network Dispatcher machine.

Network Dispatcher allows a management application that is SNMP-based to monitor Network Dispatcher status by receiving basic statistics and potential alert situations. Refer to “SNMP Management” in the *Protocol Configuration and Monitoring Reference Volume 1* for more information.

Network Dispatcher brings distinct advantages in load balancing traffic to clustered servers, resulting in stable and efficient management of your site.

Balancing TCP and UDP Traffic Using Network Dispatcher

There are many different approaches to load balancing. Some of these approaches allow users to choose a different server at random if the first server is slow or not responding. Another approach is round-robin, in which the domain name server selects a server to handle requests. This approach is better, but does not take into consideration the current load on the target server or even whether the target server is available.

Network Dispatcher can load balance requests to different servers based on the type of request, an analysis of the load on servers, or a configurable set of weights that you assign. To manage each different type of balancing, the Network Dispatcher has the following components:

Executor

Load balances connections based on the type of request received. Typical request types are HTTP, FTP, and Telnet. This component always runs.

Advisors

Queries the servers and analyzes the results by protocol for each server. The advisor passes this information to the ***manager*** to set the appropriate weight. The advisor is an optional component.

Network Dispatcher supports advisors for FTP, HTTP, SMTP, NNTP, POP3, and Telnet as well as a TN3270 advisor that works with TN3270 servers in IBM 2210s, IBM 2212s, and IBM 2216s and an MVS advisor that works with Workload Manager (WLM) on MVS systems. WLM manages the amount of workload on an individual MVS ID. Network Dispatcher can use WLM to help load balance requests to MVS servers running OS/390 V1R3 or later release.

There are no protocol advisors specifically for UDP protocols. If you have MVS servers, you can use the MVS system advisor to provide server load information. Also, if the port is handling TCP and UDP traffic, the appropriate TCP protocol advisor can be used to provide advisor input for the port. Network Dispatcher will use this input in load balancing both TCP and UDP traffic on that port.

Manager

Sets weights for a server based on:

- Internal counters in the executor
- Feedback from the servers provided by the protocol advisors
- Feedback from a system monitor (MVS advisor).

The manager is an optional component. However, if you do not use the manager, the Network Dispatcher will balance the load using a round-robin scheduling method based on the current server weights.

When using Network Dispatcher to load balance stateless UDP traffic, you must only use servers that respond to the client using the destination IP address from the request. See “Configuring a Server for Network Dispatcher” on page 103 for a more complete explanation.

High Availability for Network Dispatcher

The base Network Dispatcher function has the following characteristics that makes it a single point of failure from many different perspectives:

- It examines all the traffic on the way in. If some of the packets for an existing connection use a different path through a different Network Dispatcher to reach a server, the server immediately resets the connection.
- It keeps track of all established connections and although it does not terminate them, entries lost from the Network Dispatcher connection table will result in the resetting of a connection.
- It appears to any previous hop router as the last hop, and the connection’s termination.

All these characteristics make the following failures critical for the whole cluster:

- If the Network Dispatcher fails for any reason, all the connection tables are lost, therefore all existing connections from the client to the server are also lost. Assuming there is a second Network Dispatcher that can direct a client to the servers, new connections will be able to go through only after the usual routing protocol delays which could be several minutes.
- If the configured Network Dispatcher interface to the previous IP router fails, there must either be another interface to get to the same Network Dispatcher, in which case recovery is performed by the IP router (using the ARP aging mechanism with delays in the order of several minutes), or all connections will be lost.
- If Network Dispatcher interface to the servers fails, the previous hop router assumes that the Network Dispatcher is the last hop, and therefore will not reroute new connections. Existing connections will be lost and new connections will not be established.

In all these failure cases, which are not only Network Dispatcher failures but also Network Dispatcher neighborhood failures, all the existing connections are lost. Even with a backup Network Dispatcher running standard IP recovery mechanisms, recovery is, at best, slow and applies only to new connections. In the worst case, there is no recovery of the connections.

To improve Network Dispatcher availability, the Network Dispatcher High Availability function uses the following mechanisms:

- Two Network Dispatchers with connectivity to the same clients, and the same cluster of servers, as well as connectivity between the Network Dispatchers.
- A “Heartbeat” mechanism between the two Network Dispatchers to detect Network Dispatcher failure.
- A reachability criteria, to identify which IP host can and cannot be reached from each Network Dispatcher.

- Synchronization of the Network Dispatcher databases (that is, the connection tables, reachability tables, and other databases).
- Logic to elect the active Network Dispatcher, which is in charge of a given cluster of servers, and the standby Network Dispatcher, which continuously gets synchronized for that cluster of servers.
- A mechanism to perform fast IP takeover, when the logic or an operator decides to switch active and standby.

Failure Detection

Besides the basic criteria of failure detection, (the loss of connectivity between active and standby Network Dispatchers, detected through the Heartbeat messages) there is another failure detection mechanism named “reachability criteria.” When you configure the Network Dispatcher, you provide a list of hosts that each of the Network Dispatchers should be able to reach to work correctly. The hosts could be routers, IP servers or other types of hosts. Host reachability is obtained by pinging the host.

Switchover takes place either if the Heartbeat messages cannot go through, or if the reachability criteria are no longer met by the active Network Dispatcher and the standby Network Dispatcher is reachable. To make the decision based on all available information, the active Network Dispatcher regularly sends the standby Network Dispatcher its reachability capabilities. The standby Network Dispatcher then compares the capabilities with its own and decides whether to switch.

Database Synchronization

The primary and backup Network Dispatchers keep their databases synchronized through the ″Heartbeat″ mechanism. The Network Dispatcher database includes connection tables, reachability tables and other information. The Network Dispatcher High Availability function uses a database synchronization protocol that insures that both Network Dispatchers contain the same connection table entries. This synchronization takes into account a known error margin for transmission delays. The protocol performs an initial synchronization of databases and then maintains database synchronization through periodic updates.

Recovery Strategy

In the case of a Network Dispatcher failure, the IP takeover mechanism will promptly direct all traffic toward the standby Network Dispatcher. The Database Synchronization mechanism insures that the standby has the same entries as the active Network Dispatcher. When the failure occurs in the network (any intermediate piece of hardware or software between the client and the back-end server), and there is an alternate path through the standby Network Dispatcher that works, the switchover is performed across the alternate path.

IP Takeover

Note: Cluster IP Addresses are assumed to be on the same logical subnet as the previous hop router (IP router).

The IP Router will resolve the cluster address through the ARP protocol. To perform the IP takeover, the Network Dispatcher (standby becoming active) will issue an ARP request to itself, that is broadcasted to all directly attached networks belonging

to the logical subnet of the cluster. The previous hops' IP router will update their ARP tables (according to RFC826) to send all traffic for that cluster to the new active (previously standby) Network Dispatcher.

Configuring Network Dispatcher

There are many ways that you can configure Network Dispatcher to support your site. If you have only one host name for your site to which all of your customers will connect, you can define a single cluster and any ports to which you want to receive connections. This configuration is shown in Figure 5.

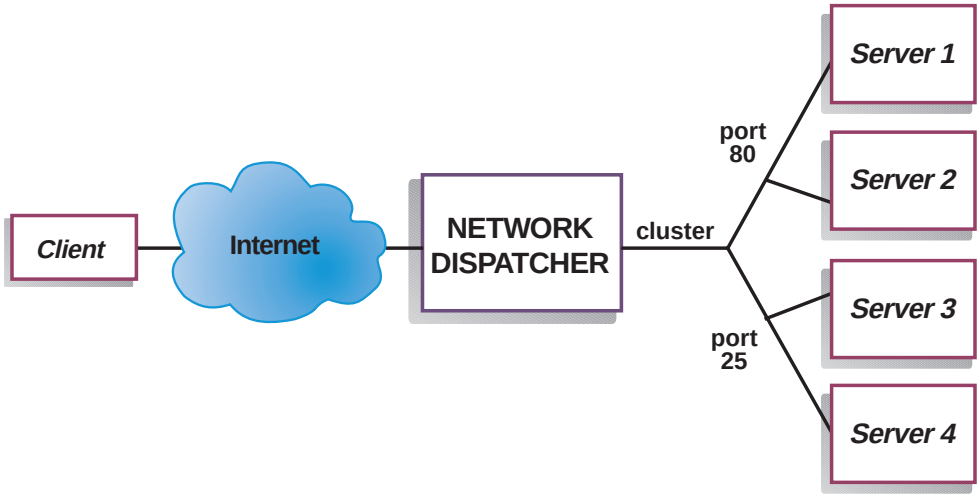


Figure 5. Example of Network Dispatcher Configured With a Single Cluster and 2 Ports

Another way of configuring Network Dispatcher would be necessary if your site does content hosting for several companies or departments, each one coming into your site with a different URL. In this case, you might want to define a cluster for each company or department and any ports to which you want to receive connections at that URL as shown in Figure 6 on page 100.

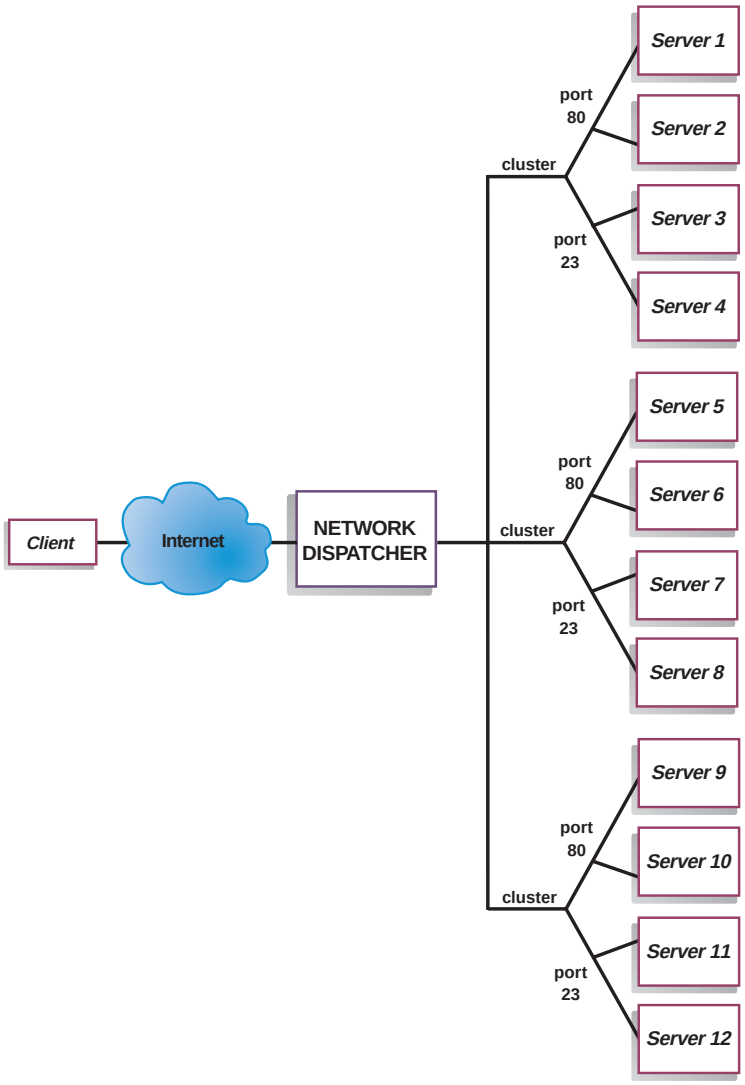


Figure 6. Example of Network Dispatcher Configured With 3 Clusters and 3 URLs

A third way of configuring Network Dispatcher would be appropriate if you have a very large site with many servers dedicated to each protocol supported. For example, you may choose to have separate FTP servers with direct T3 lines for large downloadable files. In this case, you might want to define a cluster for each protocol with a single port but many servers as shown in Figure 7 on page 101.

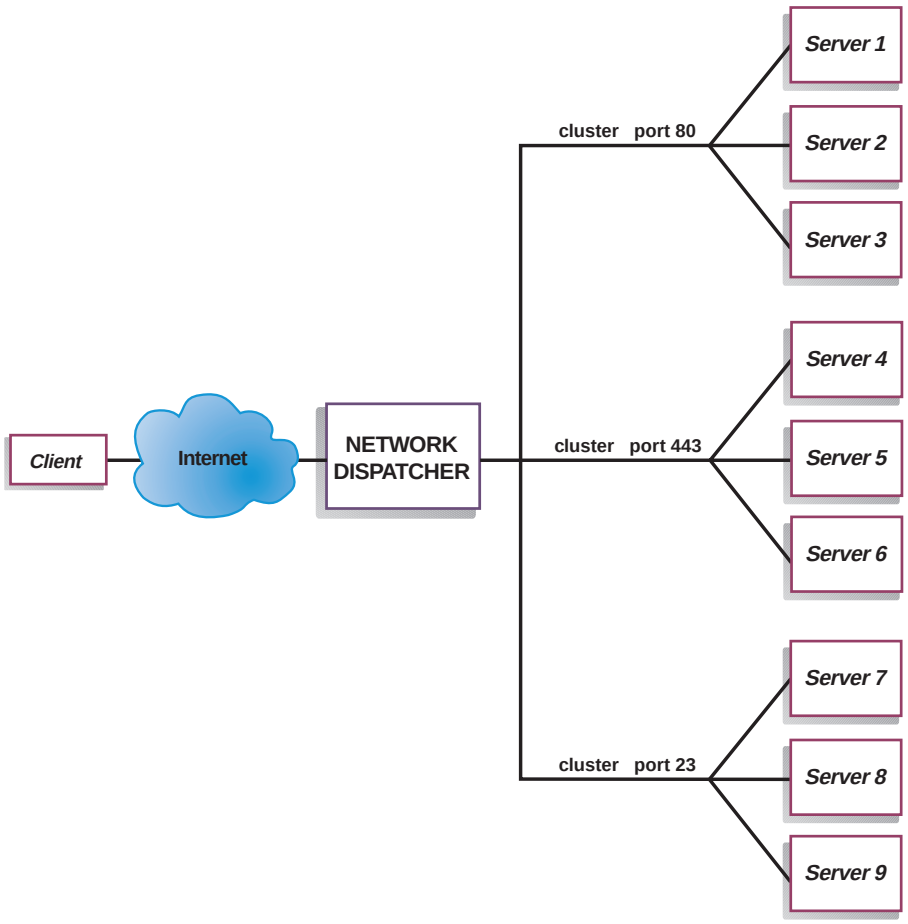


Figure 7. Example of Network Dispatcher Configured with 3 Clusters and 3 Ports

Configuration Steps

Before configuring Network Dispatcher:

1. Make sure that the Network Dispatcher has direct interfaces to servers. Servers can have independent connections to the enterprise router or Internet, such that the outgoing traffic from servers to clients can bypass the Network Dispatcher; however, you do not have to configure the independent connection.

   If high availability is important for your network, a typical high availability configuration is shown in Figure 8 on page 102.

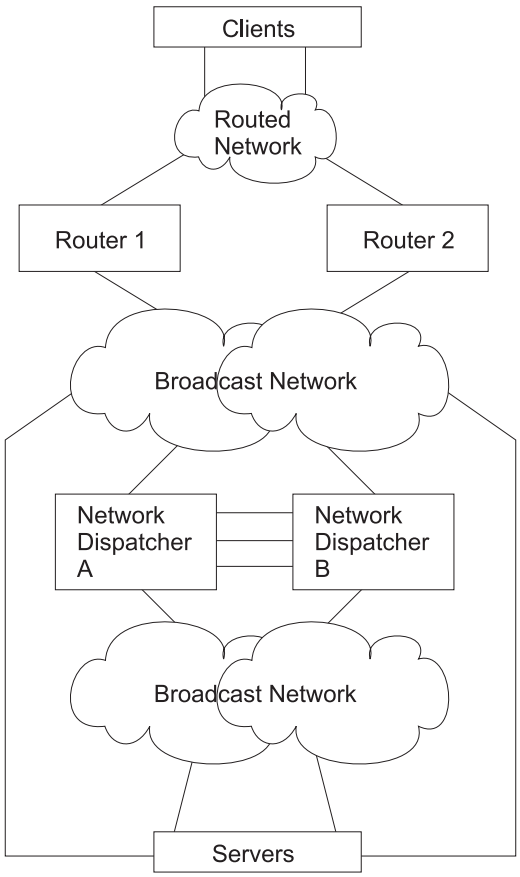


Figure 8. High Availability Network Dispatcher Configuration

2. Configure the interfaces of the device. This includes configuring all interfaces, IP addresses on all interfaces, and any applicable routing protocols. The internal IP address of the router is used by Network Dispatcher, so it must also be configured using the set internal-ip-address command. The internal IP address must not match a cluster address configured in Network Dispatcher. See the chapter Configuring and Monitoring IP in *Protocol Configuration and Monitoring Reference Volume 1* for more information about the **set internal-ip-address** command.
3. Reboot or restart the device.

Configuring Network Dispatcher on a IBM 2210

To configure Network Dispatcher on a IBM 2210:

1. Access the Network Dispatcher feature, using the **feature ndr** command.
2. Enable the executor and the manager using the **enable executor** and **enable manager** commands.
3. Configure the clusters using the add cluster command. Network Dispatcher does not specifically advertise cluster addresses, so cluster addresses should be selected that are part of an advertised subnet that is local to the Network Dispatcher router. This would typically be the subnet on which Network Dispatcher receives client traffic from the next hop router.

   Note: Cluster IP Addresses must not match the internal IP address of the router and must not match any interface IP addresses defined on the router.

4. Configure the TCP and UDP destination ports using the **add port** command for each cluster of servers that will serve the corresponding protocol. Examples of the ports are: 80 for HTTP, 20 and 21 for FTP, and 23 for Telnet.
5. Configure the servers using the **add server** commands. A server is always associated with a port and a cluster. A server can serve more than one port, a port can be served on more than one server, and a server can belong to more than one cluster, if the server's operating system supports multiple aliasing.
6. Configure any advisors using the **add advisor** command.

   Notes:

   a. For the MVS advisor, do not define the Port Number value (default = 10007) under any cluster. This port number is used only by the MVS advisor to communicate with WLM in the MVS systems.

   b. For the TN3270 advisor, two port values are entered. The Port Number value used for client-server communication (default = 23) must be defined under the appropriate clusters. Do not define the Communication Port value (default = 10008) under any cluster. The Communication Port value is used only by the TN3270 advisor to collect load information from the TN3270 servers.
7. Enable the advisors that you configured using the **enable advisor** command.

If you are configuring the Network Dispatcher for high availability, continue with the following steps. Otherwise, you have completed the configuration.

Note: Perform these steps on the primary Network Dispatcher and then on the backup. To ensure proper database synchronization, the executor in the primary Network Dispatcher must be enabled before the executor in the backup.

8. Configure whether this Network Dispatcher is a primary or backup and whether the switchover is manual or automatic using the **add backup** command.
9. Configure all paths on which the heartbeat is going to take place between the primary and backup Network Dispatchers using the **add heartbeat** command. A path is specified by source and destination IP addresses. Configuring more than one heartbeat path between the primary and backup Network Dispatchers is highly recommended to insure the failure of a single interface will not disrupt the heartbeat communication between the primary and backup machines.
10. Configure the list of host IP addresses that the Network Dispatcher must be able to reach in order to insure a full service, using the **add reach** command. Typically, this will be a subset of servers, the enterprise router, or an administration station.

You can change the configuration using the **set**, **remove**, and **disable** commands. See "Chapter 9. Configuring and Monitoring the Network Dispatcher Feature" on page 109 for more information about these commands.

Configuring a Server for Network Dispatcher

To configure a server for use with Network Dispatcher:

1. Alias the loopback device.

   For the TCP and UDP servers to work, you must set (or preferably alias) the loopback device (usually called ***lo0***) to the cluster address. Network Dispatcher does not change the destination IP address in the IP packet before forwarding

the packet to a server machine. When you set or alias the loopback device to the cluster address, the server machine will accept a packet that was addressed to the cluster address.

It is important that the server use the cluster address rather than its own IP address to respond to the client. This is not a concern with TCP servers, but some UDP servers use their own IP address when they respond to requests that were sent to the cluster address. When the server uses its own IP address, some clients will discard the server's response because it is not from an expected source IP address. You should use only UDP servers that use the destination IP address from the request when they respond to the client. In this case, the destination IP address from the request is the cluster address.

If you have an operating system that supports network interface aliasing such as AIX, Solaris, or Windows NT, you should alias the loopback device to the cluster address. The benefit of using an operating system that supports aliases is that you can configure the server machines to serve multiple cluster addresses.

If you have a server with an operating system that does not support aliases, such as HP-UX and OS/2, you must set ***lo0*** to the cluster address.

If your server is an MVS system running TCP/IP V3R2, you must set the VIPA address to the cluster address. This will function as a loopback address. The VIPA address must not belong to a subnet that is directly connected to the MVS node. If your MVS system is running TCP/IP V3R3, you must set the loopback device to the cluster address. If you are using high availability, you must enable RouteD in the MVS system so that the high availability takeover mechanism will function properly.

Note: The commands listed in this chapter were tested on the following operating systems and levels: AIX 4.1.5 and 4.2, HP-UX 10.2.0, Linux, OS/2 Warp Connect Version 3.0, OS/2 Warp Version 4.0, Solaris 2.5 (Sun OS 5.5), and Windows NT 3.51 and 4.0.

Use the command for your operating system as shown in Table 10 to set or alias the loopback device.

Table 10. Commands to alias the loopback device (lo0) for Dispatcher

System	Command
AIX	**ifconfig lo0 alias** *cluster_address*
HP-UX	**ifconfig lo0** *cluster_address*
Linux	**ifconfig lo:1 cluster_address netmask up**
OS/2	**ifconfig lo cluster_address**
Solaris	**ifconfig lo0:1 cluster_address 127.0.0.1 up**

Table 10. Commands to alias the loopback device (lo0) for Dispatcher (continued)

System	Command
Windows NT	a. Click Start, then click Settings. b. Click Control Panel, then double-click Network. c. If you have not done so already, add the MS Loopback Adapter Driver. 1) In the Network window, click Adapters. 2) Select MS Loopback Adapter, then click OK. 3) When prompted, insert your installation CD or disks. 4) In the Network window, click Protocols. 5) Select TCP/IP Protocol, then click Properties. 6) Select MS Loopback Adapter, then click OK. d. Set the loopback address to your cluster address. Accept the default subnet mask (255.0.0.0) and do not enter a gateway address. **Note:** You may have to exit and reenter Network Settings before the MS Loopback Driver shows up under TCP/IP Configuration.

2. Check for an extra route.

   On some operating systems a default route may have been created and needs to be removed.

   a. Check for an extra route on Windows NT with the following command: **route print**

   b. Check for an extra route on all UNIX systems and OS/2 with the following command: **netstat -nr**

   c. Windows NT Example: After route print is entered, a table similar to the following will be displayed. (This example shows finding and removing an extra route to cluster 9.67.133.158 with a default netmask of 255.0.0.0.)

```
Active Routes:
       Network Address          Netmask  Gateway Address        Interface  Metric
               0.0.0.0          0.0.0.0      9.67.128.1      9.67.133.67     1
               9.0.0.0        255.0.0.0     9.67.133.158     9.67.133.158     1
           9.67.128.0    255.255.248.0      9.67.133.67      9.67.133.67     1
          9.67.133.67  255.255.255.255        127.0.0.1        127.0.0.1     1
         9.67.133.158  255.255.255.255        127.0.0.1        127.0.0.1     1
        9.255.255.255  255.255.255.255      9.67.133.67      9.67.133.67     1
            127.0.0.0        255.0.0.0        127.0.0.1        127.0.0.1     1
            224.0.0.0        224.0.0.0     9.67.133.158     9.67.133.158     1
            224.0.0.0        224.0.0.0      9.67.133.67      9.67.133.67     1
      255.255.255.255  255.255.255.255      9.67.133.67      9.67.133.67     1
```

   d. Find your cluster address under the ″Gateway Address″ column. If you have an extra route, the cluster address will appear twice. In the example given, the cluster address (9.67.133.158) appears in row 2 and row 8.

   e. Find the network address in each row in which the cluster address appears. You need one of these routes and will need to delete the other route, which is extraneous. The extra route to be deleted will be the one whose network address begins with the first digit of the cluster address, followed by three zeroes. In the example shown, the extra route is the one in row two, which has a network address of 9.0.0.0:

```
9.0.0.0        255.0.0.0     9.67.133.158     9.67.133.158     1
```

3. Delete any extra routes.

   Use the command from Table 11 on page 106 for your operating system to delete any extra routes.

Table 11. Commands to Delete Routes for Various Operating Systems

Operating System	Command
AIX	**route delete -net** *network_address cluster_address*
HP-Unix	**route delete** *cluster_address cluster_address*
Solaris	No need to delete route.
OS/2	No need to delete route.
Windows NT	**route delete** *network_address cluster_address* **Note:** This command should be entered at an MS-DOS prompt.

Using Network Dispatcher with TN3270 Server

Network Dispatcher can be used with a cluster of 2210s, 2212s, Network Utilities or 2216s running TN3270 server function to provide TN3270E server support for large 3270 environments. The TN3270 advisor allows the Network Dispatcher to collect load statistics from each TN3270E server in real time to achieve the best possible distribution among the TN3270 servers. In addition to the TN3270 servers external to the Network Dispatcher router, one of the TN3270 servers in the cluster can be internal - it can run in the same router as Network Dispatcher.

Keys to Configuration

Configuration of the TN3270E servers is essentially the same whether or not you have a Network Dispatcher in front of the servers. In fact, the TN3270E server is unaware that the traffic from the clients is being dispatched through another machine. However, there are some points to keep in mind when setting up the external TN3270 servers for use with Network Dispatcher:

- Since the Network Dispatcher does not alter the destination IP address in the packets (i.e. the cluster address) it forwards to the servers, the TN3270 server IP address in each server must be set equal to the cluster IP address.
- The routers running TN3270 server function must know the IP address of the TN3270 function running in the router in order to deliver packets to the server function. Therefore, the TN3270 server IP address (i.e. the cluster address) must also be defined on each TN3270 server router as either the internal IP address of the router, or as a secondary address on one of the router's interfaces.
- You must ensure that any routing protocols being used on the TN3270E servers (for example, OSPF or RIP) will not advertise the cluster address. The Network Dispatcher router must "own" the cluster address as far as the client network is concerned. Network Dispatcher does not specifically advertise cluster addresses, but cluster addresses should be selected that are part of an advertised subnet that is local to the Network Dispatcher router.
- If the client to Network Dispatcher traffic flows on the same LAN as the Network Dispatcher to server traffic, you must make sure the servers do not respond to ARP for the cluster address, so the cluster address cannot be defined on the server's interface to this LAN. Network Dispatcher must be the only one responding to ARP on the LAN on which it receives client traffic. The cluster address can alternatively be configured on the TN3270 server as an interface address on another interface or it can be configured as the internal IP address of the TN3270 server.

- Each TN3270 server must be configured in Network Dispatcher with a unique server IP address. This address must also be configured as an interface address on the router performing the TN3270 server function.

When the TN3270 server is in the same router as Network Dispatcher, the following applies:

- The TN3270 server IP address for the internal TN3270 server must be set to the cluster address, but for the internal server, this address must not be defined on the router as the internal IP address or as the interface address.
- When the TN3270 server is external the TN3270 server IP address must be defined on the router as the internal IP address or as an interface address. When the TN3270 server is internal the TN3270 server IP address must not be defined on the router as the internal IP address or as an interface address. A TN3270 server can be set up as either internal or external, but it cannot be both and it cannot switch back and forth. As a result, when implementing an Network Dispatcher high availability solution with internal TN3270 servers in both Network Dispatcher routers, the Network Dispatcher in one router cannot load balance to the TN3270 server in the other router. It can load balance it's own internal server and any servers that are set up as external servers.

Explicit LUs and Network Dispatcher

Special care has to be taken for explicit LU definition in a Network Dispatcher environment. A session request for either a implicit or a explicit LU can be dispatched to any server. This means that the explicit LU has to be defined in each server, since it is not known in advance to which server the session will be dispatched.

Using Network Dispatcher with Scaleable High Availability Cache (SHAC)

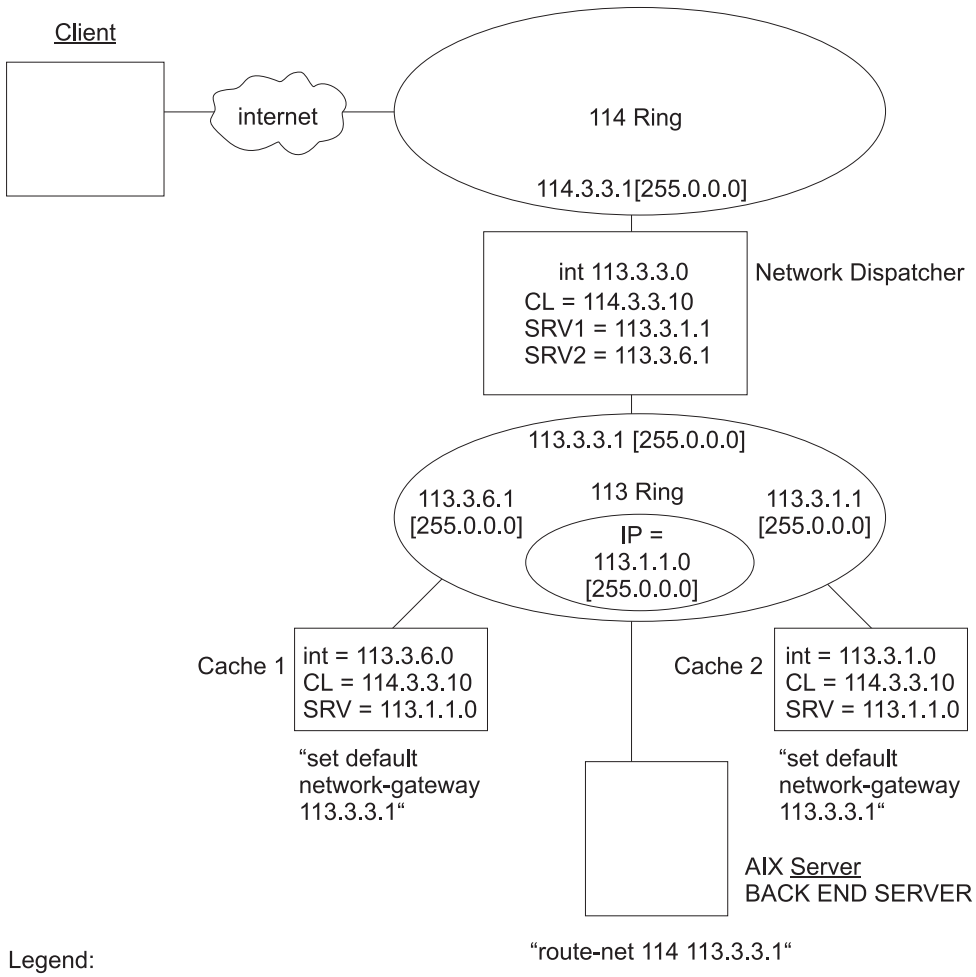
This section describes using Network Dispatcher with Scaleable High Availability Cache (SHAC) and Figure 9 on page 108 shows a diagram of a SHAC in a network. Scaleable High Availability Cache (SHAC) consists of a group of Web Server Caches plus a separate Network Dispatcher; the caches are configured as servers in the Network Dispatcher. The caches in a group share a common cluster and port. The identical cluster and port values are programmed in Network Dispatcher. The mode of the port is set to `extcache` to indicate that it feeds an external scalable cache array. See the **add port** command in "Add" on page 109.

Note: More than one cache group may be located in the Network Dispatcher.

The advisor and manager are critical to SHAC. The HTTP advisor must be enabled on any ports for which there are SHACs. The advisor queries are used to determine whether the configured caches are operational. Initially, at connection establishment time connections are routed to caches based on the manager. Therefore, the manager proportions should be set to incorporate the advisor. This is important if the caches become enabled or disabled.

Like other servers, the interface IP addresses of the caches are used for the server addresses. Figure 9 on page 108 shows an example. It includes the important IP addresses, netmasks, and routing information. In practice, most clients would be located on the Internet. In any case, the route from client to caches must go

through the Network Dispatcher (therefore, clients cannot be attached to the 113 ring in the picture).



Legend:

CL: Cluster Address. Note - this example assumes the use of port 80, the default http port.
INT: Internal address for 22XX router
SRV: Server address(s) associated with CL
“....”: additional routing commands to establish connectivity.

Figure 9. Two caches with Network Dispatcher, client and backend server

Chapter 9. Configuring and Monitoring the Network Dispatcher Feature

This chapter describes the Network Dispatcher Feature configuration and operational commands. It contains the following sections:

Accessing the Network Dispatcher Configuration Commands

To access the Network Dispatcher configuration environment:

1. Enter **talk 6** at the OPCON prompt (*).
2. Enter **feature ndr** at the `Config >` prompt.

Network Dispatcher Configuration Commands

Table 12 summarizes the Network Dispatcher configuration commands and the rest of the section explains these commands. Enter these commands at the `NDR Config >` prompt.

Table 12. Network Dispatcher Configuration Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page xxix.
Add	Configures various components of the Network Dispatcher including advisors, clusters, ports, and servers.
Clear	Clears the entire Network Dispatcher configuration.
Disable	Disables the backup, executor, and manager components of the Network Dispatcher. Also disables specific advisors.
Enable	Enables the backup, executor, and manager components of the Network Dispatcher. Also enables specific advisors.
List	Displays the entire Network Dispatcher Configuration or specific portions of the configuration.
Remove	Removes specific portions of the Network Dispatcher configuration.
Set	Changes the configuration parameters for advisors, clusters, ports, servers, or the Network Dispatcher manager.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page xxix.

Add

Use the **add** command to configure advisors, clusters, ports, servers, and reach addresses. For High Availability you can also configure whether this Network Dispatcher is a primary or backup and which IP addresses to use for heartbeat.

Syntax:

add advisor . . .

backup . . .

cluster . . .

heartbeat . . .

port . . .

reach . . .

server . . .

Advisor *name port# interval timeout comm-port*

Specifies the name and port for an advisor. This parameter also specifies how frequently the advisor will collect information on a particular protocol and a time period after which the advisor considers the protocol unavailable.

name Specifies the type of advisor.

Table 13. Advisor Names and Port Numbers

Advisor Number	Advisor Name	Default Port#
0	FTP	21
1	HTTP	80
2	MVS	10007
3	TN3270	23
4	SMTP	25
5	NNTP	119
6	POPS	110
7	TELNET	23

Valid values: 0 - 7

Default value: 1

port# Specifies the port number for this advisor.

Valid values: 1 to 65535

Default values: See Table 13.

interval

Specifies the frequency, in seconds, with which the advisor queries its protocol for each server. After half of this value without a response from the server, the adviser considers the protocol unavailable.

Valid values: 0 to 65535

Default value: 5

timeout

Specifies the interval of time, in seconds, after which the advisor considers the protocol unavailable.

To make sure that out-of-date information is not used by the manager in its load-balancing decisions, the manager will not use information from the advisor whose time stamp is older than the time set in this parameter. The advisor timeout should be larger than the advisor polling interval. If the timeout is smaller, the manager will ignore reports that should be used. By default, advisor reports do not time out.

This timeout value typically applies if you disable an advisor. Do not confuse this parameter with the interval/2 timeout previously described, which relates to a server not responding.

Valid values: 0 to 65535

Default value: 0, which means the protocol is considered always available.

Comm-port
Specifies the port number used by the TN3270 advisor to communicate with the TN3270 servers. This parameter is input only for the TN3270 advisor.

Valid values: 1 to 65535

Default value: 10008

Note: Because the manager component is a prerequisite for the advisor, you must enable the manager before any advisor can be enabled. You must also set the manager proportions so that the manager will consider advisor input when setting the server weights that are used to make load balancing decisions. You must also set the internal ip address using the **set internal-ip-address** command for the advisor to run correctly. See Configuring and Monitoring IP in *Protocol Configuration and Monitoring Reference Volume 1* for more information about the **set internal-ip-address** command.

Example 1:

```
add advisor
Advisor name (0=ftp,1=http,2=MVS,3=TN3270,4=smtp,5=nntp,6=pop3,7=telnet) [1]? 1
Port number [80]?
Interval (seconds) [5]? 10
Timeout (0=unlimited) [0]? 10
```

Example 2:

```
add advisor
Advisor name (0=ftp,1=http,2=MVS,3=TN3270,4=smtp,5=nntp,6=pop3,7=telnet) [1]? 3
Port number [23]?
Interval (seconds) [5]? 10
Timeout (0=unlimited) [0]? 10
Communication Port number [10008]?
```

backup *role strategy*
Specifies whether this Network Dispatcher is a backup or primary.

role Defines whether this is a primary or a backup Network Dispatcher. Use this command only if you intend to have a redundant configuration, and want the High Availability function to run. In this case, you must also configure the heartbeat (**add heartbeat**) and reachability (**add reach**).

Valid values: 0 or 1

0 = primary

1 = backup

Default value: 0

strategy
Specifies whether the Network Dispatcher will switch back to primary mode automatically or manually. Whenever a Primary Network Dispatcher fails and becomes standby (which means a backup performed the IP takeover function), and then becomes

available, it will automatically become the active Network Dispatcher if the strategy is set to *automatic*. If strategy is set to *manual*, the old primary will go to standby mode and the operator must use the **switchover** command in talk 5 to make it active again. See “Switchover” on page 133.

Valid values: 0 or 1

0 = automatic

1 = manual

Default value: 0

Example:

```
add backup
Role (0=Primary, 1=Backup) [0]?
Switch back strategy (0=Auto, 1=Manual) [0]?
```

cluster *address FIN-count FIN-timeout Stale-timer*

Specifies a cluster’s IP address and the frequency for the executor to perform garbage collection from the Network Dispatcher database. Network Dispatcher does not specifically advertise cluster addresses, so cluster addresses should be selected that are part of an advertised subnet that is local to the Network Dispatcher router. This would typically be the subnet on which Network Dispatcher receives client traffic from the next hop router.

Note: Cluster IP Addresses must not match the internal IP address of the router and must not match any interface IP addresses defined on the router.

address

Specifies the IP address for the cluster.

Valid values: Any valid IP address

Default value: 0.0.0.0

FIN-count

Specifies the number of connections that must be in FIN state before the executor tries to remove the unused connection information from the Network Dispatcher database after *FIN-timeout* or *Stale-timer* has elapsed.

Valid Values: 0 to 65535

Default value: 4000

FIN-timeout

Specifies the number of seconds, that a connection has been in the FIN state, after which the executor tries to remove the unused connection information from the Network Dispatcher database.

Valid Values: 0 to 65535

Default value: 30

Stale-timer

Specifies the number of seconds, that a connection has been inactive, after which the executor tries to remove a connection’s information from the Network Dispatcher database.

Valid Values: 0 to 65535

Default value: 1500

Example:

```
NDR Config>add cluster
Cluster Address [0.0.0.0]? 113.3.1.12
FIN count [4000]?
FIN time out [30]?
Stale timer [1500]?
Cluster 113.3.1.12 has been added.
Fincount has been set to 4000 for cluster 113.3.1.12
Fintimeout has been set to 30 for cluster 113.3.1.12
Staletimer has been set to 1500 for cluster 113.3.1.12
NDR Config>
```

heartbeat *address1 address2*

Specifies one path for Heartbeat messages. It is recommended that you configure more than one entry for reliable behavior. The Heartbeat message will flow from *address1*, which belongs to this Network Dispatcher, to *address2*, which belongs to the peer Network Dispatcher.

address1

Specifies the IP address of the interface of this Network Dispatcher from which Heartbeat messages will flow.

Valid Values: Any IP address.

Default value: 0.0.0.0

address2

Specifies the IP address of the interface of the peer Network Dispatcher to which Heartbeat messages will flow. This address must be reachable from the interface specified in *address1.*

Valid Values: Any IP address.

Default value: 0.0.0.0

Example:

```
add heartbeat
Source Heartbeat address [0.0.0.0]? 131.2.25.90
Target Heartbeat Address [0.0.0.0]? 131.2.25.92
```

port *cluster-address port# port-type max-weight port-mode*

Specifies the port and port's attributes.

cluster-address

Specifies the IP address of the cluster.

Valid Values: Any IP address.

Default value: 0.0.0.0

port# Specifies the port number of the protocol for this cluster.

Valid Values: 1 to 65535

Default value: 80

port-type

Specifies the types of IP traffic that can be load balanced on this port. Supported types are:

- 1 = TCP
- 2 = UDP
- 3 = both

Valid Values: 1, 2, 3

Default value: 3

max-weight
Specifies the maximum weight for servers on this port. This affects how much difference there can be between the number of requests the executor will give each server.

Valid Values: 0 to 100

Default value: 20

port-mode
Specifies whether the port will feed all requests from a single client to a single server (known as sticky), use passive ftp (pftp), feed an external scalable cache array (extcache), or use no particular protocols on this cluster (none).

Valid Values: 0, 1, 2, 4,, where:

- 0 = none
- 1 = sticky
- 2 = pftp
- 4 = extcache

Default value: 0

Example:

```
Config>feature ndr
NDR>add cluster 1.2.3.4 4000 30 1500
NDR>add port
Cluster address [0.0.0.0]? 1.2.3.4
Port number [80]? 80
Port type [3]?
Maximum weight [20]?
Port mode [0=none, 1=sticky, 2=pftp, 4=extcache ]? 0
```

reach *address*
Specifies any host address that the Network Dispatcher must be able to reach to run correctly. It can be a server address, a router address, an administration station address or other IP host.

address
Specifies the target IP address.

Valid Values: Any IP address

Default value: 0.0.0.0

Example:

```
add reach
Address to reach [0.0.0.0]?
```

server *cluster-address port# server-address server-weight server-state*
Specifies the attributes of a server in a cluster.

cluster-address
Specifies the IP address of the cluster to which this server belongs.

Valid Values: Any IP address

Default value: 0.0.0.0

port# Specifies the protocol running over the connection to this server.

Valid Values: 1 to 65535

Default value: 80

server-address
Specifies the IP address of the server.

Valid Values: Any IP address

Default value: 0.0.0.0

server-weight
Specifies the weight of the server for the executor. This affects how frequently the Network Dispatcher sends requests to this particular server.

Valid Values: 0 to the value of *max-weight* specified on the add port command.

Default value: max-weight on port command

server-state
Specifies whether the executor should regard the server as available or unavailable when the executor begins processing.

Valid Values: 0 (down) or 1 (up)

Default value: 1

Example:

```
add server
Cluster address [0.0.0.0]? 131.2.25.91
Port number [80]? 80
Server address [0.0.0.0]? 131.2.25.94
Server weight [35]?
Server state (down=0 up=1) [1]?
```

Parameter Configuration Limits

Table 14 lists the limits for the various items you can configure for a Network Dispatcher.

Table 14. Parameter Configuration Limits

Parameter	Limit
Advisors	8 per 2210
Clusters	32 per 2210
Heartbeats	8 per 2210
Ports	8 per cluster
Reachs	8 per 2210
Servers	32 per configured port, 128 for each port number under all clusters configured.
Unique server IP address	32 per 2210

Clear

Use the **clear** command to clear the entire Network Dispatcher configuration.

Syntax:

clear

Disable

Use the **disable** command to disable a Network Dispatcher component.

Syntax:

disable advisor . . .
backup
executor
manager

advisor *name port#*

Disables an advisor from the Network Dispatcher.

name Specifies the type of advisor.

See Table 13 on page 110 for additional information.

Valid values: 0 - 7

Default value: 0

port# Specifies the port number for this advisor.

Valid values: 1 to 65535

Default value: None. You must enter a port number.

Example:

```
disable advisor
Advisor name (0=ftp,1=http,2=MVS,3=TN3270,4=smtp,5=nntp,6=pop3,7=telnet) [1]? 1
Port number [0]? 80
```

backup

Disables the Network Dispatcher's backup function.

Example:

```
disable backup
Backup is now disabled.
```

executor

Disables the Network Dispatcher executor. Disabling the executor disables the Network Dispatcher feature.

Example:

```
disable executor
Executor is now disabled.
```

Note: Disabling the executor will stop the manager, advisors, and the high availability function, if they are currently running.

manager

Disables the Network Dispatcher manager. The manager is an optional component. However, if you do not use the manager, the Network Dispatcher will balance the load using a round-robin scheduling method based on the current server weights.

Example:

```
disable manager
Manager is now disabled.
```

Note: Because the manager component is prerequisite for advisors, disabling the manager will stop all the advisors from running.

Enable

Use the **enable** command to enable a Network Dispatcher component.

Syntax:

enable advisor . . .
backup
executor
manager

advisor *name port#*

Enables an advisor to the Network Dispatcher.

name Specifies the type of advisor.

See Table 13 on page 110 for additional information.

Valid values: 0 - 7

Default value: 0

port# Specifies the port number for this advisor.

Valid values: 1 to 65535

Default value: None. You must enter a port number.

Example:

```
enable advisor
Advisor name (0=ftp,1=http,2=MVS,3=TN3270,4=smtp,5=nntp=6=pop3,7=telnet) [1]? 1
Port number [0]? 80
```

Note: Because the manager component is a prerequisite for the advisor, you must enable the manager before any advisor can be enabled. You must also set the manager proportions so that the manager will consider advisor input when setting the server weights that are used to make load balancing decisions. You must also set the internal ip address using the **set internal-ip-address** command for the advisor to run correctly. See the chapter Configuring and Monitoring IP in *Protocol Configuration and Monitoring Reference Volume 1* for more information about the **set internal-ip-address** command.

backup

Enables the Network Dispatcher's backup function.

Example: `enable backup`

Note: Before enabling backup, you must add at least one heartbeat

executor

Enables the Network Dispatcher executor.

Example:

```
enable executor
Executor is now enabled.
```

manager

Enables the Network Dispatcher manager.

Example:

```
enable manager
Manager interval was set to 2.
Manager proportions were set to  50   50   0   0
Manager refresh cycle was set to 2
Manager sensitivity was set to 5.
Manager smoothing factor was set to 1.50.
```

When the manager is enabled for the first time, a manager record is created with the following default values:

Interval: 2 seconds

Refresh-Cycle: 2

Sensitivity: 5 %

Smoothing: 1.5

Proportions:

- **Active:** 50%
- **New:** 50%
- **Advisor:** 0
- **System:** 0

See “Set” on page 121 for a description of the above parameters.

List

Use the **list** command to display information about the Network Dispatcher.

Syntax:

list all | advisor | backup | cluster | manager | port | server

all Displays all Network Dispatcher configuration information. This includes the same information displayed for advisors, backup, cluster, manager, ports, and servers.

Example:

```
NDR Config> list all

Executor: Enabled

Manager: Enabled

        Interval        Refresh-Cycle   Sensitivity     Smoothing
        2               2               5 %             1.50
        Proportions:    Active  New     Advisor         System
                        50 %    50 %    0 %             0 %

Advisor:
        Name    Port    Interval        TimeOut         State     CommPort
        http    80      5               0               Enabled
        MVS     10007   15              0               Enabled
        TN3270  23      5               0               Enabled   10008
```

```
Backup: Enabled
        Role            Strategy
        PRIMARY         AUTOMATIC

        Reachability:   Address         Mask            Type
                        131.2.25.93     255.255.255.255 HOST
                        131.2.25.94     255.255.255.255 HOST

        HeartBeat Configuration:
           Source Address:  131.2.25.90 Target Address:   131.2.25.92
           Source Address:  132.2.25.90 Target Address:   132.2.25.92

Clusters:
        Cluster-Addr    FIN-count       FIN-timeout     Stale-timer
        131.2.25.91     4000            30              1500

Ports:
  Cluster-Addr    Port#   Weight  Port-Mode  Port-Type
  131.2.25.91     23      20 %    none       TCP
  131.2.25.91     80      20 %    none       Both

Servers:
        Cluster-Addr    Port#   Server-Addr     Weight  State
        131.2.25.91     23      131.2.25.93     20 %    up
        131.2.25.91     23      131.2.25.94     20 %    up
        131.2.25.91     80      131.2.25.93     20 %    up
        131.2.25.91     80      131.2.25.94     20 %    up
```

advisor
Displays the configuration for the Network Dispatcher advisors.

backup
Displays the backup configuration for the Network Dispatcher.

cluster
Displays the configuration of the Network Dispatcher clusters.

manager
Displays the configuration of the Network Dispatcher manager.

port Displays the configuration of the Network Dispatcher ports.

server Displays the configuration of the servers associated with the Network Dispatcher clusters.

Remove

Use the **remove** command to delete part of the Network Dispatcher configuration.

Syntax:

remove advisor . . .
backup
cluster . . .
heartbeat . . .
port . . .
reach . . .
server . . .

advisor *name port#*
Removes a specific advisor from the Network Dispatcher configuration.

name Specifies the type of advisor.

See Table 13 on page 110 for additional information.

Valid values: 0 - 7

Default value: 0

port# Specifies the port number for this advisor.

Valid values: 1 to 65535

Default value: None. You must enter a port number.

Example:

```
remove advisor
Advisor name (0=ftp,1=http,2=MVS,3=TN3270,4=smtp,5=nntp,6=pop3,7=telnet) [0]?
Advisor port [0]? 80
```

backup

Removes the high availability function.

Note: Because backup is a prerequisite for the heartbeat and reach functions removing backup will stop heartbeat and reach from running.

Example: `remove backup`

cluster *address*

Removes a cluster from the Network Dispatcher configuration.

address

Specifies the IP address for the cluster.

Valid values: Any valid IP address

Default value: 0.0.0.0

Note: Removing a cluster address also removes all the ports and servers associated with that cluster.

Example:

```
remove cluster
WARNING: Deleting a cluster will make any port or server
         associated with it to also be deleted.
Cluster address [0.0.0.0]? 131.2.25.91
```

heartbeat *address*

Removes the heartbeat address from the Network Dispatcher configuration.

address

Specifies the IP address for the target Network Dispatcher.

Valid values: Any valid IP address

Default value: 0.0.0.0

Example:

```
remove heartbeat
Target address [0.0.0.0]? 131.2.25.92
```

port *cluster-address port#*

Removes a port from a specific cluster in the Network Dispatcher configuration.

cluster-address

Specifies the IP address of the cluster.

Valid Values: Any IP address.

Default value: 0.0.0.0

port# Specifies the port number of the protocol for this cluster.

Valid Values: 1 to 65535

Default value: None. You must enter a port number.

Example:

```
remove port
WARNING: Deleting a port will make any server
associated with it also be deleted.   [0.0.0.0]? 7.82.142.15
Port number [0]? 80
Cluster address [0.0.0.0]? 20.21.22.15
```

reach *address*

Removes a server from the list of hosts the Network Dispatcher must be able to reach.

address

Specifies the IP address of the cluster.

Valid Values: Any IP address.

Default value: 0.0.0.0

Example:

```
remove reach
Target address [0.0.0.0]? 9.82.142.15
```

server *cluster-address port# server-address*

Removes a server from a cluster and port in the Network Dispatcher configuration.

cluster-address

Specifies the IP address of the cluster.

Valid Values: Any IP address.

Default value: 0.0.0.0

port# Specifies the port number of the protocol for this cluster.

Valid Values: 1 to 65535

Default value: None. You must enter a port number.

server-address

Specifies the IP address of the cluster.

Valid Values: Any IP address.

Default value: 0.0.0.0

Example:

```
remove server
Cluster address [0.0.0.0]? 7.82.142.15
Port number [0]? 80
Server address [0.0.0.0]? 20.21.22.15
```

Set

Use the **set** command to change the attributes of an existing advisor, cluster, port, or server. You can also define attributes for the Network Dispatcher manager.

Syntax:

set advisor . . .

cluster . . .

manager . . .

port . . .

server . . .

advisor *name port# interval timeout comm-port*

Changes the port number, interval, and timeout for an advisor.

name Specifies the type of advisor.

See Table 13 on page 110 for additional information.

Valid values: 0 - 7

Default value: 0

port# Specifies the port number for this advisor.

Valid values: 1 to 65535

Default value: None. You must enter a port number.

interval

Specifies the frequency with which the advisor queries its protocol for each server. After half of this value expires without a response from the server, the adviser considers the protocol unavailable.

Valid values: 0 to 65535

Default value: 5

timeout

Specifies the interval of time, in seconds, after which the advisor considers the protocol unavailable.

To make sure that out-of-date information is not used by the manager in its load-balancing decisions, the manager will not use information from the advisor whose time stamp is older than the time set in this parameter. The advisor timeout should be larger than the advisor polling interval. If the timeout is smaller, the manager will ignore reports that should be used. By default, advisor reports do not time out.

This timeout value typically applies if you disable an advisor. Do not confuse this parameter with the interval/2 timeout previously described, which relates to a server not responding.

Valid values: 0 to 65535

Default value: 0, which means the protocol is considered always available.

comm-port

Specifies the port number used by the TN3270 advisor to communicate with the TN3270 servers. This parameter is input only for the TN3270 advisor.

Valid values: 1 to 65535

Default value: 10008

Example:

```
set advisor
Advisor name (0=ftp,1=http,2=MVS,3=TN3270,4=smtp,5=nntp=6=pop3,7=telnet) [0]?
Port number [0]? 21
Interval (seconds) [5]? 10
Timeout (0=unlimited) [0]? 20
```

cluster *address FIN-count FIN-timeout Stale-timer*
Changes the FIN-count, FIN-timeout, and Stale-timer for a cluster in the Network Dispatcher configuration.

address
Specifies the IP address for the cluster.

Valid values: Any valid IP address

Default value: 0.0.0.0

FIN-count
Specifies the number of connections that must be in FIN state before the executor tries to remove the unused connection information from the Network Dispatcher database after *FIN-timeout* or *Stale-timer* has elapsed.

Valid Values: 0 to 65535

Default value: 4000

FIN-timeout
Specifies the number of seconds after which the executor tries to remove the unused connection information from the Network Dispatcher database.

Valid Values: 0 to 65535

Default value: 30

Stale-timer
Specifies the number of seconds that a connection has been inactive, after which the executor tries to remove a connection's information from the Network Dispatcher database.

Valid Values: 0 to 65535

Default value: 1500

Example:

```
set cluster
Cluster address [0.0.0.0]? 131.2.25.91
FIN count [4000]? 4500
FIN timeout [30]? 40
Stale timer [1500]? 2000
```

manager *interval proportion refresh sensitivity smoothing*
Sets the values that the manager uses to determine the best server to satisfy a request.

interval
Specifies the amount of time, in seconds, after which the manager updates the server weights that the executor uses in load balancing connections.

Valid values: 0 to 65535

Default value: 2

proportion
Specifies the relative importance of external factors in the manager's weighting decisions. The sum of the proportions must equal 100. The factors are:

active The number of active connections on each TCP/IP server as tracked by the executor.

Valid values: 0 to 100

Default value: 50

new The number of new connections on each TCP/IP server as tracked by the executor.

Valid values: 0 to 100

Default value: 50

advisor
Input from the protocol advisors defined to the Network Dispatcher.

Valid values: 0 to 100

Default value: 0

system
Input from the MVS system advisor provided by the MVS WLM system monitoring tool.

Valid values: 0 to 100

Default value: 0

refresh
Specifies the frequency with which the manager requests status from the executor. This parameter is specified as a number of *intervals*.

Valid values: 0 to 100

Default value: 2

sensitivity
Specifies the percentage weight change for all the servers on a port, after which the manager updates the weights that the executor uses in load balancing connections.

Valid values: 0 to 100

Default value: 5

smoothing
Specifies a limit to the amount that a server's weight can change. Smoothing minimizes the frequency of change in the distribution of requests. A higher smoothing index will cause the weights to change less. A lower smoothing index will cause the weights to change more.

Valid values: a decimal value between 1.0 and 42 949 673.00

Default value: 1.5

Note: You can only specify two places after the decimal point.

Example:

```
set manager
Interval (in seconds) [2]? 3
Active proportion [50]? 40
New proportion [50]? 38
Advisor proportion [0]? 20
System proportion [0]? 2
Refresh cycle [2]? 4
Sensitivity threshold [5]? 10
Smoothing index (>1.00) [1.50]? 200
```

port *cluster-address port# port-type max-weight port-mode*

Changes the port-type, max-weight, and port-mode for a specific cluster and port number.

cluster-address

Specifies the IP address of the cluster.

Valid Values: Any IP address.

Default value: 0.0.0.0

port# Specifies the port number of the protocol for this cluster.

Valid Values: 1 to 65535

Default value: None. You must enter a port number.

port-type

Specifies the type of IP traffic that can be load balanced on this port.

Valid Values:

tcp=1

upd=2

both=3

Default value: 3

max-weight

Specifies the weight for servers on this port. This affects how much difference there can be between the number of requests the executor will give each server.

Valid Values: 0 to 100

Default value: 20

port-mode

Specifies whether the port will feed all requests from a single client to a single server (known as sticky), use passive ftp (pftp), feed an external scaleable cache array, or use no protocols on this cluster (none).

Valid Values:

none=0

sticky=1

pftp=2

extcache=4

Default value: 0 (none)

Example:

```
set port
Cluster address [0.0.0.0]? 131.2.25.91
Port number [0]? 23
Port type (tcp=1, udp=2, both=3) [0]?
Max. weight (0-100) [20]? 30
Only one pftp port per cluster allowed
Port mode (none=0, sticky=1, pftp=2 extcache=4) []?
```

server *cluster-address port# server-address weight state*

Changes the server state, and server weight for a specific server in a cluster.

cluster-address

Specifies the IP address of the cluster to which this server belongs.

Valid Values: Any IP address

Default value: 0.0.0.0

port# Specifies the port number of the protocol for this cluster.

Valid Values: 1 to 65535

Default Value: None. You must enter a port number.

server-address

Specifies the IP address of the server.

Valid Values: Any valid server address

Default Value: 0.0.0.0

state Specifies whether the executor should regard the server as available or unavailable when the executor begins processing.

Valid Values: 0 (down) or 1 (up)

Default value: 1

weight

Specifies the weight of the server for the executor. This affects how frequently the Network Dispatcher sends requests to this particular server.

Valid Values: 0 to the value of *max-weight* specified on the add port command.

Default value: max-weight on port command

Example:

```
set server
Cluster address [0.0.0.0]? 131.2.25.91
Port number [0]?
Server address [0.0.0.0]?
Server weight [20]? 25
Server state (down=0, up=1) [1]? 1
```

Accessing the Network Dispatcher Monitoring Commands

To access the Network Dispatcher monitoring environment:

1. Enter **talk 5** at the OPCON prompt (*).
2. Enter **feature ndr** at the GWCON prompt (+).

Network Dispatcher may also be monitored using SNMP. Refer to “SNMP Management” in the *Protocol Configuration and Monitoring Reference Volume 1* for more information.

Network Dispatcher Monitoring Commands

Table 15 summarizes the Network Dispatcher monitoring commands and the rest of the section explains these commands. Enter these commands at the `NDR >` prompt.

Table 15. Network Dispatcher Monitoring Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
List	Displays the currently configured attributes of the advisor, clusters, ports, or servers.
Quiesce	Specifies that no more connection request should be sent to a server. Also temporarily stops the heartbeat and reach functions.
Report	Displays a report of information related to the advisor and the manager.
Status	Displays the current status of the counters, clusters, ports, servers, advisor, manager, and backup.
Switchover	Forces a Network Dispatcher that is running in standby mode to become the active Network Dispatcher. Use of this command is necessary if you specified manual as the switchover mode.
Unquiesce	Allows the Network Dispatcher manager to assign a weight greater than 0 to a previously quiesced server on every port that the server is configured. This action allows new connection requests to flow to the selected server.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

List

Use the **list** command to display information about the Network Dispatcher.

Syntax:

list advisor
cluster
port
server

advisor

Displays the configuration for the Network Dispatcher advisors.

Example:

```
list advisor
Advisor list requested.

-----------------------------------------
| ADVISOR |  PORT  |  TIMEOUT  |  STATUS  |
-----------------------------------------
|  ftp    |    21  |       5   |  ACTIVE  |
|  Http   |    80  | unlimited |  ACTIVE  |
|  MVS    | 10007  | unlimited |  ACTIVE  |
|  TN3270 |    23  | unlimited |  ACTIVE  |
-----------------------------------------
```

Configuring Network Dispatcher

cluster

Displays the configuration of the Network Dispatcher clusters.

Example:

```
list cluster
EXECUTOR INFORMATION:
 --------------------
Version: 01.01.00.00 - Tue Dec 10 14:15:58 EST 1996
Number of defined clusters: 2

CLUSTER LIST:
-------------
  131.2.25.91
  10.11.12.2
```

port Displays the configuration of the Network Dispatcher ports.

Example:

```
list port
Cluster Address [0.0.0.0]? 131.2.25.91
```

CLUSTER: 131.2.25.91			
PORT	MAXWEIGHT	PORT MODE	PORT TYPE
23	30	none	TCP
80	20	none	both

server Displays the configuration of the servers associated with the Network Dispatcher clusters.

Example:

```
list server
Cluster Address [0.0.0.0]? 131.2.25.91

PORT 23 INFORMATION:
 --------------------
Maximum weight.................. 20
Port mode....................... NONE
Port type....................... TCP
All up nodes are weight zero.... FALSE
Total target nodes.............. 2
Currently marked down........... 0
Servers providing service to this port:
Address: 131.2.25.93 Weight: 20 Count: 0 TCP Count: 0 UPD Count: 0
Active: 0 FIN 0 Complete 0 Status: up Saved Weight: -1
Address: 131.2.25.94 Weight: 20 Count: 0 TCP Count: 0 UPD Count: 0
Active: 0 FIN 0 Complete 0 Status: up Saved Weight: -1

PORT 80 INFORMATION:
 --------------------
Maximum weight.................. 20
Port mode....................... NONE
Port type....................... BOTH
All up nodes are weight zero.... FALSE
Total target nodes.............. 2
Currently marked down........... 0
Servers providing service to this port:
Address: 131.2.25.93 Weight: 20 Count: 0 TCP Count: 0 UPD Count: 0
Active: 0 FIN 0 Complete 0 Status: up Saved Weight: -1
Address: 131.2.25.94 Weight: 20 Count: 0 TCP Count: 0 UPD Count: 0
Active: 0 FIN 0 Complete 0 Status: up Saved Weight: -1
```

Quiesce

Use the **quiesce** command to temporarily stop the heartbeat or reach functions or to specify that no more connection requests should be sent to a server.

Syntax:

quiesce heartbeat
manager
reach

heartbeat *address*
Stops the selected path for the heartbeat function. The *address* is the IP address of the remote network dispatcher to which this Network Dispatcher is sending Heartbeat messages.

Example:

```
quiesce heartbeat
Remote Address [0.0.0.0]? 131.2.25.94
```

manager *address*
Specifies that no more connection requests are to be made to the specified server. *Address* is the IP address of the server.

Example:

```
quiesce manager
Server Address [0.0.0.0]? 131.2.25.93
```

reach *address*
Stops the Network Dispatcher's polling of the specified address to determine if it is reachable, where *address* is the IP address that is part of the reachability criteria.

Example:

```
quiesce reach
Reach Address [0.0.0.0]? 131.2.25.92
```

Report

Use the **report** command to display a report of the advisor or manager

Syntax:

report advisor
manager

advisor *type port#*
Displays a report of information about a specific advisor.

type Is the type of advisor. See Table 13 on page 110 for advisor types.

port# Is the port number.

Example:

```
report advisor
0=ftp,1=http,2=MVS,3=TN3270,4=smtp,5=nntp,6=pop3,7=telnet
Advisor name [0]? 1
Port number [0]? 80

------------------------------
|      ADVISOR:     http     |
|      PORT:          80     |
------------------------------
| 131.2.25.93      |       0  |
| 131.2.25.94      |      16  |
------------------------------
```

manager
Displays a report of the current manager information.

Example:

```
report manager

---------------------------------
|  HOST TABLE LIST  |   STATUS   |
---------------------------------
|  131.2.25.93      |    ACTIVE  |
|  131.2.25.94      |    ACTIVE  |
---------------------------------

-------------------------------------------------------------------------------
|131.2.25.91    |WEIGHT | ACTIVE % 50  |  NEW % 50    | PORT %  0 |SYSTEM %  0|
|               ---------------------------------------------------------------|
|PORT:   23     |NOW|NEW| WT | CONNECT | WT | CONNECT | WT | LOAD | WT | LOAD |
-------------------------------------------------------------------------------
|131.2.25.93    | 10| 10|  10|        0|  10|        0|   0|     0|-999|    -1|
|131.2.25.94    | 10| 10|  10|        0|  10|        0|   0|     0|-999|    -1|
|---------------------------------------------------------------------------|
|PORT TOTALS:   | 20| 20|    |        0|    |        0|    |     0|    |    -2|
-------------------------------------------------------------------------------


-------------------------------------------------------------------------------
|131.2.25.91    |WEIGHT | ACTIVE % 50  |  NEW % 50    | PORT %  0 |SYSTEM %  0|
|               ---------------------------------------------------------------|
|PORT:   80     |NOW|NEW| WT | CONNECT | WT | CONNECT | WT | LOAD | WT | LOAD |
-------------------------------------------------------------------------------
|131.2.25.93    | 10| 10|  10|        0|  10|        1|  16|     0|-999|    -1|
|131.2.25.94    | 10| 10|  10|        0|  10|        1|   3|    16|-999|    -1|
|---------------------------------------------------------------------------|
|PORT TOTALS:   | 20| 20|    |        0|    |        0|    |    16|    |    -2|
-------------------------------------------------------------------------------


-------------------------------------------
| ADVISOR |  PORT  |  TIMEOUT  |  STATUS  |
-------------------------------------------
|  http   |    80  | unlimited |   ACTIVE |
|  MVS    | 10007  | unlimited |   ACTIVE |
-------------------------------------------
Manager report requested.
```

Status

Use the **status** command to obtain the status of the advisors, backup, counter, clusters, manager, ports, and servers.

Syntax:

status advisor
backup
cluster
counter
manager
ports
servers

advisor *name port#*

Obtains the status of a specific advisor.

name Specifies the type of advisor. See Table 13 on page 110 for advisor types.

port# Is the port number.

Example:

```
status advisor
0=ftp, 1=http, 2=MVS 3=TN3270, 4=SMTP, 5=NNTP, 6=POP3, 7=TELNET
Advisor name [0]?
Port number [0]? 21
```

```
Advisor ftp on port 21 status:
==============================
Interval.................. 10
```

backup

Obtains the status of the backup function.

Example:

```
status backup
 Dumping status ...
 Role : PRIMARY   Strategy : AUTOMATIC  State : ND_ACTIVE Sub-State : ND_SYNCHRONIZED
 <<Preferred Target : 132.2.25.92>>

 Dumping HeartBeat Status ...
 .....Heartbeat target : 131.2.25.92 Status : UNREACHABLE
 .....Heartbeat target : 132.2.25.92 Status : REACHABLE

 Dumping Reachability Status ...
 .....Host:131.2.25.93   Local:REACHABLE
 .....Host:131.2.25.94   Local:REACHABLE
```

cluster *address*

Obtains the status of a specified cluster, where *address* is the IP address of the cluster.

Example:

```
status cluster
Cluster Address [0.0.0.0]? 131.2.25.91

EXECUTOR INFORMATION:
 ---------------------
Version: 01.01.00.00 - Tue Dec 10 14:15:58 EST 1996

CLUSTER INFORMATION:
 --------------------
Address......................... 131.2.25.91
Number of target ports.......... 2
FIN clean up count.............. 4000
Connection FIN timeout.......... 30
Active connection stale timer... 1500

PORT 23 INFORMATION:
 --------------------
Maximum weight.................. 20
Port mode....................... NONE
Port type....................... TCP
All up nodes are weight zero.... FALSE
Total target nodes.............. 2
Currently marked down........... 0
Servers providing service to this port:
Address: 131.2.25.93 Weight: 20 Count: 0
Active: 0 FIN 0 Status: up Saved Weight: -1
Address: 131.2.25.94 Weight: 20 Count: 0
Active: 0 FIN 0 Status: up Saved Weight: -1

PORT 80 INFORMATION:
 --------------------
Maximum weight.................. 20
Port type....................... BOTH
Port mode....................... NONE
All up nodes are weight zero.... FALSE
Total target nodes.............. 2
Currently marked down........... 0
Servers providing service to this port:
Address: 131.2.25.93 Weight: 20 Count: 0
Active: 0 FIN 0 Status: up Saved Weight: -1
Address: 131.2.25.94 Weight: 20 Count: 0
Active: 0 FIN 0 Status: up Saved Weight: -1
```

counter

Obtains the status of all counters.

Example:

```
status counter
Internal counters from executor:
--------------------------------
Total number of packets into executor..... 2684
Total packets for cluster processing (C)... 2684
Packets not addressed to a cluster(port)... 0

Cluster processing results:
```

```
--------------------------
Errors..................................... 0
Discarded.................................. 0
Forward requested.......................... 2684
Forward requested.......................... 0
Forward discarded with error............... 0

Other processing problems:
--------------------------
Total packets dropped (C).................. 0
```

manager

Obtains the status of the manager.

Example:

```
status manager
Number of defined hosts... 2
Sensitivity.............. 0%
Smoothing factor.......... 2
Interval.................. 3
Weights refresh cycle..... 4

Active connections gauge proportion......... 40%
New connections counter(delta) proportion... 38%
Advisor gauge proportion.................... 20%
System Metric proportion....................  2%

Manager status requested.
```

port *cluster-address port#*

Obtains the status of a specific port, where:

cluster-address
is the IP address of the cluster.

port# is the port number on the cluster.

Example:

```
status port
Cluster Address [0.0.0.0]? 131.2.25.91
Port number  [0]? 80

PORT 80 INFORMATION:
 --------------------
Maximum weight.................. 20
Port mode....................... NONE
Port type....................... BOTH
All up nodes are weight zero.... FALSE
Total target nodes.............. 2
Currently marked down........... 0
Servers providing service to this port:
Address: 131.2.25.93 Weight: 20 Count: 12345 TCP Count: 10000 UDP count 2345
Active: 3431 FIN 3780 Complete 3431 Status: up Saved Weight: -1
Address: 131.2.25.94 Weight: 20 Count: 7890  Active: 2980 FIN 2390 Status: up
Saved Weight: -1
```

server *address*

Obtains the status of a specific server, where *address* is the IP address of the cluster to which the server belongs.

Example:

```
status server
Cluster Address [0.0.0.0]? 131.2.25.91

PORT 23 INFORMATION:
 --------------------
Maximum weight.................. 20
Port mode....................... NONE
Port type....................... TCP
All up nodes are weight zero.... FALSE
Total target nodes.............. 2
Currently marked down........... 0
Servers providing service to this port:
Address: 131.2.25.93 Weight: 20 Count: 140 TCP Count: 100 UDP Count: 40
Active: 50 FIN 45 Complete 50  Status: up Saved Weight: -1
Address: 131.2.25.94 Weight: 20 Count: 250 TCP Count: 100 UDP Count: 40
Active: 60 FIN 54 Complete 50  Status: up Saved Weight: -1

PORT 80 INFORMATION:
```

```
 --------------------
Maximum weight.................. 20
Port mode....................... NONE
Port type....................... BOTH
All up nodes are weight zero.... FALSE
Total target nodes.............. 2
Currently marked down........... 0
Servers providing service to this port:
Address: 131.2.25.93 Weight: 20 Count: 12345 TCP Count: 10000 UDP Count: 2345
Active: 3431 FIN 3780 Complete 3431  Status: up Saved Weight: -1
Address: 131.2.25.94 Weight: 20 Count: 7890  TCP Count: 10000 UDP Count: 2345
Active: 2980 FIN 2390 Complete 3431  Status: up Saved Weight: -1
```

Switchover

Use the **switchover** command to force a Network Dispatcher that is running in standby mode to become the active Network Dispatcher when the switchover strategy is manual. This command must be entered on the host that is running the Network Dispatcher that is in standby mode.

Syntax:

switchover

Unquiesce

Use the **unquiesce** command to restart a heartbeat, manager, or reach function that was previously stopped with the **quiesce** command.

Syntax:

unquiesce heartbeat
manager
reach

heartbeat *address*

Restarts the path for Heartbeat messages, where *address* is the IP address of the remote network dispatcher to which this Network Dispatcher is sending Heartbeat messages.

Example:

```
unquiesce heartbeat
Remote Address [0.0.0.0]? 9.10.11.1
```

manager *address*

Restarts sending connection requests to the specified server. *Address* is the IP address of the server.

Example:

```
unquiesce manager
Server Address [0.0.0.0]? 20.21.22.15
```

reach *address*

Restarts the Network Dispatcher's polling of the specified address to determine if it is reachable, where *address* is the IP address that is part of the reachability criteria.

Example:

```
unquiesce reach
Reach address [0.0.0.0]? 20.3.4.5
```

Chapter 10. Configuring and Monitoring the Encoding Subsystem

Data compression and encryption functions are grouped together in the Encoding Subsystem (ES). ES provides access to the encoding software device for interfaces or protocols and is automatically activated whenever a link is activated for compression or encryption. The software device consists of operational software that performs compression and encryption. The compression and encryption algorithms are run on the router's processor. You do not need to change the default configuration to use the software device.

Note: See "Chapter 11. Configuring and Monitoring Data Compression" on page 143 for instructions about configuring compression sessions over PPP or Frame Relay, see "Chapter 14. Using and Configuring Encryption Protocols" on page 179 for instructions about configuring encryption sessions over PPP or Frame Relay, and see "Chapter 19. Configuring and Monitoring IP Security" on page 273 for instructions about configuring IPSec sessions.

Monitoring the ES activity can be done by entering **feature es** from the monitoring (talk 5) prompt.

The ES configuration parameters allow you to limit the amount of memory used by the ES software device. The default configuration allows the ES to get as much memory as required. To limit memory usage, use the **set** command under **feature es** in the configuration process (Talk 6).

This chapter consists of the following sections:

Configuring the Encoding Subsystem

The ES configuration parameters provide a way to control the number of compression and encryption sessions that are using the software encoding device at one time. The software encoding device is essentially a collection of compression and encryption libraries that are run on the router's processor. A session consists of a full-duplex connection over a particular interface that has been configured to use compression or encryption.

Generally, data encoding is a processor-intensive operation. By limiting the number of software encoding sessions, the impact of data encoding on the performance of the router can be controlled to a certain extent. As an example, if the router has 20 dial-in interfaces configured for compression and it has been determined that compressing more than 10 interfaces at once has an adverse effect on the performance of the router, the maximum number of compression sessions should be set to 10. This allows any 10 of the 20 interfaces to use compression.

The memory requirements of the software encoding device may also be a reason to limit the number of sessions. Each software compression session uses approximately 30 KB of router memory and an encryption session uses approximately 2 KB. If too much memory is used by the ES, other functions may

become memory-restricted and the router's performance can be adversely affected. See "Considerations" on page 146 for more information.

You can set the minimum or maximum number of ES sessions by stating the number of sessions or by specifying one of the values *unlimited*, *default*, or a number. The values *unlimited* and *default* have the same meaning; these values allow the router to support all the sessions that have been activated for encryption or compression, until the memory is exhausted.

Note: None of the ES configuration parameters (talk 6) can be dynamically reconfigured. To activate parameter values after you have changed them, you must restart or reload the router.

In the Config process (talk 6), enter **feature es** at the `Config>` prompt to access the ES configuration commands. The `ES Config>` prompt appears. Table 16 lists the commands.

Table 16. ES Configuration Commands

Command	Action
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page xxix.
List	Displays the current setting of compression and encryption sessions.
Set	Sets the maximum number of encryption and compression sessions available for all interfaces.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page xxix.

List

Use the **list** command to display the current setting of the compression and encryption sessions.

Syntax:

list

Example:

```
ES Config> list
Data Compression and Encryption System Configuration
----------------------------------------------------

Parameters used for host-based encoding:
  Compression sessions:
    Reserved at initial bootup:           0
    Maximum allowed:                      unlimited
  Encryption sessions:
    Reserved at initial bootup:           0
    Maximum allowed:                      unlimited
```

Set

Use the **set** command to set the maximum number of data encryption or compression sessions.

Syntax:

set sw minimum compression-sessions *n, unlimited, or default*

sw maximum compression-sessions *n, unlimited, or default*

sw minimum encryption-systems *n, unlimited, or default*

sw maximum encryption-systems *n, unlimited, or default*

Note: The letters sw are an abbreviation for software.

software minimum compression-sessions *n, unlimited, or default*
Sets the minimum number of compression sessions available for the interfaces. The router reserves this many sessions so that they are always available.

Default Value: 0

Valid Values: 0 to *unlimited*; alternatively, *default*

software maximum compression-sessions *n, unlimited, or default*
Sets the maximum number of compression sessions available for the interfaces. Once this number of sessions has been activated, new sessions cannot be activated.

Default Value: 0

Valid Values: 0 to *unlimited*; alternatively, *default*

software minimum encryption-sessions *n, unlimited, or default*
Sets the minimum number of encryption sessions available for the interfaces. The router reserves this number of sessions so that they are always available.

Default Value: 0

Valid Values: 0 to *unlimited*; alternatively, *default*

software maximum encryption-sessions *n, unlimited, or default*
Sets the maximum number of encryption sessions available for the interfaces. Once this number of sessions has been activated, new sessions cannot be activated.

Default Value: 0

Valid Values: 0 to *unlimited*; alternatively, *default*

Monitoring the Encoding Subsystem

In the monitoring process, enter **feature es** at the + prompt to access the ES monitoring commands. The `ES Monitor>` prompt appears. Table 17 lists the available commands.

Table 17. ES Monitoring Command

Command	Action
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
List	Lists ES ports, circuits, devices, configuration, status, or summary.

Table 17. ES Monitoring Command (continued)

Command	Action
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

List

Use the **list** command to list information about ES. See the **list summary** command for an example of the output of the **list** command that includes ports, devices, and status.

Syntax:

list ports
circuits
devices
config
status
summary

ports The **list ports** command lists the encoding ports that have been created by potential clients of the encoding system. A port establishes a linkage between the encoding system and the clients that have been configured to use ES. For example, if compression or encryption is configured over the PPP interface Net 1, a port is associated with that interface. The QLen field shows the sum of all the outstanding compression or encryption requests for all of the circuits associated with the port. A client, such as PPP configured over a particular interface, presents a request to ES when it designates a particular buffer of data for encoding.

The Status field shows *Idle* if nothing is queued at the port, or *Busy* or *Waiting* if requests are in process or queued on the port.

circuits

The **list circuits** command displays the circuits that have been defined by clients of the encoding system. Each circuit corresponds to a full-duplex connection. Date encrypted or compressed at one endpoint is decrypted or decompressed at the other.

By default, only active circuits are displayed. Use the command **list circuits all** to include both active and inactive circuits.

For each circuit found, the port and user are displayed as in the **list ports** command. In addition, two lines of information are shown, a Tx line for the outbound circuit and an Rx line for the inbound circuit. The circuit ID is an arbitrary number provided by the client so that it can tag each circuit that it creates. For Frame Relay circuits, this number corresponds to the ID of the associated Frame Relay data-link circuit (DLCI). Point-to-Point links create only one circuit, which is always identified by the number *1*.

In addition, the following items are displayed:

Dev This is the number that represents the encoding device that is servicing that stream. It is 1 when the encoding is being done by software activating the CPU and 2 when the encoding is being done by the compression/encryption adapter.

Cmpr This field displays the compression or decompression algorithm active for that stream. If it is *LZC*, STAC-LZC compression is being used; if it is *MPPC*, Microsoft® PPC is used. An asterisk (*) is appended to the name of the algorithm if the stream is operating in stateless mode. Stateless mode is a mode in which the history of the data packet is not maintained after that packet has been processed, as opposed to continuous mode in which history is maintained from handling one packet in order to handle the next. For example, in continuous compression, the encoder maintains a cache of information gathered from previous packets in order to more effectively compress the current packets.

Encr This field displays the encryption or decryption algorithm being used. It is *DES* for standard DES, *3DES* for Triple DES, or *RC4* if RSA's RC4 algorithm is used. An asterisk (*) is appended to the name if the stream is operating in stateless mode. This is significant for RC4 but means little for DES/3DES. Note that the name shown corresponds to the basic encryption algorithm employed, not to the encapsulation format used by the client. For example, PPP supports two encapsulation methods: DESE (RFC 1969) which encrypts with DES, and MPPE (Microsoft nonstandard), which uses RC4.

QLen This parameter shows the number of outstanding packets sitting in the stream's queue waiting to be encoded or decoded. Note that this number only reflects packets that have actually been submitted to ES for processing. Some clients may keep their own queues and feed only a few packets at a time to the encoding system from these private queues.

Status

A quick indication of the stream's status. It is not unusual for all streams to have a waiting status and none to appear to be busy. Seeing a busy status requires catching the queue activity during a fairly narrow window of time in the processing cycle. These are the possible states:

Idle No packets are queued on this stream

Busy The system is currently processing packets on this stream (meaning that the item at the head of the queue is going through the encoding engine at that moment).

Waiting

Requests are pending, but nothing from that stream is currently undergoing processing.

devices

The **list devices** command lists the encoding devices that the system has available to it. An encoding device usually refers to a compression/encryption adapter. The software that is used when a hardware accelerator is not available is implemented as a virtual device and will also show up in this list as a *Host Software* device. There are two forms for this command: **list devices** and **list device n**. The first form produces a short summary listing of all the devices recognized by the system. The second form will produce a detailed listing for a specific device n, where n is the unit number. Unit 1 represents host software, which is a virtual encoding device, and unit 2 represents the compression/encryption adapter. An asterisk (*) can be used in place of the number n, in which case a listing is provided for both units.

config The **list config** command displays the current configuration parameters. These are the parameters read from the non-volatile memory at the time that the router is restarted or reloaded. The information displayed is identical to that displayed by the configuration (Talk 6) **list config** command.

status The **list status** command displays the encoding system status, which consists of some global status flags and some miscellaneous system statistics. These are the descriptions of the fields that are displayed by the **list status** command:

Last Error
The last error code returned to any client of the encoding system. This is meant for debugging and should be used by service personnel.

Internal Condition flags
This field shows certain internal conditions, as defined in the following list:

Ready The system is up and operational. This is the normal condition.

Not Working
The encoding system is inoperative due to some internal error.

No Devices Available
Indicates that no device is available to do the encoding. This condition should not occur because if a hardware-based encoder is not present, encoding is accomplished by internal software.

Out of Memory
The system tried to allocate memory and failed. This condition indicates that the router is low on RAM and that the encoding system has been adversely affected.

Number of Ports
This field indicates the number of clients that have established ports for themselves in the ES. See the **list ports** command for a definition of a port.

Number of Circuits
See the **list circuits** command for a definition of circuits.

Global Request pool size
The number of request buffers allocated and free. Roughly one request buffer is used for each packet that is encoded. If the number of buffers free is smaller than the number allocated, encoding is in process.

Total # of Requests processed
This value shows the total number of buffers that have been processed by the encoding engine. This number corresponds roughly to the total number of packets that have been compressed or encrypted by all the clients of the system since the last router restart or reload.

summary

This command displays a summary of the system. It is a composite command that combines the output from the **list status**, **list devices**, and **list ports** commands.

Example:

```
list summary

Encoding System Status
----------------------

Last Error:                             14 (Stream not active)

Internal Condition flags:               0x00000001   -->
                                         Ready

Number of Ports:                        2

Global Request pool size:               Alloc: 32  Free: 32
Total # of Requests processed:          7059



                        Encoding System Devices
 Encoding System Devices

Device Type                        Slot/Port  Status
------ --------------------------  ---------  ----------
     1 Host Software                  0/0     Ready
     0 Null Device                    0/0     Ready


                        Encoding System Ports
                        ---------------------

                                  +--Encoder State---+   +--Decoder State---+
Port  User                        QLen  Status           QLen  Status
----  ------------------------    ----  --------------   ----  --------------
   1  Net 2   (PPP/0)               0  Idle               0  Idle
   2  Net 3   (PPP/1)               0  Idle               0  Idle
```

Monitoring ES

Chapter 11. Configuring and Monitoring Data Compression

This chapter discusses data compression on a 2210 over Frame Relay and PPP interfaces. It includes these sections:

Data compression is supported on Frame Relay and PPP interfaces.

Data Compression Overview

The data compression system provides a means to increase the effective bandwidth of networking interfaces on the device. It is primarily intended for use on slower speed WAN links.

Data compression on the device is supported on PPP and Frame Relay interfaces:

- For PPP interfaces, compression is implemented according to the Compression Control Protocol (CCP) as defined in the Internet Engineering Task Force's RFC 1962. CCP provides the underlying mechanisms by which the use of compression is negotiated and a means for choosing among multiple possible compression protocols.

  The device provides two compression protocols: the Stac-LZS protocol, defined in RFC 1974; and the Microsoft Point-to-Point Compression protocol (MPPC), described in RFC 2118. Both of these are based on compression algorithms provided by Stac Electronics.
- For Frame Relay interfaces, compression is implemented according to FRF.9, the *Data Compression over Frame Relay Implementation Agreement* produced by the Frame Relay Forum Technical Committee. FRF.9 describes a Data Compression Protocol (DCP), modeled after PPP's CCP, and similarly provides a means for negotiating various compression algorithms and options. The device supports DCP "mode 1" negotiation. FRF.9 also describes a more generalized "mode 2"; this is not supported. Compression itself is done using the same compression engine as used for the PPP Stac-LZS protocol.

Data Compression Concepts

Data compression on the device provides a means to increase throughput on network links by making more efficient use of the available bandwidth on a link. The basic principle behind this is simple: represent the data flowing across a link in as compact a manner as possible so that the time needed to transmit it is as low as possible, given a set speed on a link.

Data compression may be performed at many layers in the networking model. At one end of the spectrum, applications may compress data prior to transmitting it to peer applications elsewhere in the network, while at the other end of the spectrum devices may be performing compression at the data link layer, working purely on the bit stream passing between two nodes. How this compression is done and how

effective it is depends on a variety of factors, including such things as what network layer the compression is performed at, how much intrinsic knowledge the compressor and decompressor have about the data being compressed, the compression algorithm chosen, and the actual data being compressed. The best compression can usually be performed at the application layer; for example, a file transfer application usually has the luxury of having an entire file of data available to it prior to attempting compression, and it may be able to try different compression algorithms on the file to see which performs best on that particular file's data. Although this may provide excellent compression for that one type of application, it does little to solve the general problem of compressing the bulk of the traffic flowing over a network, as most networking applications do not currently compress data as they generate it.

Compression on the device takes place at a much lower networking layer, at the data link layer. In the device, compression is performed on the individual packets which are transmitted across a link. The compression is done in real-time as packets flow through the device: the sender compresses a packet just prior to transmitting it, and the decompressor decompresses the packet as soon as it receives it. This operation is transparent to the higher layer networking protocols.

Data Compression Basics

Data compressors work by recognizing "redundant" information in data, and producing a different set of data which contains as little redundancy as possible. "Redundant" information is any information which can be derived and recreated based on the currently available data. For example, a compressor might function by recognizing repeated character patterns in a data stream and replacing these repeated patterns with a shorter code sequence to represent that pattern. As long as the compressor and decompressor agree on what these code sequences are then the decompressor can always recreate the original data from the compressed data.

This mapping of sequences in the original data to corresponding sequences in the compressed output is commonly called a ***data dictionary***. These dictionaries may be statically defined - experienced-based information available to the compressor and decompressor - or they may be dynamically generated, usually based on the information being compressed. Static dictionaries are most applicable to environments where the data being processed is of a limited, known nature, and not very effective for general-purpose compressors. Most compression systems use dynamic dictionaries, including any compressors used on the device. On a 2210 the data dictionaries are based on the current packet being processed and possibly previously seen packets, but there is no ability to "look ahead" in the data stream as may exist when compression is performed at other layers. For systems where the data dictionary is dynamically derived and based only on previously seen data, the dictionary is also commonly known as a ***history***. The terms history and data dictionary will be used interchangeably throughout the remainder of this chapter, though it should be understood that in other environments a history is a specific form of data dictionary.

The fact that the device uses dynamic dictionaries and that the compressor and decompressor must keep their dictionaries in synchronization means that data compression works on a stream of data passing between two endpoints. Hence, compression on the router is a connection-oriented process, where the endpoints of the connection are the compressor and decompressor themselves. When compression is started on the stream, both ends reset their data dictionaries to some known starting state, and then they update that state as data is received.

Compression could be performed on each individual packet, resetting the histories prior to processing each packet. Normally though, the data dictionaries are not reset between packets, which means that the histories are based not only on the contents of the current packet, but also the contents of previously seen packets. This usually improves the overall compression effectiveness, because it increases the amount of data which the compressor searches looking for redundancy to remove. As an example, consider the case of one host “pinging” another host with IP: a series of packets is sent out, each one usually nearly identical to the last one sent. The compressor may have little luck compressing the first packet, but it may recognize that each subsequent packet looks very much like the last one sent, and produce highly compressed versions of those packets.

Because the compressor and decompressor histories change with each packet received, the compression mechanisms are sensitive to lost, corrupted, or reordered packets. The compression protocols employed by the device include signalling mechanisms whereby the compressor and decompressor can detect loss of synchronization and resynchronize to each other, such as might be necessary when a packet is lost due to a transmission error. Typically this is done by including a sequence number in each packet which the decompressor will check to make sure it is receiving all packets, in order. If it detects an error, it will reset itself to some known starting state, signal the compressor to do likewise, and then wait (discarding incoming compressed packets) until the compressor acknowledges that it has also reset itself.

Compression on a link typically is performed on data going in both directions over the link. Normally, each end of a connection has both a compressor and decompressor running on it, communicating with their analogs at the other end of the connection, as shown in Figure 10 on page 146. The output (compression) side runs independently of the input (decompression) side. It is possible for completely different compression algorithms to be operating for each direction of the link. When a link connection is established, the compression control protocol for the link will negotiate with the peer to determine the compression algorithms used for the connection. If the two ends cannot agree on compression protocols to use, then no compression will be performed and the link will operate normally - packets will simply be sent in uncompressed form.

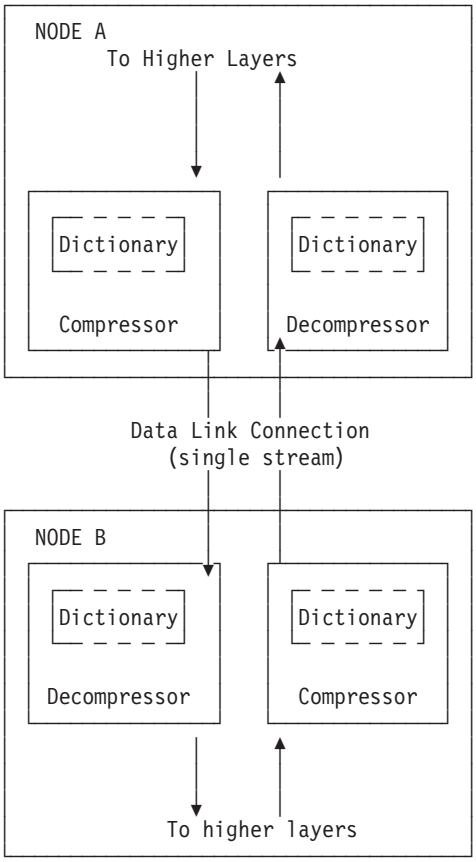


Figure 10. Example of Bidirectional Data Compression with Data Dictionaries

A stream really represents a connection between a specific compression process on one end of a link and an associated decompression process on the other end of a link, and thus is more specific than just a “connection” between two nodes; it is possible that a sophisticated compression protocol could split the data flowing between two hosts into multiple streams, compressing each of these streams independently. For example, PPP’s CCP has the ability to negotiate the use of multiple histories over a single PPP link, though the router does not support this.

Considerations

The choice of whether or not to use data compression is not always an easy one. There are several factors which should be considered before enabling compression on a connection.

CPU Load

Data compression is a computationally expensive procedure. As the amount of data being compressed increases (per unit time), the more of a load is put on the device’s processor. If the load becomes too great, the performance of the device degrades - on all network interfaces, not just the ones where compression is being performed.

The device actually contains multiple processors and uses asymmetric multiprocessing - for example, link I/O controllers which operate in tandem with the main processor - so the effect of the processor loading is not always readily

measured. Because the compression operation may be overlapped with the transmission of packets, this loading may in fact be totally transparent and pose no problem. Nonetheless, it is possible to overburden the device's processor and degrade performance.

As a general rule of thumb, compression should only be enabled on slow speed WAN links - probably only for links with speeds up to about 64 kilobits per second (the speed of a typical ISDN dial link). The total bandwidth for data being compressed on all links probably should be limited to several hundred kilobits per second. Running compression on all channels of an ISDN Primary Rate adapter would be unwise.

The Encoding Subsystem parameters allow you to limit the number of connections which may be concurrently running compression. More interfaces can be enabled for compression then than are actually running it. Once the limit on the number of active compression connections is reached, additional connections will simply not negotiate the use of compression, at least not until an existing compression link shuts down.

Memory Usage

Another issue to consider when configuring compression is the memory requirement. Compression and decompression histories occupy a fair amount of memory, which is a limited resource in the device. The Stac-LZS algorithm for example requires about 16 KB for a compression history, and about 8 KB for a decompression history. This problem is magnified by the fact that these histories must exist for each connection which is established: a compression history is synchronized with a corresponding decompression history in a peer router. For a PPP link, this implies one compression history and one decompression history (assuming that data compression is running bidirectionally on the link). On a Frame Relay link, there could be many such histories required, one pair for each virtual connection (DLCI) which is established.

The device creates a pool of compression and decompression histories when it boots. These are always allocated in pairs known as ***compression sessions*** - a session is simply one compression history coupled with one decompression history. Technically, compression and decompression are independent functions, but in practice compression is almost always run bidirectionally and so memory is managed and configured in terms of sessions rather than individual histories as a way of simplifying operation. Since different compression algorithms have different memory requirements for compression and decompression, the session is sized to approximately 30 KB to handle the worst case. The pool of compression sessions is populated as configured in the Encoding Subsystem feature. See "Chapter 10. Configuring and Monitoring the Encoding Subsystem" on page 135 for details.

Whenever the device attempts to establish a compression connection on a link, it begins by reserving a session from the allocated pool of sessions. If no sessions are available, then compression is not performed on that connection. The router may attempt to start compression on that connection later as sessions become available.

The number of compression sessions which are allocated is a configurable parameter. Setting the number of sessions allocated limits both the amount of memory used and the maximum number of connections which may be

simultaneously operating with compression. Limiting the number of simultaneously operating compression connections provides a means to help control the CPU loading problem.

Data Content

The actual nature of the data being transmitted on a connection should be considered before enabling compression for that connection. Compression works better on some types of data than others. Packets which contain a lot of nearly identical information - for example a set of packets generated from an IP “ping” - will normally compress extremely well. A typical assortment of random text and binary data going over a link will usually compress in ratios around 1.5:1 to 3:1. Some data simply will not compress well at all. In particular, data which has already been compressed will seldom compress further. In fact, data which has been previously compressed may actually expand when fed through the compression engine.

If it is known in advance that most of the data flowing over a connection will consist of compressed data, then it is recommended that compression not be enabled for that connection. An example where this might occur is a connection to a host which was set up to be primarily a FTP file archive site, where all the files available to be transferred are stored in compressed form on the host.

Link Layer Compression

A final factor to consider is the nature of the network link between the two hosts. Compression could be performed at a lower layer than even the device’s hardware interfaces. In particular, many modern modems incorporate data compression mechanisms in their hardware and firmware. If compression is being performed on the link at a lower layer (outside the device), then it is best not to enable data compression on the device for that interface. As already mentioned, compressing an already compressed data stream is normally ineffective, and in fact may degrade performance slightly. Unless there is some particular reason to believe that the router will do a much better job of compression than the link hardware, it is best to let the link hardware do the compression.

Configuring and Monitoring Data Compression on PPP Links

The 2210 uses the PPP Compression Control Protocol (CCP) to negotiate the use of compression on a link. CCP provides a generalized mechanism to negotiate the use of a particular compression protocol, possibly even using a different protocol in each direction of the link, and various protocol-specific options. The software supports the Stac-LZS and MPPC protocols, so the peer must also provide support for at least one of these algorithms to successfully negotiate data compression between the two nodes. The two nodes must also agree on the algorithm-specific options for compression to operate.

Configuring Data Compression on PPP Links

To configure data compression on PPP links:

1. Enable the CCP protocol on the link with the **enable ccp** command. This enables the link to negotiate compression with the other node. Negotiation includes what compression algorithm to use and any protocol-specific options.
2. Select which compression algorithms may be negotiated using the **set ccp algorithms** command.

3. Set the negotiable parameters for each compression algorithm using the **set ccp options** command.

You can display the current compression configuration using the **list ccp** command.

Table 18 lists the available commands and Figure 11 is an example of configuring compression on a PPP link. For detailed descriptions of these commands, see 'Point-to-Point Configuration Commands' in *Software User's Guide*.

Table 18. PPP Data Compression Configuration Commands

Data Compression Command	Action
disable ccp	Disables data compression.
enable ccp	Enables data compression.
set ccp options	Sets options for the compression algorithm.
set ccp algorithms	Specifies a prioritized list of compression algorithms.
list ccp	Displays compression configuration.

```
Config>net 6  1
PPP 6 Config>enable ccp
PPP 6 Config>set ccp alg  2
Enter a prioritized list of compression algorithms (first is preferred),
all on one single line.
Choices (can be abbreviated) are:
STAC-LZS MPPC
Compressor list [STAC-LZS]? stac mppc
PPP 6 Config>set ccp options
STAC: check mode (0=none, 1=LCB, 2=CRC, 3=Seq, 4=Ext) [3]?
STAC: # histories [1]?
PPP 6 Config>li ccp

CCP Options
-----------
Data Compression enabled
Algorithm list: STAC-LZS MPPC
STAC histories: 1
STAC check_mode: SEQ

MPPE Options
------------
MPPE disabled
Optional encryption
Key generation: STATEFUL
```

Figure 11. Example of Configuring Compression on a PPP Link

Notes:

1. The network command selects the network interface for the PPP link. If the link is a PPP dial circuit, you must then use the **encapsulator** command to access the PPP configuration menu.
2. If you enable CCP and do not set algorithms for the link, the software automatically sets the link to use protocols STAC and MPPC as if you had entered the command **set ccp algorithms stac mppc**.

   If you set multiple algorithms, the order of the algorithms determines the negotiation preference for the link.

   Certain dial-in client implementations may not be able to connect if the router supports multiple compression protocols on one link. If you encounter this, set the ccp protocol to either STAC or MPPC.

   If you enter **set ccp algorithms none**, the software will automatically disable compression on the link.

   If MPPE is enabled and CCP is enabled, MPPC is the compression algorithm.

Monitoring Data Compression on PPP Links

You monitor compression as you would other PPP components. ‘Accessing the Interface Monitoring Process’ in *Software User’s Guide* describes how to access the PPP console environment and details about the commands. Table 19 lists the compression-related commands. Figure 12 shows an example of listing compression on a PPP interface.

Table 19. PPP Data Compression Monitoring Commands

Command	Function
list control ccp	Lists CCP state and negotiated options.
list ccp	Lists CCP packet statistics.
list cdp or **list compression**	Lists compressed datagram statistics.

```
+ network 1
PPP > list control ccp

CCP State:                Open
Previous State:           Ack Sent
Time Since Change:        2 minutes and 52 seconds

Compressor:   STAC-LZS histories 1, check_mode SEQ
Decompressor: STAC-LZS histories 1, check_mode SEQ
MPPE:         Not negotiated

PPP > list ccp

CCP Statistic             In                      Out
-------------             --                      ---

Packets:                  2                       3
Octets:                   18                      27
Reset Reqs:               0                       0
Reset Acks:               0                       0
Prot Rejects:             1                       -

PPP > list cdp

Compression Statistic     In                      Out
---------------------     --                      ---

Packets:                  19541                   19542
Octets:                   2550673                 2740593
Compressed Octets:        821671                  899446
Incompressible Packets:   0                       0
Discarded Packets:        0                       -
Prot Rejects:             0                       -
Compression Ratios:       3.11                    3.24
```

Figure 12. Monitoring Compression on a PPP Interface

Configuring and Monitoring Data Compression on Frame Relay Links

After configuring the global compression parameters and enabling compression on the interface, you must then set the parameters for each individual circuit (PVC) on the Frame Relay interface. Each circuit defined for the interface may have compression enabled on the circuit, and each circuit which successfully negotiates the use of compression uses one compression session from the global pool. You can also disable compression on the interface which means none of the circuits on that interface will be eligible to carry compressed data traffic.

Configuring Data Compression on Frame Relay Links

To configure data compression on FR links:

1. Enable compression on the interface using the **enable compression** command. This enables the link to negotiate compression with the other node.
2. Enable compression on each new PVC that will carry compressed data with the **add permanent-virtual-circuit** command. You can change existing PVCs using the **change permanent-virtual-circuit** command.

You can display the current compression configuration using the **list lmi** or **list permanent-virtual-circuit** commands.

Table 20 on page 152 lists the commands available for configuring compression on a Frame Relay link and Figure 13 on page 152 is an example of configuring a Frame Relay Link. See “Frame Relay Configuration Commands” in *Software User’s Guide* for details.

```
Config> net 2

Frame Relay user configuration

FR Config> enable compression
Maximum number of run-time compression circuits (zero means no limit) [0]? 0
Do you want orphan PVCs to perform compression [Y]? n
The number of currently defined non-compression PVCs is 4
Would you like to change them all to compression PVCs [N]? y

FR Config> add perm

Circuit number [16]? 22
Committed Information Rate (CIR) in bps [65536]?
Committed Burst Size (Bc) in bits [64000]?
Excess Burst Size (Be) in bits [0]?
Assign circuit name []? cir22
Is circuit required for interface operation [N]?
Do you want to have data compression performed [Y]?

FR Config>list lmi

                        Frame Relay Configuration


  LMI enabled            =      No  LMI DLCI                       =       0
  LMI type               =    ANSI  LMI Orphans OK                 =     Yes
  CLLM enabled           =      No  Timer Ty seconds               =      11

  Protocol broadcast     =     Yes  Congestion monitoring          =     Yes
  Emulate multicast      =     Yes  CIR monitoring                 =      No
  Notify FECN source     =      No  Throttle transmit on FECN      =      No

  Data compression       =     Yes  Orphan compression             =      No
  Compression PVC limit  =    None  Number of compression PVCs     =       2

  PVCs P1 allowed        =      64  Interface down if no PVCs      =      No
  Timer T1 seconds       =      10  Counter N1 increments          =       6
  LMI N2 error threshold =       3  LMI N3 error threshold window =       4
  MIR % of CIR           =      25  IR % Increment                 =      12
  IR % Decrement         =      25  DECnet length field           =      No
  Default CIR            =   65536  Default Burst Size            =   64000
  Default Excess Burst   =       0

FR Config>list perm

  Maximum PVCs allowable =      64
  Total PVCs configured  =       2

             Circuit                Circuit   Circuit     CIR    Burst  Excess
              Name                  Number     Type      in bps  Size   Burst
---------------------------------- ------- ----------- ------- ------- -----
 circ16                              16    @ Permanent   65536   64000       0
 cir22                               22    @ Permanent   65536   64000       0

* = circuit is required
# = circuit is required and belongs to a required PVC group
@ = circuit is data compression capable
```

Figure 13. Example of Configuring Compression on a Frame Relay Link

Table 20. Data Compression Configuration Commands

Command	Action
add permanent-virtual-circuit #	Use to enable data compression on a specific PVC defined on an interface.
change permanent-virtual-circuit #	Use to change whether a specific PVC will compress data.
disable compression	Disables data compression.
enable compression	Enables data compression.
list lmi	Displays the current configuration of the interface.
list permanent	Lists summary information about circuits.

Note: Enabling compression on orphan circuits will decrease the number of available compression sessions available for the native PVCs on the device.

If you enable compression on a Frame Relay interface that already has compression enabled, the software asks you if you want to change compression parameters on the interface, as shown in the following example. You can change compression on the interface without disabling compression.

Example of changing compression on Frame Relay interfaces:

```
Config> net 2

Frame Relay user configuration

FR Config> enable compression
Data compression already enabled.
Do you wish to continue and change an interface parameter [Y]
Maximum number of run-time compression PVCs (zero means no limit) [0]? 32
Do you want orphan circuits to perform compression [Y]?
The number of currently defined circuits is 5
Change all of these circuits to perform compression?
```

Monitoring Data Compression on Frame Relay Links

You monitor compression as you would other Frame Relay components. “Frame Relay Monitoring Commands” in *Software User’s Guide* describes how to access the Frame Relay console environment and details about the commands. Table 21 lists the compression-related commands. “Example: Monitoring Compression on a Frame Relay Interface or Circuit” shows an example of listing compression on a Frame Relay interface.

Table 21. Frame Relay Data Compression Monitoring Commands

Command	Display
list lmi	Lists the current status of the interface.
list permanent	Lists summary information about circuits.
list circuit	Lists the current status of a circuit.

Example: Monitoring Compression on a Frame Relay Interface or Circuit

```
+ network 2
FR 2 > list lmi

Management Status:
  ------------------

    LMI enabled          =      No  LMI DLCI                   =       0
    LMI type             =    ANSI  LMI Orphans OK             =     Yes
    CLLM enabled         =      No

    Protocol broadcast   =     Yes  Congestion monitoring      =     Yes
    Emulate multicast    =     Yes  CIR monitoring             =      No
    Notify FECN source   =      No  Throttle transmit on FECN  =      No
    PVCs P1 allowed      =      64  Interface down if no PVCs  =      No
    Line speed (bps)     =   64000  Maximum frame size         =    2048
    Timer T1 seconds     =      10  Counter N1 increments       =       6
    LMI N2 threshold     =       3  LMI N3 threshold window    =       4
    MIR % of CIR         =      25  IR % Increment             =      12
    IR % Decrement       =      25  DECnet length field        =      No
    Default CIR          =   65536  Default Burst Size         =   64000
    Default Excess Burst =       0

    Current receive sequence  =       0
    Current transmit sequence =       0
    Total status enquiries    =       0  Total status responses   =       0
    Total sequence requests   =       0  Total responses          =       0
```

```
  Data compression enabled  =     Yes  Orphan Compression       =      No

  Compression PVC limit     =    None  Active compression PVCs  =       1

PVC Status:
-----------

  Total allowed  =      64  Total configured  =       1
  Total active   =       1  Total congested   =       0
  Total left net =       0  Total join net    =       0


FR 2 > list permanent

Circuit                                  Orphan  Type/   Frames      Frames
Number           Circuit Name            Circuit State Transmitted  Received
------- -------------------------------- ------- ----- ----------- --------

   16  circ16                              No   @ P/A        58364      58355
   22  circ22                              No   & P/A        58364      58355

  A - Active   I - Inactive   R - Removed   P - Permanent   C - Congested
  * - Required                # - Required and belongs to a PVC group
  @ - Data compression capable but not operational
  & - Data compression capable and operational


FR 2 > list circuit 22

Circuit name = circ22

    Circuit state        =    Active  Circuit is orphan   =          No
    Frames transmitted   =     58391  Bytes transmitted   =     2676894
    Frames received      =     58383  Bytes received      =     2671009
    Total FECNs          =         0  Total BECNs         =           0
    Times congested      =         0  Times Inactive      =           0
    CIR in bits/second   =     65536  Potential Info Rate =       64000
    Committed Burst (Bc) =     64000  Excess Burst (Be)   =           0
    Minimum Info Rate    =     16000  Maximum Info Rate   =       64000
    Required             =        No  PVC group name      =   Unassigned

    Compression capable  =       Yes  Operational         =         Yes
    R-R's received       =         0  R-R's transmitted   =           0
    R-A's received       =         0  R-A's transmitted   =           0
    R-R mode discards    =         0  Enlarged frames     =           0
    Decompress discards  =         0  Compression errors  =           0
    Rcv error discards   =         0

    Compression ratio    =  1.00 to 1  Decompression ratio =    1.00 to 1

    Current number of xmit frames queued                  =           0
    Xmit frames dropped due to queue overflow             =           0
```

Chapter 12. Using Local or Remote Authentication

Authentication is the process of determining who a user (or entity) is. Authenticating user access for the PPP protocol on the 2210 extends the flexibility of user profile management as it relates to PPP authentication protocols PAP, MSCHAP, CHAP, and SPAP. See 'PPP Authentication Protocols' in *Software User's Guide* for additional information about configuring PAP, MSCHAP, CHAP, and SPAP.

Authentication can be configured locally or can be configured to consolidate user configuration using authentication servers that are available on the network to service authentication requests for the entire network. The IBM 2210 implements locally maintained authentication as well as the following authentication server protocols:

- Radius
- TACACS
- TACACS+

Using Authentication, Authorization, and Accounting (AAA) Security

Authentication, Authorization, and Accounting (AAA) Security are configurable protocols that allow you to control access to your services. You can configure AAA to perform for local or remote authentication.

You can configure a security protocol for three types of functions.

- PPP links
- Login users (Telnet/Console Login)
- Tunnels

The configuring is done by setting a primary and secondary server. The server information is configured and stored separately from the AAA configuration. You use a server profile by a name that is provided at configuration time.

Under all circumstances accounting cannot be done locally and must be either Radius or TACACS+.

Authorization can only be done locally, or through remote authentication that uses Radius or TACACS+.

What is AAA Security?

AAA Security is the name of the security system for this device. It includes:

Authentication
: The process of identifying a user. Authentication utilizes a name and a password for access.

Authorization
: The process of determining the services to which a user is allowed access. Authorization processing might find that the user is not authenticated. The authorization agent then determines whether an unauthenticated user is allowed access to the services in question.

Accounting
The process of recording when a user has started or stopped a session. There are two types of accounting records supported.

Start records
Indicates that a service is about to begin.

Stop records
Indicates that a service has ended.

Using PPP

For the Point-to-Point Protocol (PPP) you can configure the following:

- Authentication
- Authorization
- Accounting

Each function can have its own security protocol that you configure independently.

- Setting the authentication protocol will have no effect on authorization or accounting.
- Setting the authorization protocol will have no effect on authentication or accounting.
- Setting the accounting protocol will have no effect on authentication or authorization.
- Setting AAA to remote will set authentication to remote, authorization to remote and set accounting to remote.
- Setting AAA to local will set authentication to local, authorization to local, and set accounting to ignore. You cannot disable authentication or authorization.

See Point-to-Point Configuration Commands in *Software User's Guide* for details about the PPP configuration commands that you use in this environment.

Valid PPP Security Protocols

The following are valid PPP security protocols:

Authentication Methods
Local, RADIUS, TACACS+, TACACS

Authorization Methods
Local, RADIUS, TACACS+

Accounting Methods
RADIUS, TACACS+

Table 22. Set PPP Security Protocols

Action	Authent	Author	Acct
set AAA local	local	local	ignore
set AAA remote	remote	remote	remote
set AUTHENT local	local	ignore	ignore
set AUTHOR local	ignore	local	ignore
set AUTHENT remote	remote	ignore	ignore
set ACCOUNTING local	n/a	n/a	n/a
set AUTHOR remote	ignore	remote	ignore

Table 22. Set PPP Security Protocols (continued)

Action	Authent	Author	Acct
set ACCOUNTING remote	ignore	ignore	remote
disable ACCOUNTING	ignore	ignore	disabled
disable AUTHENT	n/a	n/a	n/a
disable AUTHOR	n/a	n/a	n/a

Using Login

For AAA login configuration, either remote or local can be selected. If local authentication is desired, then Local authorization must also be used. If remote authentication is selected, then, remote authorization must be used. accounting is not supported locally, so when authenticating and authorizing locally you must disable accounting.

Attention: Before enabling console login, save the configuration with console login disabled. If login authentication is set to a remote server using Radius, TACACS, or TACACS+ and the router is unable to reach the authentication server, then access to the router is denied. Disabling the console login prevents a lockout situation.

When configuring remote authentication, you can set authorization to another remote authorization protocol Radius or TACACS+, and set accounting to use Radius or TACACS+.

- Setting AAA to local sets authentication to local, authorization to local, and accounting to disabled.
- Setting AAA to remote sets authentication to remote, authorization to remote, and accounting to remote.
- Setting the authentication protocol to local automatically sets the authorization protocol to the same and disables accounting.
- Setting the authentication protocol to remote automatically sets the authorization protocol to the same only if the authorization protocol is set to local and ignores the accounting protocol.
- Setting the authorization protocol to remote automatically sets the authentication protocol to the same only if the authentication protocol is set to local and ignores the accounting protocol.
- Setting the accounting protocol to remote automatically sets authentication protocol to same only if the authentication protocol is set to local, and sets the authorization protocol to the same only if authorization is set to local.
- Setting the accounting protocol to disable has no effect on the authentication or authorization protocol.
- Disabling authentication or authorization is not allowed.

Valid Login/Admin Security Protocols

The following are valid Login/Admin security protocols.

Authentication/Authorization Methods
Local, RADIUS, TACACS Plus

Accounting Methods
RADIUS, TACACS Plus

Using Local or Remote Authentication

Table 23. Set Login Security Protocols

Action	Authent	Author	Acct
set AAA local	local	local	disabled
set AAA remote	remote	remote	remote
set AUTHENT local	local	local	disabled
set AUTHOR local	local	local	disabled
set AUTHENT remote	remote	remote, if local else ignore	ignore
set AUTHOR remote	remote, if local else ignore	remote	ignore
set ACCOUNTING remote	remote, if local else ignore	remote, if local else ignore	remote
disable ACCOUNTING	ignore	ignore	disabled
disable AUTHEN	n/a	n/a	n/a
disable AUTHOR	n/a	n/a	n/a

Using Tunnels

Set tunnel authentication the same as tunnel authorization. When you set tunnel authentication to either local or remote, you can then enable accounting. The tunnel authorization and authentication server must be the same.

Valid Tunnel Security Protocols

The following are valid Tunnel security protocols:

Authentication/Authorization Methods
Local, RADIUS

Accounting Methods
RADIUS, TACACS Plus

Table 24. Set Tunnel Security Protocols

Action	Authent	Author	Acct
set AAA local	local	local	ignore
set AAA remote	remote	remote	remote
set AUTHENT local	local	local	ignore
set Author local	local	local	ignore
set AUTHENT remote	remote	remote	ignore
set AUTHOR remote	remote	remote	ignore
set ACCOUNTING remote	ignore	ignore	remote
disable ACCOUNTING	ignore	ignore	disabled
disable AUTHENT	n/a	n/a	n/a
disable AUTHOR	n/a	n/a	n/a

Password Rules

Local authentication allows you to use a password to control login access. The password can be checked against any or all of the following rules.

Note: The following rules only apply for PPP user login and not console login.

- Be a minimum number of characters in length. You set the number of characters required.
- Contain at least one alphabetic character.
- Contain at least one non-alphabetic character.
- Contain a non-numeric character in the first position.
- Contain a non-numeric character in the last position.
- Contain no more than three identical consecutive characters that were used in the previous password.
- Contain no more than two consecutive characters.
- Not contain the userid as a part of the password.
- Not the same as any of the previous three passwords.
- Be changed after a certain number of days. You set the number of days between password changes.
- Locked out after a specific number of login failures. You set the number of failures.

Understanding Authentication Servers

An ***authentication server*** is a server in the network that validates userids and passwords for the network. If a device is configured for authentication through an authentication server and the device receives a packet from an authentication protocol, the device passes a userid and password to the server for authentication. If the userid and password are correct, the server responds positively. The device can then communicate with the originator of the request. If the server does not find the userid and password that it receives from the device, it responds negatively to the device. The device then rejects the session from which it got the authentication request.

SecurID Support

The 2210 can authenticate dial-in clients that use SecurID with a Security Dynamics ACE/Server. This support uses TACACS, TACACS+, or RADIUS on the ACE/Server for authentication of the client. Configure the dial-in client the same as other dial-in clients on the 2210.

The dial-in client logs on as usual, but uses the SecurID passcode for the password. The SecurID passcode consists of a 4 to n-digit PIN number that is followed by the number from the SecurID token card. (The maximum number of digits in the PIN depends on the server.) The userid and password could appear as:

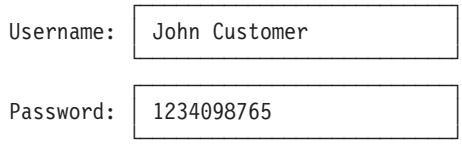


Figure 14. SecurID Username and Passcode

When the ACE/Server authenticates the logon, it may request the next token from the client. The next token is the next token on the token card. The maximum

number of digits in the next token depends on the SecurID token card the client is using. The client can enter the passcode and the next token when prompted for the password by using the format `passcode*token` as in the following:

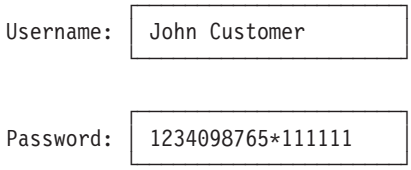


Figure 15. SecurID Passcode with Next Token

Note: When the server requests the client to enter the next token, the client must:

1. Enter the PIN
2. Wait for a new token from the card and enter that token
3. Enter * followed by the next token from the card

The ACE/Server administrator configures the conditions that cause the server to request the next token or new PIN.

The dial-in clients should use SPAP so they can receive alerts from the authentication system when they need to enter the next token. If the client is not using SPAP and they are not successful logging on, they should try entering a new passcode using the `passcode*token` format. If the client is still not successful, there could be other problems between the client and the ACE/Server.

Limitations

The following limitations exist:

- Security Dynamics Inc. (SDI) and DES encryption are not supported.
- The SecurID “New PIN” function is not supported.
- TACACS does not support the “New PIN” or “Next-Token” functions. The client can specify a next-token when logging in, but the server will not use it.
- Clients configured for callback are not supported.
- When using CHAP with TACACS or TACACS+, set the CHAP rechallenge interval to 0.
- Do not use CHAP when using RADIUS authentication.
- Your clients can obtain the best results by using TACACS+ and SPAP.
- Windows 3.1 DIALs client with SecurID authentication using multilink is not supported.
- When using SecurID authentication, it is highly recommended to use the latest client software (for example, Windows 95 or OS/2).

Chapter 13. Configuring Authentication

This chapter describes the configuration and operational commands for authentication. It includes the following sections:

- "Accessing the Authentication Configuration Prompt"
- "Authentication Configuration Commands"

Accessing the Authentication Configuration Prompt

To access the `Authent config >` prompt:

1. Enter **talk 6** at the `*` prompt.
2. Enter **feature auth** at the `Config >` prompt.

Authentication Configuration Commands

Table 25 lists the commands available at the `Authent config >` prompt.

Table 25. Authentication Configuration Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page xxix.
Disable	Disables accounting for AAA.
List	Displays the AAA configuration parameters.
Login	Configures AAA for login.
Nets-info	Displays information about local PPP authentication.
Password-rules	Configures password rules (enables or disables).
PPP	Configures AAA for PPP.
Quickset	Configures the authentication method quickly.
Servers	Configures individual remote AAA servers.
Set	Configures Authentication parameters regardless of type.
Tunnel	Configures AAA for L2TP tunnels.
User-profile	Configures local PPP users.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page xxix.

Disable

Use the **disable** command to disable accounting.

Syntax:

disable accounting

List

Use the **list** command to display the AAA parameters.

Syntax:

list accounting

authentication

authorization

all

config

```
AAA Config> list all
ppp AAA configuration...
 ppp authentication        : Radius        serv01
   authorizeAuthent          YES
   Primary server address    1.1.1.1
   Secondary server address  2.2.2.2
     Request tries           3
     Request interval        3
   Key for encryption        <notSet>
 ppp authorization         : locallist
 ppp accounting            : Disabled
tunnel AAA configuration...
 tunnel authentication     : Radius        serv01
   authorizeAuthent          YES
   Primary server address    1.1.1.1
   Secondary server address  2.2.2.2
     Request tries           3
     Request interval        3
   Key for encryption        <notSet>
 tunnel authorization      : Radius        serv01
   authorizeAuthent          YES
   Primary server address    1.1.1.1
   Secondary server address  2.2.2.2
     Request tries           3
     Request interval        3
   Key for encryption        <notSet>
 tunnel accounting         : Disabled
login AAA configuration...
 login authentication      : Radius        serv01
   authorizeAuthent          YES
   Primary server address    1.1.1.1
   Secondary server address  2.2.2.2
     Request tries           3
     Request interval        3
   Key for encryption        <notSet>
 login authorization       : Radius        serv01
   authorizeAuthent          YES
   Primary server address    1.1.1.1
   Secondary server address  2.2.2.2
     Request tries           3
     Request interval        3
   Key for encryption        <notSet>
 login accounting          : Radius        serv01
   authorizeAuthent          YES
   Primary server address    1.1.1.1
   Secondary server address  2.2.2.2
     Request tries           3
     Request interval        3
   Key for encryption        <notSet>

AAA Config> list accounting all
accounting AAA configuration...
 accounting ppp            : Disabled
 accounting tunnel         : Disabled
 accounting login          : Radius        serv01
   authorizeAuthent          YES
   Primary server address    1.1.1.1
   Secondary server address  2.2.2.2
     Request tries           3
     Request interval        3
```

```
   Key for encryption        <notSet>
AAA Config> list accounting config
accounting ppp            : Disabled
accounting login          : Radius        serv01
accounting tunnel         : Disabled

AAA Config> list authentication all
authentication AAA configuration...
 authentication ppp        : Radius        serv01
   authorizeAuthent          YES
   Primary server address    1.1.1.1
   Secondary server address  2.2.2.2
     Request tries           3
     Request interval        3
   Key for encryption        <notSet>
 authentication tunnel     : Radius        serv01
   authorizeAuthent          YES
   Primary server address    1.1.1.1
   Secondary server address  2.2.2.2
     Request tries           3
     Request interval        3
   Key for encryption        <notSet>
```

Login

Use the **login** command to configure AAA for login.

Table 26 lists the subcommands available with the **login** command.

Table 26. Login Subcommands

Command	Function
Disable	Disables accounting for login.
List	Displays the AAA configuration parameters for login.
Set	Sets the AAA configuration parameters for login.

Disable

Use the **login disable** command to disable accounting.

Syntax:

login disable accounting

List

Use the **login list** command to display the AAA configuration parameters.

Syntax:

login list all
accounting
authentication
authorization
config

Set

Use the **login set** command to configure authentication parameters.

Syntax:

login set aaa
accounting
authentication
authorization

aaa *authtype*
Sets the authentication, authorization, and accounting type. *Authtype* is one of the following:

local Sets the authentication, authorization, and accounting type to use a locally-maintained user database.

remote
Sets the authentication, authorization, and accounting type to use a remote user database.

server id
Specifies the identifier of the remote database.

accounting *authtype*
Sets the accounting type. *Authtype* is one of the following:

remote
Sets the authentication type to use a remote user database.

server id
Specifies the identifier of the remote database.

authentication *authtype*
Sets the authentication type. *Authtype* is one of the following:

local Sets the authentication type to use a locally-maintained user database.

remote
Sets the authentication type to use a remote user database.

server id
Specifies the identifier of the remote database.

authorization *authtype*
Sets the authorization type. *Authtype* is one of the following:

local Sets the authorization type to use a locally-maintained user database.

remote
Sets the authorization type to use a remote user database.

server id
Specifies the identifier of the remote database.

Nets-info

Use the **nets-info** command to display the currently configured PPP authentication protocol on each PPP interface.

Syntax:

nets-info

Password-rules

Use the **password-rules** command to configure the password (enable or disable).

Table 27 lists the subcommands available with the **password-rules** command.

Table 27. Login Subcommands

Command	Function
Disable	Disables a password rule.
Enable	Enables a password rule.
List	Displays the current state of the password rules (enabled or disabled).

Disable

Use the **password-rules disable** command to disable any or all of the password rules.

Syntax:

password-rules disable all
compare-ident-prev
change-days
first-non-numeric
ident-chars
last-non-numeric
lockout
minimum-length
one-alpha
one-nonalpha
prev-three
userid-contained

compare-ident-prev
Compares the previous user identity with the user requesting a password change.

change-days
The maximum number of days before a password change is required.

Valid values: 0 to 360

Default value: 180

first_non-numeric
The first character of a password cannot be numeric.

Valid values: any non-numeric character

Default value: none

ident-chars
Cannot contain more than 3 characters used in a previous password in the same position.

last-non-numeric
The last character in the password cannot be numeric.

Valid values: any non-numeric character

Default value: none

lockout
The number of times you can try a password before you are locked out.

Valid values: 0 to 360

Default value: 3

minimum-length
The least number of characters required to have a valid password.

Valid values: 1 to 31

Default value: 8

maximum-length
The maximum number of characters a password can contain.

Valid values: 1 to 31

Default value: 8

one-alpha
At least one character in the password must be an alpha.

one-nonalpha
At least one character in the password must be numeric.

prev-three
The password cannot be the same as any of the last three passwords.

userid-contained
The password cannot contain the userid as a part of the password.

Enable

Use the **password-rules enable** command to enable any or all of the password rules. See the **disable** command for a list of password rule descriptions.

Syntax:

password-rules enable all
compare-ident-prev
change-days
first-non-numeric
ident-chars
last-non-numeric
lockout
minimum-length
one-alpha

one-nonalpha
prev-three
userid-contained

List

Use the **password-rules list** command to display the current state of the password rules (disabled or enabled).

Syntax:

password-rules list

PPP

Use the **ppp** command to configure AAA for PPP.

Table 28 lists the subcommands available with the **ppp** command.

Table 28. PPP Subcommands

Command	Function
Disable	Disables accounting for PPP.
List	Displays the AAA configuration parameters for PPP.
Set	Sets the AAA configuration parameters for PPP.

Disable

Use the **ppp disable** command to disable accounting for PPP.

Syntax:

ppp disable accounting

List

Use the **ppp list** command to display the AAA configuration parameters for PPP.

Syntax:

ppp list all
accounting
authentication
authorization
config

Set

Use the **ppp set** command to set the AAA configuration parameters for PPP.

Syntax:

ppp set aaa
accounting
authentication

authorization

aaa *authtype*
Sets the authentication, authorization, and accounting type. *Authtype* is one of the following:

local Sets the authentication, authorization, and accounting type to use a locally-maintained user database.

remote
Sets the authentication, authorization, and accounting type to use a remote user database.

server id
Specifies the identifier of the remote database.

accounting *authtype*
Sets the accounting type. *Authtype* is one of the following:

remote
Sets the authentication type to use a remote user database.

server id
Specifies the identifier of the remote database.

authentication *authtype*
Sets the authentication type. *Authtype* is one of the following:

local Sets the authentication type to use a locally-maintained user database.

remote
Sets the authentication type to use a remote user database.

server id
Specifies the identifier of the remote database.

authorization *authtype*
Sets the authorization type. *Authtype* is one of the following:

local Sets the authorization type to use a locally-maintained user database.

remote
Sets the authorization type to use a remote user database.

server id
Specifies the identifier of the remote database.

Servers

Use the **servers** command to configure individual remote AAA servers.

Table 29 lists the subcommands available with the **servers** command.

Table 29. Server Subcommands

Command	Function
Add	Adds a remote AAA server profile.
Change	Changes a remote server profile.
Delete	Deletes a remote server profile.
Lists	Displays the AAA server profile information.

Add

Use the **servers add** command to add a remote server profile.

Syntax:

servers add name

radius Sets the authentication type to use the radius authentication server protocol.

Values for the following parameters can be set:

key-for-encryption:
Specifies the encryption key.

Valid Values: Any alphanumeric character string up to 32 characters long.

Default Value: None.

primary-server-address:
Specifies the address of the primary authentication server.

Valid Values: Any valid IP address

Default Value: 0.0.0.0

retries

Valid Values: 1 to 100

Default Value: 3

retry-interval

Valid Values: 1 to 60

Default Value: 3

secondary-server-address:
Specifies the address of the secondary authentication server.

Valid Values: Any valid IP address

Default Value: 0.0.0.0

Author-Authent
Specifies whether authorization attributes are transferred during authentication.

Valid Values: yes, no

Default Value: yes

tacacs
Sets the authentication type to use the TACACS authentication server protocol.

Values for the following parameters can be set:

primary-server-address:
Specifies the address of the primary authentication server.

Valid Values: Any valid IP address

Default Value: 0.0.0.0

retries

Valid Values: 1 to 100

Default Value: 3

retry-interval

Valid Values: 1 to 60

Default Value: 3

secondary-server-address:
Specifies the address of the secondary authentication server.

Valid Values: Any valid IP address

Default Value: 0.0.0.0

tacacsplus
Sets the authentication type to use the TACACS+ authentication server protocol.

Values for the following parameters can be set:

encryption:
Specifies whether encryption will be used.

Valid Values: yes, no

Default Value:

key-for-encryption:
Specifies the encryption key to be used.

Valid Values: Any 16-hexadecimal digit value

Default Value:

primary-server-address:
Specifies the address of the primary authentication server.

Valid Values: Any valid IP address

Default Value: 0.0.0.0

privilege-level

Valid Values: 0 through 15

Default Value: 0

restarts
Sets the number of restarts. This parameter does not include timeout restarts and only pertains to restarts requested by the server.

Valid Values: 0 to 3200

Default Value: 0

time-to-connect
The amount of time to allow to obtain the authentication from the server.

Valid Values: 1 to 60

Default Value: 9

secondary-server-address:
Specifies the address of the secondary authentication server.

Valid Values: Any valid IP address

Default Value: 0.0.0.0

Change

Use the **servers change** command to change a remote server profile. See the **add** command for the remote server profile descriptions.

Syntax:

servers change	radius
	tacacs
	tacacsplus

See the **servers add** command for remote server profile descriptions.

Delete

Use the **servers delete** command to delete a remote server profile. See the **add** command for the remote server profile descriptions.

Syntax:

servers delete	radius
	tacacs
	tacacsplus

See the **servers add** command for the remote server profile descriptions.

List

Use the **servers list** command to display the AAA server profile information.

Syntax:

servers list	all
	names
	profile

Set

Use the **set** command to set the parameters for login, PPP, and L2TP tunnel.

Syntax:

set	aaa
	accounting
	authentication
	authorization

aaa *authtype*

Sets the authentication, authorization, and accounting type. *Authtype* is one of the following:

local Sets the authentication, authorization, and accounting type to use a locally-maintained user database.

remote
Sets the authentication, authorization, and accounting type to use a remote user database.

server id
Specifies the identifier of the remote database.

accounting *authtype*
Sets the accounting type for login, PPP and tunnel. *Authtype* is one of the following:

remote
Sets the authentication type to use a remote user database.

server id
Specifies the identifier of the remote database.

authentication *authtype*
Sets the authentication type for login, PPP, tunnel. *Authtype* is one of the following:

local Sets the authentication type to use a locally-maintained user database.

remote
Sets the authentication type to use a remote user database.

server id
Specifies the identifier of the remote database.

authorization *authtype*
Sets the authorization type for login, PPP, and tunnel. *Authtype* is one of the following:

local Sets the authorization type to use a locally-maintained user database.

remote
Sets the authorization type to use a remote user database.

server id
Specifies the identifier of the remote database.

Tunnel

Use the **tunnel** command to configure AAA for L2TP tunnel.

Table 30 lists the subcommands available with the **tunnel** command.

Table 30. Tunnel Subcommands

Command	Function
Disable	Disables accounting for L2TP tunnel.
List	Displays AAA configuration parameters for L2TP tunnel.
Set	Sets the AAA configuration parameters for L2TP tunnel.

Disable

Use the **tunnel disable** command to disable accounting for L2TP tunnel.

Syntax:

tunnel disable accounting

List

Use the **tunnel list** command to display the AAA for L2TP tunnel.

Syntax:

tunnel list all
accounting
authentication
authorization
config

Set

Use the **tunnel set** command to set the AAA configuration parameters for L2TP tunnel.

Syntax:

tunnel set aaa
accounting
authentication
authorization

aaa *authtype*
Sets the authentication, authorization, and accounting type. *Authtype* is one of the following:

local Sets the authentication, authorization, and accounting type to use a locally-maintained user database.

remote
Sets the authentication, authorization, and accounting type to use a remote user database.

server id
Specifies the identifier of the remote database.

accounting *authtype*
Sets the accounting type. *Authtype* is one of the following:

remote
Sets the authentication type to use a remote user database.

server id
Specifies the identifier of the remote database.

authentication *authtype*
Sets the authentication type. *Authtype* is one of the following:

local Sets the authentication type to use a locally-maintained user database.

remote
Sets the authentication type to use a remote user database.

server id
Specifies the identifier of the remote database.

authorization *authtype*
Sets the authorization type. *Authtype* is one of the following:

local Sets the authorization type to use a locally-maintained user database.

remote
Sets the authorization type to use a remote user database.

server id
Specifies the identifier of the remote database.

User-profiles

Use the **user-profiles** command to access the `User profile config>` command prompt. From this prompt, you can access the following commands.

Table 31. User-profile Configuration Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Add	Adds a PPP user profile.
Change	Changes a PPP user profile.
Delete	Deletes a PPP user profile.
Disable	Disables a PPP user profile.
Enable	Enables a PPP user profile.
List	Lists the PPP user profile information.
Report	Generates a PPP user profile report.
Reset-user	Resets a PPP user profile.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Add

Use the **user profiles add** command to add the user profile of a remote user to the local PPP user data base or to give a tunnel peer access through an IP network to the router.

Syntax:

add ppp-user
tunnel

ppp-user
Adds the user profile of a remote user to the local PPP user data base. You can add up to 500 users. You add a PPP user for each remote router or DIALS client that can connect to the device you are configuring.

See Add in the chapter “The CONFIG Process (CONFIG - Talk 6) and Commands” in *Software User’s Guide* for a description of the command syntax and options.

Example:

```
Config> add ppp-user
Enter name:  [ ]? pppusr01
Password:
Enter again to verify:
Allow inbound access for user? (Yes, No): [yes]
Will user be tunneled? (Yes, No): [No]
Number of days before account expiry[0-1000] [0]? 10
Number of grace logins allowed after an expiry[0-100] [0]? 5
IP address: [0.0.0.0]? 1.1.1.1
Set ECP encryption key for this user? (Yes, No): [No] no
Disable user ? (Yes, No): [No]

     PPP user name: pppusr01
   User IP address: 1.1.1.1
      Virtual Conn: disabled
        Encryption: disabled
            Status: enabled
    Login Attempts: 0
    Login Failures: 0
  Lockout Attempts: 0
   Account expires: Sun 17Feb2036 06:28:16
  Account duration: 10 days  00.00.00
   Password Expiry: <unlimited>

User 'pppusr01' has been added
```

Example:

```
Config> add ppp-user
Enter name:  [ ]? tunusr01
Password:
Enter again to verify:
Allow inbound access for user? (Yes, No): [yes]
Will user be tunneled? (Yes, No): [No] yes
Enter hostname to use when connection to this peer: []? host01
Tunnel-Server endpoint address: [0.0.0.0]? 1.1.1.1

--more--           PPP user name: tunusr01
--more--                Endpoint: 1.1.1.1
--more--                Hostname: host01

User 'tunusr01' has been added
```

tunnel Gives a tunnel peer access through an IP network to the router. The peer is then authorized to initiate tunneled PPP sessions into the router.

See Add in the chapter “Configuring the CONFIG Process” in *Software User’s Guide* for a description of the command syntax and options.

Example:

```
Config> add tunnel
Enter name:  []? tunnel02
Enter hostname to use when connecting to this peer: []? host02
Set shared secret? (Yes, No): [No]? yes
Shared secret for tunnel authentication:
Enter again to verify:
Tunnel-Server endpoint address: [0.0.0.0]? 2.2.2.22

       Tunnel name: tunnel02
          Endpoint: 2.2.2.22
```

Change

Use the **change** command to change a user-profile.

Syntax:

change	ppp-user
	tunnel

Delete

Use the **delete** command to delete a user-profile.

Syntax:

delete	ppp-user
	tunnel

Disable

Use the **disable** command to disable a user-profile.

Syntax:

disable	*name*

Enable

Use the **enable** command to enable a user-profile.

Syntax:

enable	*name*

List

Use the **list** command to list user-profile information.

Syntax:

list	ppp-user
	tunnel

```
User profile config> list ppp-user
List  (Name, Verb, User, Addr, Encr, zdump): [Verb]
     PPP user name: ppp01
            Expiry: <unlimited>
   User IP address: Interface Default
        Encryption: Not Enabled
            Status: Enabled
    Login Attempts: 0
    Login Failures: 0
  Lockout Attempts: 0
1 record displayed.
```

List Specifies how to access the list information.

Valid values: name, verb, user, addr, encr, zdump

Default value: verb

PPP user name
Lists the user name.

Expiry
List the expiration date.

User IP address
List the users IP address.

Encryption
Lists whether encryption is enabled or not enabled.

Status
Lists whether status is enabled or not enabled

Login attempts
Lists the number of times the user has attempted to login.

Login failures
Lists the number of failed attempts to login.

Lockout attempts
Lists the number of lockout attempts.

Report

Use the **report** command to generate a PPP user profile report.

Syntax:

report addresses
all
callback
dialout
dump
encrypt
name
password
time
user

```
User profile config> report addresses
PPP user name      User IP address
-----------------  ------------------
ppp01              Interface Default
1 record displayed.

User profile config> report all
     PPP user name: ppp01
            Expiry: <unlimited>
   User IP address: Interface Default
        Encryption: Not Enabled
            Status: Enabled
    Login Attempts: 0
    Login Failures: 0
  Lockout Attempts: 0
1 record displayed.
```

```
User profile config> report callback
PPP user name      Callback type        Phone Number
-----------------  -------------------  --------------------
ppp01
1 record displayed.

User profile config> report dialout
PPP user name      Dial-out
-----------------  ---------------
ppp01
1 record displayed.

User profile config> report dump
Enter user name:  []? user01

User profile config> report encrypt
PPP user name      Encryption
-----------------  ---------------
ppp01              Not Enabled
1 record displayed.

User profile config> report name
PPP user name
-----------------
ppp01
1 record displayed.

User profile config> report password
PPP user name      Expiry        Grace
-----------------  ------------  ------
ppp01              <unlimited>
1 record displayed.

User profile config> report time
PPP user name      Time alotted
-----------------  -------------------
ppp01
1 record displayed.

User profile config> report user
Enter user name:  []? login01
     PPP user name: login01
        Expiry: <unlimited>
   User IP address: Interface Default
        Encryption: Not Enabled
```

Reset-user

Use the **reset-user** command to reset a user-profile.

Syntax:

reset-user *name*

Chapter 14. Using and Configuring Encryption Protocols

Note: Encryption support is optional. If your software load does not include encryption, you will not see encryption-related parameters.

The use of multiple encryption (using encryption at both the IP Security Layer and at the Frame Relay or PPP data-Link Layer) within the router is restricted by U.S.A. Government export regulations. It is only supported in software loads that are under strict export control (software loads that support RC4 with 128 bit keys and Triple DES).

The objective of encryption is to transform data into an unreadable form to ensure privacy. The ***encrypted*** data needs to be decrypted to get the original data.

The 2210 supports:

- The RC4 encryption algorithm with 40 and 128 bit keys for Microsoft Point-to-Point Encryption (MPPE) on PPP interfaces.
- The Data Encryption Standard in Cipher Block Chaining Mode (DES-CBC) algorithm with 56-bit keys for PPP Encryption Control Protocol support as described in RCFs 1968 and 1969.
- The commercial Data Masking Facility (CDMF) which uses 40-bit keys for Frame Relay Encryption. This support is proprietary.
- Frame Relay also uses triple-DES and a 128-bit key.

PPP Encryption Using Encryption Control Protocol

The Encryption Control Protocol (ECP) is used in the router to negotiate the use of encryption on the point-to-point links communicating using PPP protocol. The Encryption Control Protocol provides a generalized mechanism to negotiate which encryption and decryption algorithms will be used over a PPP link. Different encryption algorithms can be negotiated in each direction of the PPP link.

A method of encryption and decryption is called an ***encryption algorithm***. Encryption algorithms use a key to control encryption and decryption. Unlike compression, the router encrypts in both directions of the link, because encrypting in only one direction is a security risk. The link will be terminated whenever ECP cannot negotiate encryption algorithms in both directions.

Configuring ECP Encryption for PPP

To configure the device to use encryption at the data link layer, you should:

1. Set the encryption keys for remote devices and local PPP interfaces.

   Set the encryption key for the remote device using the **add ppp-user** command at the `Config>` prompt. See the **Add** command in the chapter “Configuring the CONFIG Process” in *Software User’s Guide* for a description of the command syntax and options.

   Set the encryption key for the local PPP interface using the **enable ecp** command (see the talk 6 `PPP Config>` **enable** command in the *Software User’s Guide*).
2. Configure individual PPP links to use Encryption Control Protocol (ECP) by using the **enable ecp** command at the `PPP Config>` prompt.
3. Enable PAP, CHAP, or SPAP.

You can also disable encryption, change the encryption key for a user, list the status of encryption, or set the name that the device uses when requesting encryption. For information about:

- Disabling encryption, see the `PPP Config>` **disable ecp** command in the *Software User's Guide*.
- Changing the remote user's encryption key and password, see the `Config>` **change ppp-user** command in the *Software User's Guide*.
- Listing the encryption status, see the `PPP Config>` **list ecp** command in the *Software User's Guide*.
- Setting the device's name, see the `PPP Config>` **set name** command in *Software User's Guide*.

Monitoring ECP Encryption for PPP

You can monitor the various encryption settings on the interfaces by:

1. Accessing the monitoring prompt using the **talk 5** command.
2. Selecting the interface you want to monitor using the **network** command. This command puts you at the `PPP n>` prompt, in which *n* represents the network number. Refer to "Configuring and Monitoring Point-to-Point Protocol Interfaces" in the *Software User's Guide* for instructions about using the **network** command.

From this prompt, you can:

- List the current state of encryption, the most recent encryption negotiation, the elapsed time since an encryption state change, and the algorithms in use by the encrypters. (Refer to the **list control ecp** command in the *Software User's Guide*.)
- List the encryption control packets received and transmitted on the interface. (See the **list ecp** command in the *Software User's Guide*.)
- List the encrypted data packets transmitted or received on the interface. (see the **list edp** command in the *Software User's Guide*.)

Microsoft Point-to-Point Encryption (MPPE)

Microsoft Point-to-Point Encryption (MPPE) provides a way for remotely-attached Windows workstations known as Microsoft Dial-Up Networking (DUN) clients to encrypt data that is transmitted over a PPP link between themselves and the 2210. MPPE can also be used to encrypt data being transmitted over a PPP link from router to router. MPPE is always negotiated in both directions.

MPPE uses secret key algorithms to perform encryption. In secret key algorithms, the same key is used for encryption and decryption. This key is not configured by the user, but is generated in the process of the negotiation of MPPE between the sending and the receiving workstations. To use MPPE, you must configure the authentication protocol Microsoft Challenge/Handshake Authentication Protocol (MS-CHAP).

If the PPP interface is authenticated with MS-CHAP, the router goes into a "Microsoft mode", in which it will negotiate only MPPC if compresssion is enabled and negotiate only MPPE if encryption is enabled. In "Microsoft mode", the router ignores the priority list of compression algorithms and disables ECP negotiation.

Configuring MPPE

To configure MPPE, you should perform these steps for each interface:

1. Configure MS-CHAP. In the *Software User's Guide*, see "Microsoft PPP CHAP Authentication (MS-CHAP)" and "Configuring and Monitoring Point-to-Point Protocol Interfaces" for information about using and configuring MS-CHAP.
2. If you are configuring a router-to-router connection, set the name for the local PPP interface using the **set name** command (see the `PPP Config>` **set name** command in the *Software User's Guide*).
3. If you want data compression, enable MPPC using the talk 6 **enable ccp** command at the `PPP Config>` prompt. MPPE does not require data compression.
4. Enable MPPE. Use the **enable mppe** command at the `PPP Config>` prompt (see the `PPP Config>` **enable** command in the *Software User's Guide*).
5. Restart the router to activate the configuration.

You can also disable MPPE and list the MPPE options.

- Use the talk 6 **disable mppe** command at the `PPP Config>` prompt to disable MPPE.
- Use the talk 6 **list ccp** command at the `PPP Config>` prompt to list the MPPE options that have been configured.

Monitoring MPPE

Bring up the `PPP>` prompt as described in "Monitoring ECP Encryption for PPP" on page 180. Use the **list mppe** command to see the MPPE data statistics and the **list control ccp** command to see the MPPE status. Examples of the outputs of these commands are displayed in "Configuring and Monitoring Point-to-Point Protocol Interfaces" in the *Software User's Guide*.

Configuring Encryption on Frame Relay Interfaces

Note: Frame relay uses a proprietary encryption scheme.

Data encryption is supported on all interfaces on which you have enabled encryption. You can configure individual circuits on an encryption-enabled interface to perform or not perform encryption as desired.

To configure the device to use encryption on frame relay links:

1. Access the frame relay configuration prompt using the **talk 6** command.
2. Select the frame relay interface that you want to be encryption-capable using the **net** # command
3. Enable encryption on the frame relay interface using the **enable encryption** command. See the Frame Relay configuration commands in the *Software User's Guide*.
4. Add encryption—capable permanent virtual circuits and define the encryption key for each of the PVCs using the **add permanent-virtual-circuit** command. See the Frame Relay configuration commands in the *Software User's Guide*.
5. Repeat steps 1 through 4 for each encryption-capable interface you are configuring.

Note: If encryption is enabled for a FR permanent virtual circuit then data will not flow over the circuit unless encryption is successfully negotiated with the device at the other end of the virtual circuit. Encryption is not supported for orphan circuits since you must configure the PVC in order to enter the encryption key.

You can also disable encryption for an interface, change the encryption settings for a PVC or list the status of encryption. For information about

- Disabling encryption on an interface, see the Frame Relay Configuration **disable encryption** command in the *Software User's Guide*.
- Changing the encryption settings for a PVC, see the Frame Relay Configuration **change permanent-virtual-circuit** command in the *Software User's Guide*.
- Listing the encryption status, see the Frame Relay Configuration **list all,**, **list lmi,** and **list permanent-virtual-circuit** commands in the *Software User's Guide*.

Monitoring Encryption on Frame Relay Interfaces

You can monitor the various encryption settings on the interfaces by:

1. Accessing the monitoring prompt using the **talk 5** command.
2. Selecting the interface you want to monitor using the **network** # command. This command puts you at the FR *x*> prompt.

From this prompt, you can list the current encryption state for an interface, a PVC, or a circuit. See the Frame Relay Monitoring **list** command in the *Software User's Guide*.

Chapter 15. Configuring and Monitoring Quality of Service (QoS)

This chapter describes Quality of Service (QoS) configuration and operational commands for LAN and ELAN interfaces in the device. It contains the following sections:

Quality of Service Overview

The QoS feature leverages the benefits of ATM QoS capabilities for LAN Emulation Data Direct VCCs. This support is referred to as "Configurable QoS for LAN Emulation". The key attributes and the benefits of this feature are as follows:

- An LE Client makes use of configured QoS parameters for its Data Direct VCCs.
- QoS parameters can be configured for:
  - LE Client
  - ATM Interface
- The set of QoS parameters configured are for use with ATM Forum UNI 3.0/3.1 signaling. The parameters include the desired Peak Cell Rate, Sustained Cell Rate, QoS Class and Maximum Burst Size.
- Maximum Reserved Bandwidth per VCC can be configured to protect an LE Client from accepting/establishing VCCs whose traffic parameters it cannot support.
- The QoS Negotiation mechanism enables the participating LE Clients to be aware of each other's QoS parameters. A data-direct VCC is set up using the negotiated parameters.

Benefits of QoS

- Using QoS for the LE Client, ATM Interface, or Emulated LAN provides the following benefits for LANE Data Direct VCCs.
  - An LE Client can be configured with QoS if the QoS required by the client is different from the QoS required by other clients on the ELAN. For example, if an LE Client serves a file server, then the user may want to configure appropriate QoS parameters for all traffic to and from the file server.
  - An ATM Interface can be configured with QoS if a user wants all LE Clients on that ATM interface to use the same set of parameters. For example, if an ATM Interface is connected at 25 Mbps, the user can configure appropriate parameters that are different from those at a 155-Mbps interface.

QoS Configuration Parameters

This section describes nine parameters that are used for QoS configuration. The following six parameters can be configured for an LE Client, ATM Interface, and an Emulated LAN:

1. max-reserved-bandwidth
2. traffic-type
3. peak-cell-rate
4. sustained-cell-rate
5. max-burst-size
6. qos-class

The following two parameters can be configured for an Emulated LAN and an LE Client:

1. *validate-pcr-of-best-effort-vccs*
2. *negotiate-qos*

The *accept-qos-parms-from-lecs* parameter can be configured only for an LE Client.

The first six parameters control the traffic characteristics of Data Direct VCCs established by the LE Client while the first parameter also applies to the calls received by the LE Client. The following characteristics are associated with all the Data Direct VCCs established by the LE Client:

- Bandwidth is not reserved for best-effort traffic.
- Traffic parameters apply to both forward and backward directions.
- When a reserved bandwidth connection is rejected due to the traffic parameters or QoS Class, the call is retried as a best-effort connection with the configured peak cell rate (cause codes on release or release-complete messages are used to determine why a VCC was released).
- When a best-effort connection is rejected due to the Peak Cell Rate (PCR), the call may be automatically retried with a lower PCR. Retries are performed under the following conditions:
  1. If the rejected PCR is greater than 100 Mbps, the call is retried with a PCR of 100 Mbps.
  2. Otherwise, if the rejected PCR is greater than 25 Mbps, the call is retried with a PCR of 25 Mbps.

Maximum Reserved Bandwidth (max-reserved-bandwidth)

The maximum reserved bandwidth acceptable for a Data Direct VCC. This parameter applies to both Data Direct VCC calls received by the LE Client and Data Direct VCC calls placed by the LE Client. For incoming calls, this parameter defines the maximum acceptable SCR for a Data Direct VCC. If SCR is not specified on the incoming call, then this parameter defines the maximum acceptable PCR for a Data Direct VCC with reserved bandwidth.

Calls received with traffic parameters specifying higher rates will be released. If SCR is specified on the incoming call, the call will not be rejected due to the PCR or Maximum Burst Size. The constraint imposed by this parameter is not applicable to best_effort connections. For outgoing calls, this parameter sets an upper bound

on the amount of reserved bandwidth that can be requested for a Data Direct VCC. Therefore the traffic-type and sustained-cell-rate parameters are dependent upon this parameter.

Valid Values:
Integer in the range 0 to the line speed of ATM device in Kbps

Default Value:
0

Traffic Type (traffic-type)

The desired traffic type for Data Direct VCCs. If QoS parameters are not negotiated, then this parameter specifies the type of calls placed by the LE Client. Otherwise, if QoS parameters are negotiated, this parameter specifies the desired type of traffic characteristics for Data Direct VCCs. When QoS parameters are negotiated, if either the source or target LEC desires a reserved bandwidth connection and both LECs support reserved bandwidth connections (that is, max-reserved-bandwidth > 0), then an attempt will be made to establish a reserved bandwidth Data Direct VCC between the two LECs. Otherwise, the Data Direct VCC will be a best-effort connection. Dependencies: max-reserved-bandwidth

Valid Values:
best_effort or reserved_bandwidth

Default:
best_effort

Peak Cell Rate (peak-cell-rate)

The desired peak cell rate for Data Direct VCCs. If QoS parameters are not negotiated, then this parameter specifies the PCR traffic parameter for Data Direct VCC calls placed by the LE Client. Otherwise, if QoS parameters are negotiated, this parameter specifies the desired PCR traffic parameter for Data Direct VCCs. The minimum of the desired PCRs of the two LECs is used for negotiated best-effort VCCs.

When a reserved bandwidth VCC is negotiated and only one of the LE Clients requests a reserved bandwidth connection, then the desired PCR of that LEC is used for the Data Direct VCC subject to the upper bound imposed by the line rate of the local ATM device. If both LECs request a reserved bandwidth connection, then the maximum of the desired PCRs of the LE Clients is used for the Data Direct VCC subject to the upper bound imposed the line rate of the local ATM device.

Valid Values:
An integer value in the range 0 to the line speed of ATM device in Kbps

Default Value:
Line speed of LEC ATM Device in Kbps.

Sustained Cell Rate (sustained-cell-rate)

The desired sustained cell rate for Data Direct VCCs. If QoS parameters are not negotiated, then this parameter specifies the SCR traffic parameter for Data Direct VCC calls placed by the LE Client. Otherwise, if QoS parameters are negotiated, this parameter specifies the desired SCR traffic parameter for Data Direct VCCs.

When a reserved bandwidth VCC is negotiated and only one of the LE Clients requests a reserved bandwidth connection, then the desired SCR of that LEC is used for the Data Direct VCC (subject to the upper bound imposed by the max-reserved-bandwidth parameter of the other LEC). If both LECs request a reserved bandwidth connection, then the maximum of the desired SCRs of the LE Clients is used for the Data Direct VCC (subject to the upper bound imposed by the max-reserved-bandwidth parameters of both LECs). In any case (negotiation or not), if the SCR that is to be signaled equals the PCR that is to be signaled, then the call is signaled with PCR only.

Dependencies: max-reserved-bandwidth, traffic-type and peak-cell-rate. This parameter is applicable only when traffic-type is reserved_bandwidth.

Valid Values:

An integer value in the range 0 to the minimum of max-reserved-bandwidth and peak-cell-rate, specified in Kbps

Default Value

None

Maximum Burst Size (max-burst-size)

The desired maximum burst size for Data Direct VCCs. If QoS parameters are not negotiated, then this parameter specifies the Maximum Burst Size traffic parameter for Data Direct VCC calls placed by the LE Client. Otherwise, if QoS parameters are negotiated, this parameter specifies the desired Maximum Burst Size traffic parameter for Data Direct VCCs.

When a reserved bandwidth VCC is negotiated and only one of the LE Clients requests a reserved bandwidth connection, then the desired Maximum Burst Size of that LEC is used for the Data Direct VCC. If both LECs request a reserved bandwidth connection, then the maximum of the desired Maximum Burst Sizes of the LE Clients is used for the Data Direct VCC.

In any case (negotiation or not), the Maximum Burst Size is signaled only when SCR is signaled. Although this parameter is expressed in units of cells, it is configured as an integer multiple of the Maximum Data Frame Size (specified in LEC's C3 parameter) with a lower bound of 1.

Dependencies: This parameter is applicable only when traffic-type is reserved_bandwidth.

Valid Values:

An integer number of frames; must be greater than 0

Default:

1 frame

QoS Class (qos-class)

The desired QoS class for reserved bandwidth calls. If QoS parameters are not negotiated, then this parameter specifies the QoS Class to be used for reserved bandwidth Data Direct VCC calls placed by the LE Client. Otherwise, if QoS parameters are negotiated, this parameter specifies the QoS Class that is desired for Data Direct VCCs. Unspecified QoS Class is always used on best-effort calls.

Specified QoS Classes define objective values for ATM performance. Specified QoS Classes define objective values for ATM performance parameters such as cell loss ratio and cell transfer delay.

The UNI Specification states that:

Specified QoS Class 1
should yield performance comparable to current digital private line performance.

Specified QoS Class 2
is intended for packetized video and audio in teleconferencing and multimedia applications.

Specified QoS Class 3
is intended for interoperation of connection oriented protocols, such as Frame Relay.

Specified QoS Class 4
is intended for interoperation of connectionless protocols, such as IP or SMDS.

LECs must be able to accept calls with any of the above QoS Classes. When QoS parameters are negotiated, the configured QoS Classes of the two LECs are compared, and the QoS Class with the more stringent requirements is used.

Valid Values:
0: for Unspecified QoS Class

1: for Specified QoS Class 1

2: for Specified QoS Class 2

3: for Specified QoS Class 3

4: for Specified QoS Class 4

Default Value:
0 (Unspecified QoS Class)

Validate PCR of Best-Effort VCCs (validate-pcr-of-best-effort-vccs)

To validate Peak Cell Rate of Best-Effort VCCs. When FALSE, best-effort VCCs will be accepted without regard to the signaled forward PCR. When TRUE, best- effort VCCs will be rejected if the signaled forward PCR exceeds the line rate of the LE Client ATM device. Calls will not be rejected due to the backward PCR. The signaled backward PCR will be honored if it does not exceed the line rate; otherwise, transmissions to the caller will be at line rate.

Notes:

1. Accepting best-effort VCCs with forward PCRs that exceed the line rate can result in poor performance due to excessive retransmissions; however, rejecting these VCCs can result in interoperability problems.
2. The yes setting is useful when callers will retry with a lower PCR following call rejection due to unavailable cell rate.

Valid Values:
yes, no

Default Value:
no

Negotiate QoS (negotiate-qos)

Enable QoS parameter negotiation for Data Direct VCCs. This parameter should be enabled only when connecting to an IBM MSS LES. When this parameter is yes, the LE Client will include an IBM Traffic Parameter TLV in LE_JOIN_REQUEST and LE_ARP_RESPONSE frames sent to the LES. This TLV will include the values of max-reserved-bandwidth, traffic-type, peak-cell-rate, sustained-cell-rate, max-burst-size and qos-class. An IBM Traffic Parameter TLV may also be included in a LE_ARP_RESPONSE returned to the LE Client by the LES.

If there is no TLV in a LE_ARP_RESPONSE received by the LE Client, then the local configuration parameters must be used to setup the Data Direct VCC. If a TLV is included in a LE_ARP_RESPONSE, the LE Client must compare the contents of the TLV with the corresponding local values to determine the “negotiated” or “best” set of parameters acceptable to both parties before signalling for the Data Direct VCC.

Valid Values:
yes, no

Default Value:
no

Accept QoS Parms from LECS (accept-qos-parms-from-lecs)

This parameter gives the ability to configure an LE Client to accept/reject QoS parameters from a LECS. When this parameter is yes, the LE Client should use the QoS parameters obtained from the LE Clients in the LE_CONFIGURE_RESPONSE frames, that is, the QoS parameters from the LE Clients override the locally configured QoS parameters. If this parameter is no then the LE Client will ignore any QoS parameters received in an LE_CONFIGURE_RESPONSE frame from the LE Clients.

Valid Values:
yes, no

Default Value:
no

Accessing the QoS Configuration Prompt

Use the **feature** command from the CONFIG process to access the Quality of Service configuration commands. Enter **feature** followed by the feature number (6) or short name (QoS). For example:

```
Config> feature qos
Quality of Service - Configuration
QoS Config>
```

Once you access the `QoS Config>` prompt, you can configure the Quality of Service (QoS) of an LE Client, or an ATM Interface. To return to the `Config>` prompt at any time, enter the **exit** command at the `QoS Config>` prompt.

Alternatively, you can configure QoS parameters for an LE Client or an ATM Interface by accessing the entities as follows:

- LE Client
  1. At the `Config>` prompt, enter the **network** command and the LE Client interface number.

2. At the LE Client configuration> prompt enter **qos-configuration**.

   Example:

```
config> network 3
Token Ring Forum Compliant LEC Config> qos-configuration
LEC QoS Config>
```

- ATM Interface
  1. at the Config> prompt, enter the **network** command and the ATM interface number to get you to the ATM Config> prompt.
  2. Enter the **interface** parameter to get to the ATM Interface Config> prompt.
  3. At the ATM InterfaceConfig> prompt enter **qos-configuration**.

     Example:

```
config> network 0
ATM Config> interface
ATM Interface Config> qos-configuration
ATM-I/F 0 QoS>
```

Quality of Service Commands

This section summarizes the QoS configuration commands. Use the following commands to configure Quality of Service. Enter the commands from the QoS Config> prompt.

Table 32. Quality of Service (QoS) Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
le-client	Gets you to the LE Client QoS configuration > prompt for the selected LE client.
atm-interface	Gets you to the ATM Interface QoS configuration> prompt for the selected ATM interface.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

LE Client QoS Configuration Commands

This section summarizes and explains the commands for configuring QoS for a specific LE Client.

Use the following commands at the LEC QoS config> prompt.

Table 33. LE Client Quality of Service (QoS) Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
List	Lists the current QoS configuration of the LE Client.
Set	Sets the QoS parameters of the LE Client.
Remove	Removes the QoS configuration of the LE Client.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

List

Use the **list** command to list the QoS configuration of this LE Client. QoS parameters are listed only if at least one has been specifically configured (see Example 1). Otherwise, no parameters are listed (see Example 2).

Syntax:

list

Example 1:

```
LEC QoS Config> list

           LE Client QoS Configuration for Data Direct VCCs
           ========================================
     (ATM interface number = 0,  LEC interface number = 3)

  Maximum Reserved Bandwidth for a Data-Direct VCC = 10000 Kbps
  Data-Direct VCC Type ........................... = Best-Effort
  Data-Direct VCC Peak Cell Rate ................. = 155000 Kbps
  Data-Direct VCC Sustained Cell Rate ............ = 155000 Kbps
  Desired QoS Class of Reserved Connections ...... = 0
  Max Burst Size of Reserved Connections    ...... = 0 frames

  Validate Peak Rate of Best-Effort connections .. = No
  Enable QoS Parameter Negotiation ............... = Yes
  Accept QoS Parameters from LECS ................ = Yes

LEC QoS Config>
```

Example 2:

```
LEC QoS Config> list

   QoS has not been configured for this LEC.
   Please use the SET option to configure QoS.

LEC QoS Config>
```

Set

Use the **set** command to specify LE Client QoS parameters.

Syntax:

set accept-qos-parms-from-lecs
all-default-values
max-burst-size
max-reserved-bandwidth
negotiate-qos
peak-cell-rate
qos-class
sustained-cell-rate
traffic-type
validate-pcr-of-best-effort-vccs

accept-qos-parms-from-lecs
Use this option to enable/disable the LE Client to accept/reject the QoS

parameters received from an LECS as TLVs. See “Accept QoS Parms from LECS (accept-qos-parms-from-lecs)” on page 188 for a more detailed description of this parameter.

Valid Values:

yes, no

Default Value:

yes

Example:

```
LEC QoS Config> se acc y
LEC QoS Config>
```

all-default-values

Use this option to set the QoS parameters to default values. In the following example the default values are also listed.

Example:

```
LEC QoS Config> set all-default-values
   Failed to locate existing QoS configuration record!
   Using a new set of default values ...
   Initializing all parameters to default values
   LEC QoS Config>  list

              LE Client QoS Configuration for Data Direct VCCs
              ========================================
         (ATM interface number = 0,  LEC interface number = 3)

     Maximum Reserved Bandwidth for a Data-Direct VCC = 0 Kbps
     Data-Direct VCC Type ........................... = Best-Effort
     Data-Direct VCC Peak Cell Rate ................. = 155000 Kbps
     Data-Direct VCC Sustained Cell Rate ............ = 155000 Kbps
     Desired QoS Class of Reserved Connections ...... = 0
     Max Burst Size of Reserved Connections    ...... = 0 frames

     Validate Peak Rate of Best-Effort connections .. = No
     Enable QoS Parameter Negotiation ............... = No
     Accept QoS Parameters from LECS ................ = Yes

  LEC QoS Config>
```

max-burst-size

Sets the desired maximum burst size in frames. See “Maximum Burst Size (max-burst-size)” on page 186 for a more detailed description of this parameter.

Valid Values:

An integer number of frames; must be greater than 0

Default:

1 frame

Example:

```
LEC QoS Config> se ma
   Maximum Burst Size in Kbps [1]? 10000
   LEC QoS Config>
```

max-reserved-bandwidth

Use this option to set the maximum reserved bandwidth allowable per Data Direct VCC. See “Maximum Reserved Bandwidth (max-reserved-bandwidth)” on page 184 for a more detailed description of this parameter.

Valid Values:

Integer in the range 0 to the line speed of ATM device in Kbps

Default Value:

0

Example:

```
LEC QoS Config> set max-reserved-bandwidth
Maximum reserved bandwidth acceptable for a data-direct VCC (in Kbps) [0]? 20000
LEC QoS Config>
```

negotiate-qos

Use this option to enable/disable the LE Client's participation in QoS negotiation. See "Negotiate QoS (negotiate-qos)" on page 188 for a more detailed description of this parameter.

Valid Values:

yes, no

Default Value:

no

Example:

```
LEC QoS Config> se neg y
LEC QoS Config>
```

peak-cell-rate

Sets the desired peak cell rate for Data Direct. See "Peak Cell Rate (peak-cell-rate)" on page 185 for a more detailed description of this parameter.

Valid Values:

An integer value in the range 0 to the line speed of ATM device in Kbps

Default Value:

Line speed of LEC ATM Device in Kbps.

Example:

```
LEC QoS Config> set peak-cell-rate
Data-Direct VCC Peak Cell Rate in Kbps [1]? 25000
LEC QoS Config>
```

qos-class

Sets the desired QoS Class for Data Direct VCCs. See "QoS Class (qos-class)" on page 186 for a more detailed description of this parameter.

Valid Values:

0: for Unspecified QoS Class

1: for Specified QoS Class 1

2: for Specified QoS Class 2

3: for Specified QoS Class 3

4: for Specified QoS Class 4

Default Value:

0 (Unspecified QoS Class)

Example:

```
LEC QoS Config> se qos
Desired QoS Class for Data Direct VCCs [0]? 1
LEC QoS Config>
```

sustained-cell-rate

Sets the desired sustained cell rate for Data Direct VCCs. See "Sustained Cell Rate (sustained-cell-rate)" on page 185 for a more detailed description of this parameter.

Valid Values:

An integer value in the range 0 to the minimum of max-reserved-bandwidth and peak-cell-rate, specified in Kbps

Default Value

None

Example:

```
LEC QoS Config> se sus
Data-Direct VCC Sustained Cell Rate in Kbps [1]? 10000
LEC QoS Config>
```

traffic-type

Sets the desired traffic for Data Direct VCCs. See “Traffic Type (traffic-type)” on page 185 for a more detailed description of this parameter.

Valid Values:

best effort or reserved bandwidth

Default:

best effort

Example:

```
LEC QoS Config>set traffic-type
  Choose from:
     (0): Best-Effort
     (1): Reserved-Bandwidth
Data Direct VCC Type [0]? 1
Note: Peak Cell Rate has been reset to 1
      Sustained Cell Rate has been reset to 1
      Max Reserved Bandwidth has been reset to 1
      Please configure appropriate values.
LEC QoS Config>
```

validate-pcr-of-best-effort-vccs

Use this option to enable/disable validation of the Peak Cell Rate traffic parameter of the Data Direct VCC calls received by this LE Client. See “Validate PCR of Best-Effort VCCs (validate-pcr-of-best-effort-vccs)” on page 187 for a more detailed description of this parameter.

Valid Values:

yes, no

Default Value:

no

Example:

```
LEC QoS Config> se val y
LEC QoS Config>
```

Remove

Use the **remove** command to remove the QoS configuration of this LE Client.

Syntax:

remove

Example:

```
LEC QoS Config> remove
WARNING: This option deletes the QoS configuration.
         To re-configure use any of the SET options.
Should the LEC QoS configuration be deleted? [No]: yes
Deleted QoS configuration successfully
LEC QoS Config>
```

ATM Interface QoS Configuration Commands

Table 34. LE Client Quality of Service (QoS) Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
List	Lists the current ATM Interface QoS configuration.
Set	Sets the ATM Interface QoS parameters.
Remove	Removes the QoS configuration of the ATM Interface.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

List

Use the **list** command to list the QoS configuration of this ATM Interface. QoS parameters are listed only if at least one parameter has been configured (see following example). Otherwise, no parameters are listed.

Syntax:

list

Example:

```
ATM-I/F 0 QoS> list

          ATM Interface 'Quality of Service' Configuration
          ================================================
                    (ATM interface number = 0  )

     Maximum Reserved Bandwidth for a VCC = 15000 Kbps
     VCC Type ........................... = RESERVED-BANDWIDTH
     Peak Cell Rate ..................... = 20000 Kbps
     Sustained Cell Rate ................ = 5000 Kbps
     QoS Class .......................... = 4
     Maximum Burst Size ................. = 5 frames
ATM-I/F 0 QoS>
```

Set

Use the **set** command to specify ATM Interface QoS parameters.

Syntax:

set max-burst-size
max-reserved-bandwidth
peak-cell-rate
qos-class
sustained-cell-rate
traffic-type

max-burst-size

Sets the desired maximum burst size in frames. See “Maximum Burst Size (max-burst-size)” on page 186 for a more detailed description of this parameter.

Valid Values:

An integer number of frames; must be greater than 0

Default:

1 frame

Example:

```
ATM-I/F 0 QoS Config> se ma
Maximum Burst Size in Kbps [1]? 10000
ATM-I/F 0 QoS Config>
```

max-reserved-bandwidth

Use this option to set the maximum reserved bandwidth allowable for each Data Direct VCC. See “Maximum Reserved Bandwidth (max-reserved-bandwidth)” on page 184 for a more detailed description of this parameter.

Valid Values:

Integer in the range 0 to the line speed of ATM device in Kbps

Default Value:

0

Example:

```
ATM-I/F 0 QoS> se max-reserved-bandwidth
Maximum reserved bandwidth acceptable for a data-direct VCC (in Kbps) [0]?
15000
ATM-I/F 0 QoS>
```

peak-cell-rate

Sets the desired peak cell rate for Data Direct VCCs. See “Peak Cell Rate (peak-cell-rate)” on page 185 for a more detailed description of this parameter.

Valid Values:

An integer value in the range 0 to the line speed of ATM device in Kbps

Default Value:

Line speed of LEC ATM Device in Kbps.

Example:

```
ATM-I/F 0 QoS Config> set peak-cell-rate
Data-Direct VCC Peak Cell Rate in Kbps [1]? 25000
ATM-I/F 0 QoS Config>
```

qos-class

Sets the desired QoS Class for Data Direct VCCs. See “QoS Class (qos-class)” on page 186 for a more detailed description of this parameter.

Valid Values:

0: for Unspecified QoS Class

1: for Specified QoS Class 1

2: for Specified QoS Class 2

3: for Specified QoS Class 3

4: for Specified QoS Class 4

Default Value:

0 (Unspecified QoS Class)

Example:

```
ATM-I/F 0 QoS Config> se qos
Desired QoS Class for Data Direct VCCs [0]? 1
ATM-I/F 0 QoS Config>
```

sustained-cell-rate

Sets the desired sustained cell rate for Data Direct VCCs. See “Sustained Cell Rate (sustained-cell-rate)” on page 185 for a more detailed description of this parameter.

Valid Values:

An integer value in the range 0 to the minimum of max-reserved-bandwidth and peak-cell-rate; specified in Kbps

Default Value

None

Example:

```
ATM-I/F 0 QoS Config> se sus
Data-Direct VCC Sustained Cell Rate in Kbps [1]? 10000
ATM-I/F 0 QoS Config>
```

traffic-type

Sets the desired traffic for Data Direct VCCs. See “Traffic Type (traffic-type)” on page 185 for a more detailed description of this parameter.

Valid Values:

best_effort or reserved_bandwidth

Default:

best_effort.

Example:

```
ATM-I/F 0 QoS> set traffic-type
  Choose from:
     (0): Best-Effort
     (1): Reserved Bandwidth
Traffic Type of VCCs [1]? 0
ATM-I/F 0 QoS>
```

Remove

Use the **remove** command to remove the QoS configuration of this ATM Interface.

Syntax:

remove

Example:

```
ATM-I/F 0 QoS> remove
WARNING: This option deletes the QoS configuration.
         To re-configure use any of the SET options.
Should the ATM Interface QoS configuration be deleted? [No]: yes
Deleted QoS SRAM record successfully
ATM-I/F 0 QoS>
```

Accessing the QoS Monitoring Commands

Use the **feature** command from the GWCON process to access the Quality of Service monitoring commands. Enter the **feature** followed by the feature number (6) or short name (QoS). For example:

```
+feature qos
Quality of Service (QoS) - User Monitoring
QoS+
```

Once you access the QoS monitoring prompt, you can select the monitoring of a particular LE Client. To return to the GWCON prompt at any time, enter the exit command at the QoS monitoring prompt.

Alternatively, you can access the QoS Monitoring of an LE Client as follows:

1. At the GWCON prompt (+), enter the network command and the LE Client interface number.
2. At the LE Client monitoring prompt enter **qos-information**.

   Example:

```
+network 3
ATM Emulated LAN Monitoring
LEC+qos information
LE Client QoS Monitoring
LEC 3 QoS+
```

Quality of Service Monitoring Commands

This section summarizes the QoS monitoring commands. Enter these commands at the QoS+ prompt.

Table 35. Quality of Service (QoS) Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
le-client	Gets you to the `LE Client QoS console +` prompt for the selected LE client.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

LE Client QoS Monitoring Commands

This section summarizes the LE Client QoS monitoring commands. Enter the commands from the `LEC num QoS+` prompt.

Table 36. LE Client QoS Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
List	Lists the current LE Client QoS information. Options include: configuration parameters, TLVs, VCCs, and statistics.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

List

Use the **list** command to list the QoS related information of this LE Client.

Syntax:

list configuration-parameters
data-direct-VCCs (Detailed Information)
statistics
tlv-information
vcc-information

configuration-parameters

Lists the QoS configuration parameters. Because parameters can be configured for an LE Client, ATM Interface or the ELAN, these parameters are displayed along with a resolved set of parameters that are used by the LE Client.

le-client

The parameters configured for this LE Client which are obtained from the SRAM records. If the SRAM records contain an invalid set of parameters then this column will not display any parameters values.

ATM Interface

The parameters configured for the ATM Interface used by this LE Client. These parameters are obtained from the local SRAM records. If the SRAM records contain an invalid set of parameters then this column will not display any parameter values.

From LECS

The parameters received by this LE Client from the LE Configuration Server. The parameters are received as individual TLVs in the LE_CONFIGURE_RESPONSE control message.

used The resolved set of traffic parameters which are used by for its Data Direct VCCs. If none of the entities is configured with QoS parameters, then the USED parameters represent the default parameters. If parameters are configured for at least one entity, then they are resolved as follows:

- If only the LE Client or the ATM Interface is configured with parameters and either the accept-parms-from-lecs is FALSE or no parameters were received from the LECS, then the configured LE Client or the ATM Interface parameters are used.
- If both the LE Client and the ATM Interface have configured parameters, then the LE Client parameters are used.
- If the accept-parms-from-lecs is TRUE and parameters were received from the LECS, then the LE Client parameters (or the default if the LE Client is not configured) are combined with those received from the LECS to form a complete set of the first six QoS parameters described in “QoS Configuration Parameters” on page 184.
- If the set of the first six QoS parameters described in “QoS Configuration Parameters” on page 184 contains an invalid combination then the parameters from the LECS are rejected. Note that the two flags negotiate-qos and validate-pcr-of-best-effort-vccs are validated independently.

Example:

```
LEC 1 QoS+ list configuration parameters

                ATM LEC Configured QoS  Parameters
                ==================================

     QoS                                        |    LEC      ATM-IF    FROM

   PARAMETER                            USED    |    SRAM     SRAM      LECS

 ------------------------------------------------------------------------------
 Max Reserved Bandwidth (cells/sec) :    23584   |    23584        0     none
                       (Kbits/sec) :    10000   |    10000        0     none
 VCC Type ........................ :   ResvBW   |   ResvBW   BstEft        0
 Peak Cell Rate ........(cells/sec) :    18867   |    18867   365566   365566
```

```
                          (Kbits/sec) :     8000  |     8000   155000   155000
 Sustained Cell Rate ...(cells/sec) :    18867  |    18867   365566     none
                          (Kbits/sec) :     8000  |     8000   155000     none
 QoS Class ....................... :        4  |        4        0     none
 Max Burst Size .....(cells) :       95  |       95        0     none
                            (frames) :        1  |        1        0     none
 Validate PCR of Best-Effort VCCs . :       no  |       no      n/a     none
 Enable QoS Negotiation ........... :      yes  |      yes      n/a     none
 Accept QoS Parameters from LECS .. :      yes  |      yes      n/a      n/a
 ------------------------------------------------------------------------------
(BstEft = Best Effort, ResvBW = Reserved Bandwidth)
(n/a = not applicable, none = no value is specified)

LEC 1 QoS+
```

data-direct-vccs (Detailed Information)

This option lists the Data Direct VCC information of this LE Client. Similar information is also listed using **list vcc-information**.

Example:

```
LEC 1 QoS+ list data direct vccs

          LEC Data Direct VCCs - QoS Information
          ======================================

Conn Handle = 80, VPI = 0, VCI = 546
        Connection Type = RETRIED CONNECTION PARAMETERS
          TrafficType    = BEST EFFORT VCC
          PCR            = 58962 (25 Mbps)
          SCR            = 58962 (25 Mbps)
          QoS Class      = 0
          Max Burst Size = 0

Conn Handle = 78, VPI = 0, VCI = 544
        Connection Type = PARAMETERS SET BY DESTINATION
          TrafficType    = RESERVED BANDWIDTH VCC
          PCR            = 58962 (25 Mbps)
          SCR            = 16509 (7 Mbps)
          QoS Class      = 1
          Max Burst Size = 95

LEC 1 QoS+
```

statistics

Counters are maintained for the following statistics:

Successful QoS Connections
Number of RESERVED-BANDWIDTH connections established by the LE Client.

Successful Best-Effort Connections
Number of BEST-EFFORT connections established by the LE Client.

Failed QoS Connections
Number of RESERVED-BANDWIDTH connection requests made by the LE Client that failed.

Failed Best-Effort Connections
Number of BEST-EFFORT connection requests made by the LE Client that failed.

QoS Negotiation Applied
Number of times the QoS negotiation extension was applied. Parameters are negotiated if the LE Client receives the destination LE Client's parameters in an LE_ARP_RESPONSE control message.

PCR Proposal (IBM) Applied
Number of times the IBM Peak Cell Rate Proposal was applied. This proposal recommends using specific rate parameters if signaling at 100 Mbps or 155 Mbps for BEST-EFFORT connections.

This allows other participating IBM products (for example, 25-Mbps ATM adapters) to reject a connection based on the signaled peak cell rates.

QoS Connections Accepted
Number of RESERVED-BANDWIDTH connections accepted by this LE Client.

Best-Effort Connections Accepted
Number of BEST-EFFORT connections accepted by this LE Client.

QoS Connections Rejected
Number of RESERVED-BANDWIDTH connection requests received by this LE Client that were rejected.

Best-Effort Connections Rejected
Number of BEST-EFFORT connection requests received by this LE Client that were rejected.

Rejected due to PCR Validation
Number of BEST-EFFORT connections rejected by the LE Client due to validation of Peak Cell Rate when the validate-pcr-of-best-effort-vccs parameter is TRUE.

Example:

```
LEC 1 QoS+ li stat

QoS Statistics: of Data Direct Calls Placed by the LEC
------------------------------------------------------
    Successful QoS Connections          = 0
    Successful Best-Effort Connections  = 1
    Failed QoS Connections              = 1
    Failed Best-Effort Connections      = 1
    Qos Negotiation Applied             = 0
    PCR Proposal (IBM) Applied          = 0

QoS Statistics: of Data Direct Calls Received by the LEC
--------------------------------------------------------
    QoS Connections Accepted            = 1
    Best-Effort Connections Accepted    = 0
    QoS Connections Rejected            = 0
    Best-Effort Connections Rejected    = 0
    Rejected due to PCR Validation      = 0

LEC 1 QoS+
```

tlv-information
Lists the IBM Traffic Information TLV that this LE Client registered with the LE Server. The TLV is registered only if the LE Client is participating in QoS Negotiation.

Example:

```
LEC 1 QoS+ list tlv

  Traffic Info TLV of the LEC (registered with the LES)
  =====================================================
    TLV Type .........................= 268458498
    TLV Length .......................= 24
    TLV Value:
           Maximum Reserved Bandwidth = 23584 cells/sec (10 Mbps)
           Data Direct VCC Type...... = RESERVED BANDWIDTH VCC
           Data Direct VCC PCR....... = 18867 cells/sec (8 Mbps)
           Data Direct VCC SCR....... = 18867 cells/sec (8 Mbps)
           Data Direct VCC QoS Class  = 4
           Maximum Burst Size         = 95 cells (1 frames)

LEC 1 QoS+
```

vcc-information
Lists all active VCCs of the LE Client. The information includes the traffic parameters of the connections. For BEST-EFFORT connections, the

Sustained Cell Rate is displayed to be the same as the Peak Cell Rate, QoS Class and the Maximum Burst Size are displayed as 0.

The Parameter Descriptor entries are:

SrcParms
Parameters of a connection established by this LE Client.

DestParms
Parameters of a connection received by this LE Client.

NegoParms
Parameters of a connection established by the LE Client for which the QoS Negotiation was used.

RetryParms
Parameters of a connection established by this LE Client after failing at least once.

Example:

```
LEC 1 QoS+ li vcc

                          LEC VCC Table
                          =============

                                                                          Burst
Conn   Conn              Conn          VCC    PCR    SCR   QoS   Size    Parameters
Index  Handle VPI   VCI   Type  Status  Type  (kbps) (kbps) Class (cells) Descriptor
-------------------------------------------------------------------------------------
   2)     69    0   535  Cntrl  Ready  BstEft 155000 155000    0      0   SrcParms
   3)     71    0   537  Cntrl  Ready  BstEft      0      0    0      0   DestParms
   4)     72    0   538  Mcast  Ready  BstEft 155000 155000    0      0   SrcParms
   5)     74    0   540  Mcast  Ready  BstEft      0      0    0      0   DestParms
   6)     78    0   544  Data   Ready  ResvBW  25000   7000    1     95   DestParms

LEC 1 QoS+
```

Chapter 16. Using the Policy Feature

This chapter describes how the policy feature interacts with other router software components to make decisions about QOS, security, or both. It also describes the concepts and specific configuration commands related to the policy feature. The policy feature also allows an LDAP directory server to be used as a central repository for policy information. The concepts and configuration steps needed to enable the LDAP functions are also described in this chapter. The following topics discuss these concepts, the way in which routers enforce policies, and also provide examples.

Overview of Policy

The policy feature facilitates the management of IPv4 traffic in a network. You may configure policies for very simple filter rules (drop or pass) or for complex security and QOS scenarios. The combination of policies determines how routers handle IPv4 traffic in a network.

Policy Decision and Enforcement

The policy implementation in this family of routers constitutes the basis for policy decisions and the means of enforcing them. These concepts are often referred to as a policy decision point (PDP) and a policy enforcement point (PEP).

The policy database, which resides in the router's memory, is comprised of the set of policies loaded from local configuration and policies that have been read from LDAP. The policy database is built under the following conditions:

- Device reload or restart
- Talk 5 **reset database** command
- Automatic refresh
- SNMP set request

The policy database serves as the PDP, and consists of a set of policies that determine how the policy feature-related components handle packets. When a policy results in a decision (based on information such as the time of day, IP packet information, and protocol-specific information such as identification), the decision is passed to the enforcement component (PEP) to carry out the action. Figure 16 on page 204 shows the relationship of these components.

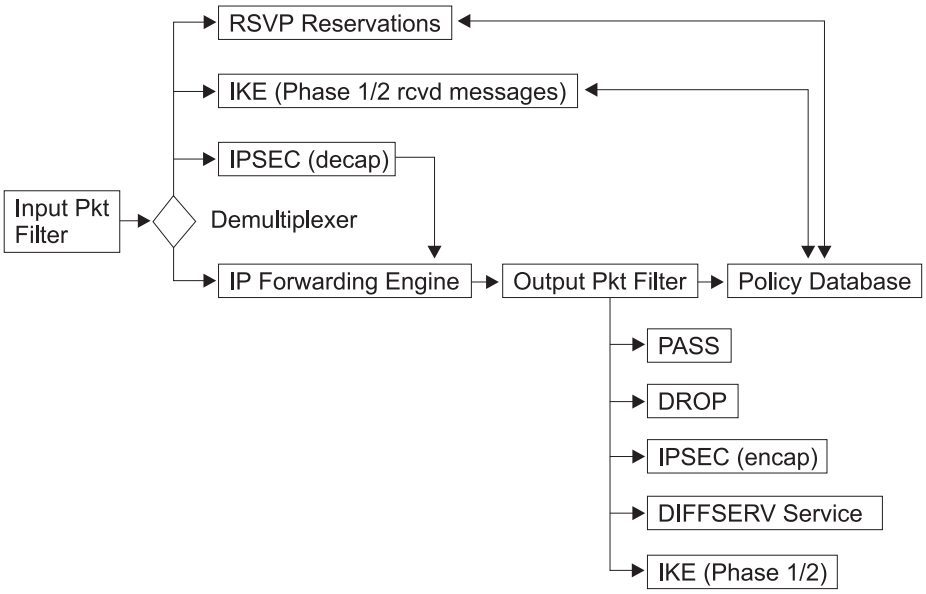


Figure 16. IP Packet Flow and the Policy Database

Policy Decision and Packet Flow

IP Packets first must pass the input packet filter before any other actions can be taken. If the input packet filter has rules present then the packet may have some action taken on it. If there is a filter match that excludes the packet or there is no match found in the input packet filter then the packet is dropped.

If the packet passes the input packet filter then it goes to a demultiplexing filter, which checks to see whether the packet is locally destined. If it is, then depending on the type of packet it is passed to other modules. These modules may be IPSec, IKE, RSVP, or others. If the packet is locally destined for IPSec, IKE, or RSVP then those modules may query the policy database to determine which action to take.

If the packet is not locally destined then it is given to the forwarding engine and a routing decision is made. If the routing decision does not drop the packet (Policy Based Routing may decide to drop the packet), then the packet goes to the output packet filter. If filter rules are present in the output packet then the packet may have address translation performed (NAT), may be passed or may be dropped. If no filter rules are present then the packet is passed. If filter rules are present and no match is found then the packet is dropped. If the packet passes the Output Packet filter then the IP Engine queries the policy database to determine whether any other actions should be performed on this packet.

Note: If the input and output packet filters are enabled for an interface(s), and packets that are to be controlled by the policy database are expected to traverse these interfaces, then a filter rule that includes these packets must be present in the input and output packet filters so they will not be dropped before the policy database is queried. One suggestion is to use to policy database to configure all the pass/drop rules and not to use the packet filters.

IP Policy Queries

When the IP forwarding engine queries the policy database, the following types of decision combinations may be returned:

- No match found—pass the packet
- Match found—drop the packet
- Match found—pass the packet
- Match found—secure the packet in IPSec manual tunnel x
- Match found—secure the packet in IKE negotiated IPSec tunnel x
- Match found—start ISAKMP negotiations for Phase 1 and 2, drop packet
- Match found—provide DiffServ QOS x, secure packet with IPSec

IPSec Policy Queries

If IPSec receives a packet then it must first decapsulate the packet and then decide whether the packet arrived in the correct IPSec tunnel (often referred to as the conformancy check). It does this by querying the policy database. The policy database may return the following types of decisions for this query:

- Conformancy check passed—forward the packet
- Conformancy check failed—drop the packet

IKE Policy Decisions

IKE may query the policy database and have the *Phase 1* IP policy decisions shown in Table 37 returned.

Table 37. IKE Phase 1 Queries and the Decisions Returned

Query Type	Decision
Message 1 (Main Mode)	No match found, drop packet
Message 1 (Main Mode)	Match found, negotiate with Phase 1 policy x
Message 5 (Main Mode)	No match found, stop negotiations with peer, drop packet
Message 5 (Main Mode)	No match found, stop negotiations with peer, drop packet
Message 5 (Main Mode)	Match found, policy x matched, finish Phase 1
Message 5 (Main Mode)	Match found, policy y matched, stop current Phase 1 and initiate new Phase 1 with new policy
Message 1 (Aggressive Mode)	No match found, drop packet
Message 1 (Aggressive Mode)	Match found, policy x matched

IKE may query the policy database and have the *Phase 2* IP policy decisions shown in Table 38 returned.

Table 38. IKE Phase 2 Queries and the Decisions Returned

Query Type	Decision
Message 2 (responder)	No match found, drop packet
Message 2 (responder)	Match found, negotiate with policy x

RSVP Policy Decisions

If a packet is an RSVP control message then RSVP queries the policy database to determine whether to accept or deny the reservation. If it is accepted then RSVP determines which attributes of the reservation to limit, based on the policy. Policies in the policy database can control the duration of the reservation, the amount of bandwidth that should be allocated, and the minimum delay to guarantee.

Policy Objects

A policy is made up of a profile, which contains a set of packet attributes upon which to base decisions, actions to take if a packet's attributes match those in the profile, and a validity period during which the decisions are made and the actions are enforced. These items are explained in greater detail in the following topics:

The parts that make up a policy are distinct named objects. Policy objects may refer to one another, and as a group of related items they comprise a policy. By separating configuration information into separate distinct objects, you can reuse many of them across multiple policy definitions, thus saving time and reducing maintenance efforts. Individual policy objects are discussed in detail in the following topics.

Policy

The policy object describes which conditionals should be checked against, and if the checks match, which actions to enforce. The policy makes named references to the validity period and the profile. For the policy to be valid, these references are required. The policy must also make a named reference to one or more of the following actions: an IPSec manual-keyed tunnel object, an IPSec action, an ISAKMP action, an RSVP action, or a DiffServ action. Valid combinations are:

- IPSec manual-keyed tunnel
- IPSec action to drop packets
- IPSec action to pass packets (no security)
- IPSec action to secure packets, ISAKMP action
- DiffServ action (drop)
- IPSec manual-keyed tunnel and DiffServ action (pass)
- IPSec action to secure packets, ISAKMP action, DiffServ action (pass)
- RSVP action
- RSVP action and DiffServ action (pass)

Note: In these combinations an IPSec manual tunnel cannot exist in the same policy definition as an IPSec action (IKE-negotiated IPSec tunnel), and an RSVP action must not be associated with any kind of IPSec action. If an IPSec action to secure packets is associated with a policy then you must also associate an ISAKMP action with the policy.

Each policy also has a priority number associated with it (the higher the number in the priority attribute, the higher the priority). The priority determines whether this policy takes precedence over another policy. Typically, you only have to set this if two or more policies' profiles conflict with each other in some way. The policy with the more specific profile should have a higher priority. For example, suppose that one policy specifies that traffic from subnet A to subnet B is to be secured with IPSec (DES) and another policy specifies that traffic from point a′ (a particular host inside of subnet A) to subnet B is to be secured with IPSec (3DES). The more specific policy (a′ to B) should have a higher priority than the policy with A to B.

It is a good idea to designate initial priority values that are 5 or more digits apart to allow room for specifying additional priority values for conflicting policies later. Each policy also has an enabled attribute, which determines whether the policy is to be enabled when loaded into the policy database. If a policy match is found during a policy database search but the policy is disabled, then the next most specific policy is enforced.

Profile

The profile determines which information is to be used to select a particular policy. The profile consists of source address and destination address information, protocol information, and source and destination port information.

Note: When defining policies for IPSec/ISAKMP, each gateway providing the security must have a policy to define the security association. The profile on each gateway must associate the source with the destination and the destination with the source. The profile for an IPSec policy must specify the source address as the traffic to be encapsulated into the tunnel and the destination address must be at the remote end of the tunnel.

The profile can also select based on the type-of-service (TOS) byte and the ingress and egress IP address. By default a packet received on any input interface and which leaves on any output interface is matched against the other selectors. In some cases, you may need the flexibility to specify exactly the interfaces on which the packet must arrive, and the interface on which the packet must leave. If you want this, then you must add interface-pair objects and associate the group name for the interface pair objects with the profile. You assign interface-pair objects to a group by giving them the same name. This allows you to specify combinations such as (any packet arriving on IPaddrX and leaving on any interface *OR* any packet coming in any interface and leaving on IPaddrX). This is particularly useful if you define a general drop rule for a public interface.

Interface Pair: Identifies the input interface and output interface. Specify the IP addresses for the interface for this selection. A value of 255.255.255.255 implies any interface.

If you want to use the profile to select an IPSec/ISAKMP policy, then you have the option of specifying the local ID to be sent during Phase 1, and the list of acceptable remote IDs during Phase 1 negotiations. By default, the local ID is the local tunnel endpoint for the IPSec/IKE traffic, and the remote ID list is *Any*. Optionally, you may specify the fully qualified domain name (FQDN), user FQDN, and key ID. Normally this is sufficient because all ISAKMP Phase 1 negotiations are authenticated with either public certificates or pre-shared keys. However, in some remote access situations in which the policy is wild-carded out for the destination addresses, it may be wise to specify a list of remote access users that are to be allowed access to network resources.

These users are still authenticated through the normal ISAKMP authentication methods, but the policy database performs an additional authentication step by ensuring that the local ID sent by the remote peer matches one of the IDs specified in the Remote User Group of the policy's profile. This is required if a public certificate authority (CA) is administering certificates to the general public, and the network administrator only wants a specific set of these users (for example, company employees) to have access. The remote user group is comprised of a list of users who belong to the same group. These users are entered by adding one or more *USER*s. A group of users can be making the group name for each user the same. This group can then optionally be associated with a profile.

Validity Period

The validity period specifies the life of the policy—the year, the months of the year, the days of the week, and the hours of the day that it is valid. This flexibility enables the network administrator to specify when a policy is valid, for example “all the time”

or “only this year, during the months of January, February and March, on Monday through Friday, from 9 AM to 5 PM.” When a policy in the policy database becomes invalid, the next most specific policy will be enforced. Thus you could define a policy that specifies on Monday through Friday from 9 am to 5 am to secure all traffic from subnet A to subnet B, and at any other time drop all traffic from subnet A to subnet B. In this case the first policy must have a higher priority (specified when you enter the Talk 5 **add policy** command).

DiffServ Action

The DiffServ action describes the quality of service that is to be provided to packets that match a policy that specifies a DiffServ action. You may configure the DiffServ action to drop packets. You may also use the DiffServ action to map packets into relative qualities of service. You may configure the bandwidth allocated as a percentage of output bandwidth or as an absolute value in Kbps. You must specify whether the best effort/assured queue or the premium queue is to provide the bandwidth allocation. For more information on these queues and how to define them, see “Chapter 20. Using the Differentiated Services Feature” on page 305 and “Chapter 21. Configuring and Monitoring the Differentiated Services Feature” on page 311.

The DiffServ action also specifies how to mark the TOS byte before it is sent on the egress interface. By default the TOS byte is not marked. It is useful to mark the packets at some point in the network based on the information in the IP packet header. Once the classification has been determined, since TOS byte marking has already been done, then the rest of the hops in the network can simply look at the new TOS byte to determine which QOS to apply to the packet. Looking at the TOS byte alone is much more efficient and may be necessary to achieve high performance in the DiffServ backbone.

RSVP Action

The RSVP action specifies whether to permit or deny RSVP flows if an RSVP reservation occurs and the reservation request matches the profile of the policy. If you want to permit the reservation, then the RSVP action also states the allowed duration of the reservation, the allowed bandwidth, and optionally, a reference to a DiffServ action. The reference to the DiffServ action enables RSVP to determine how to mark the TOS byte before the packet leaves the router. This is useful when packets pass from an RSVP network into a DiffServ network. RSVP can provide the QOS up to the RSVP boundary and then mark the TOS byte appropriately so the DiffServ network can apply the correct bandwidth.

IPSec Action

The IPSec action may specify either a drop, pass, or secure action. If the action is drop, then all packets matching this policy are dropped. If the action is pass with no security, then all packets are passed in the clear. If the action is pass with security, then all packets are secured by means of the security association (SA) specified by this action. The IPSec action also contains the IP addresses of the tunnel endpoints for the IPSec tunnel and IKE SAs.

The attributes of the SA are determined by the IPSec proposals that the IPSec action references. The IPSec action may specify multiple IPSec proposals and they are sent and checked against in the order they are specified. Having multiple

proposals in an IPSec action allows the configuration to contain all the acceptable combinations of security, thereby reducing the number of potential configuration mismatches between VPN gateways.

IPSec Proposal

The IPSec proposal contains the information about which ESP, AH, (or both) transform to propose or check against during Phase 2 ISAKMP negotiations. If you require perfect forward secrecy (a fresh Diffie Hellman calculation), then the IPSec proposal identifies which DH group to use. The transforms that the IPSec proposal references are sent or checked against in the order in which they are specified. The first ESP or AH transform in the list must be the one that is most appropriate to use. If more than one transform is in the list, then each one is compared to the peer's list of transforms to find a match. If none of the configured transforms match the peer's list then the negotiation fails. The IPSec proposal may list a combination of AH and ESP transforms, but the only valid combinations are:

- List of AH only (tunnel or transport mode)
- List of ESP only (tunnel or transport mode)
- List of AH (transport mode) and list of ESP (tunnel mode)

IPSec Transform

The attributes of the IPSec transform contain information about the IPSec encryption and authentication parameters and also specify how often the keys are refreshed. The transform is either AH (authentication only) or ESP (encryption, authentication, or both) and may be configured to operate in either tunnel or transport mode.

ISAKMP Action

The ISAKMP action specifies the key management information for Phase 1. It specifies whether the Phase 1 negotiations are to start in main mode (provides identity protection) or in aggressive mode. It also specifies whether the Phase 1 security association is to be negotiated at device start-up or on demand. The ISAKMP action also must reference one or more ISAKMP proposals. The first reference must be to the most acceptable ISAKMP proposal.

ISAKMP Proposal

The ISAKMP proposal specifies the encryption and authentication attributes of the Phase 1 security association. It also specifies which Diffie Hellman group to use to generate the keys, and the life of the Phase 1 security association. You must select the authentication method in the ISAKMP proposal. It can be either pre-shared key or certificate mode.

User

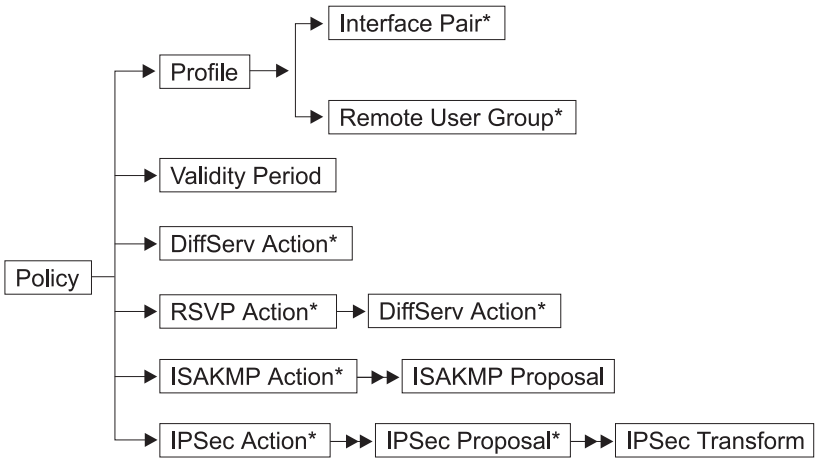
You must configure a USER for any policy that uses an ISAKMP negotiation with pre-shared key as the authentication method. The USER configuration identifies the pre-shared key to use for the ISAKMP peer. The user object contains the identifying information for a remote ISAKMP peer, that is IP address, FQDN, user FQDN or key ID, and which method the user wants to use for authentication. You may select either pre-shared key or certificate mode. If you select pre-shared key, then you must also specify whether the pre-shared key must be entered in ASCII or hexadecimal, and the value of the key. USERs may be grouped together by

assigning them to the same group name. This group can then optionally be associated with a policy's profile to perform a more strict policy lookup for Phase 1.

IPSec Manual-Keyed Tunnel

The IPSec manual-keyed tunnel is a static configuration of the encryption and authentication parameters. No negotiation is performed for the tunnel so both peers must have exactly the same configuration. The keys are actually entered as part of this configuration and must match on both sides of the tunnel. Since no negotiation is performed in this mode, the keys are never refreshed. For more information about IPSec manual-keyed tunnels, see the discussion of the IPSec feature in "Chapter 18. Using IP Security" on page 255.

Figure 17 shows the relationship between policy configuration objects.



Notes:

1. The → indicates a single reference.
2. The →→ indicates a multiple reference.
3. The * indicates an optional reference.
4. In a security policy for ISAKMP/IPSec, the traffic profile defines the traffic flowing into the secure tunnel.

Figure 17. Relationship of Policy Configuration Objects

LDAP and Policy Database Interaction

This family of routers allows a Lightweight Directory Access Protocol (LDAP) server to be the repository of policy information (the policy database). LDAP is a protocol that allows a directory server to be searched and modified. LDAP is a lightweight version of the X.500 standard. The routers support the ability to search for (but not modify) information in the directory server. The policy search agent in the router retrieves all the policy information in the directory server that is intended for that device. Any LDAP server operating at LDAP Version 2 or 3 works with the implementation in the router. An important advantage of using a directory server to store policy information as opposed to more traditional methods of locally stored configuration is the ability to make a change in one place and have that change applied across all the devices in the extended network. This includes devices in the administrative domain as well as devices across public boundaries.

For example, suppose you have an IPSec transform definition that resides in the directory. If you want to change the corporate policy for encryption from DES to 3DES, this would normally require a change in every device configuration across each network boundary. If you use the directory to deploy the policies then you only have to change one IPSec transform. Each policy-enabled device in your network would then need to rebuild the database. As another example, suppose you need to change a DiffServ action named “GoldService” to increase the bandwidth value from 40% to 45% of bandwidth. The LDAP server and policy infrastructure allow these types of configuration changes to scale much better and they reduce configuration mismatches.

If you are the network administrator, you may also take advantage of the ability to refresh the database automatically at a specified time each day. Select this option by entering the policy feature’s **set refresh** command. You may specify whether refreshing is enabled or not and, if enabled, the time at which the database is to refresh. This option is useful for making automated changes. For example, suppose that you must add a new policy so that the marketing department in the U.S. can talk to the development department in Japan across the Internet, and that the security gateways are SG1 and SG2. You can simply enter this information into the directory, and at midnight SG1 and SG2 automatically pick up this change if they are enabled for automatic refresh.

The LDAP policy search engine enables you to specify the security level to be used while building the policy database. You define these security options with the policy feature’s **set default** command. The options are:

- Pass all traffic during the search (default).
- Drop all traffic *except* LDAP policy search requests and results.
- Drop all traffic *except* LDAP policy search requests and results protected by IPSec.

In some situations either of the first two options are sufficient. However, if the LDAP traffic will traverse the public infrastructure, you should secure and authenticate the information by selecting the third option. If you do this, you must select Phase 1 and Phase 2 authentication and encryption options. You must also enter the IP addresses for the tunnel endpoints (primary and secondary LDAP servers). This boot-strap IKE/IPSec tunnel will be negotiated before any LDAP traffic is sent. This feature allows you to establish the configuration shown in Figure 18 on page 212.

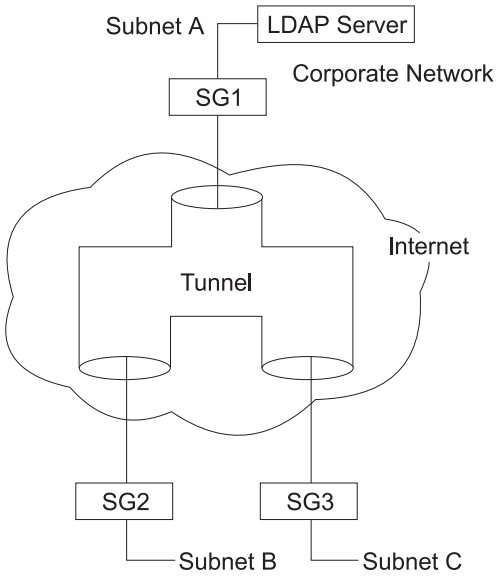


Figure 18. Securing Traffic Across the Internet

This figure shows an LDAP server on Subnet A in the corporate network. SG1, SG2, and SG3 are fetching their policies from the LDAP server. The policy search for SG2 and SG3 occurs across the Internet and is protected through IPSec.

The configuration information required for the policy database to successfully retrieve the policies from the directory is:

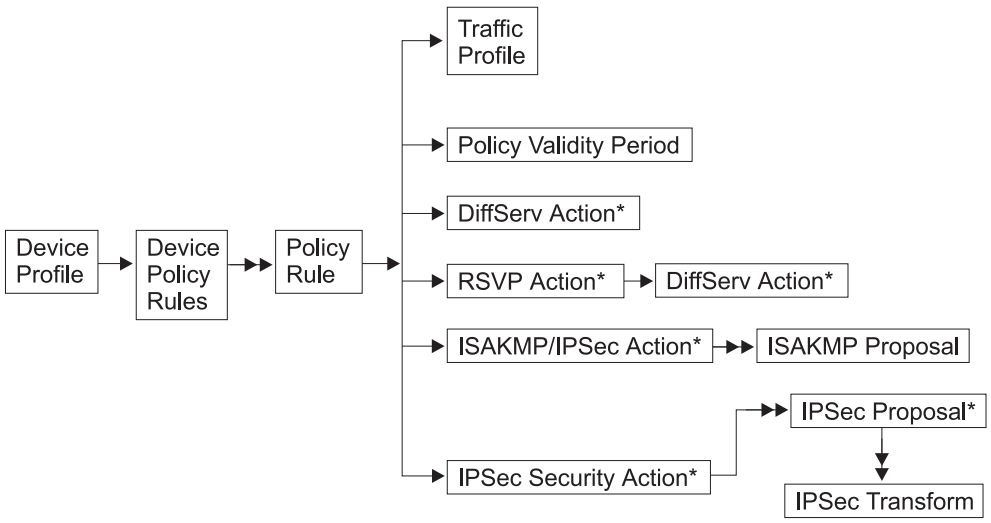
- Primary server IP address (a backup secondary server may also be configured)
- Port number on which the server is listening (Note: SSL and TLS are not supported)
- Username and password information if required
- Base distinguished name of the DeviceProfile object for this router or class of routers.
- Default policy information

After you have entered this configuration information, the next time the policy database is refreshed an attempt is made to interrogate the directory server for policy information. The policy database allows for a combination of locally configured policies and rules read from the LDAP server. If two rules are found to be conflicting and they are at the same priority, then the rule read from the local configuration take precedence over the rule read from the directory server.

Policy Schema

The LDAP schema is the set of rules and information making up the class and attribute definitions that determine the contents of entries in the directory. Typically the LDAP schema is written in ASN1 syntax, similar to SNMP MIBs. The policy schema that this family of routers supports is a work that comprises pre-standard efforts being done in the IETF. It is based on the standards track work being done by the IPSec and Policy Working Groups in the IETF and the Policy Working Group in the DMTF. The policy schema closely matches the existing configuration objects in the policy feature on the router. The policy schema definition files and LDAP server configuration files may be found by accessing the following URL: **http://www.networking.ibm.com/support**. Please select the router product you

want and then select the *Downloads* link. Figure 19 shows the overall structure of the policy schema.



Notes:

1. The → indicates a single reference.
2. The →→ indicates a multiple reference.
3. The * indicates an optional reference.
4. In a security policy for ISAKMP/IPSec, the traffic profile defines the traffic flowing into the secure tunnel.

Figure 19. Policy Schema Structure

The DeviceProfile and DevicePolicyRules are two key objects in the policy schema. They enable the policy search agent to locate the policies needed for the device. The DeviceProfile contains information about the device's administrative IP address and a mandatory DevicePolicyRules reference. You may group devices together into one DeviceProfile or each device in the network can have its own DeviceProfile. The choice you make depends on whether more than one device in the network must fetch the same set of rules. Typically, for security gateways this is not the case because every gateway has a different tunnel endpoint. For QOS-only devices, it is conceivable that all devices in a group would all read the same set of policies.

The DevicePolicyRules object is retrieved based on the value in the DeviceProfile that is fetched for the device. Once the DevicePolicyRules object has been retrieved, then the list of PolicyRules for that device can be retrieved. If any object is not found or if an error is detected during a consistency check on a object then the search is aborted and messages are displayed to the ELS (PLCY messages) identifying the error. If an error occurs, the network administrator may configure one of the following choices to handle it:

- Delete all locally read policies and revert to a drop or pass all rule
- Keep all locally read policies. Specify this option with the policy feature's **set default** command.

In either case, the search is attempted again at the configured retry interval. If the primary LDAP server cannot be contacted, then after 5 attempts the secondary

server is tried. If the secondary server cannot be reached, then after 5 attempts the primary server is tried again. You can specify the retry interval with the policy feature's **set ldap retry-interval** command. If a search is failing because of network latency, you may change the search timeout from the default of 3 seconds using the policy feature's **set ldap search-timeout** command.

Generating Rules

Configure a policy to specify how you want the network to operate. The router translates the policy information into a set of rules that it compares to traffic flows. In the past you may have done this manually by defining inbound and outbound packet filters for each traffic pattern. The policy database eliminates this, because with it you only configure a single policy.

Most of the work is done internally each time the policy database is built. In some cases a router translates a policy directly into a single rule. In the case of ISAKMP/IPSec, it translates a policy into five rules. Five rules are needed to account for the traffic directions (in and out) and for the control flows that occur during Phase 1 and Phase 2 of IKE negotiations. The relationship between policies and rules is as follows:

One DiffServ policy → One DiffServ rule

One RSVP policy → One RSVP rule

One ISAKMP/IPSec policy → Five ISAKMP/IPSec rules

Example: Secure the traffic from subnet A to subnet B; the tunnel endpoints are SGa and SGb

1. Phase 1 Inbound (Profile = SGb to SGa, Proto UDP, Src Port 500, Dst Port 500): This rule is needed to filter incoming Phase 1 negotiations from the remote ISAKMP peer if the device is functioning as an ISAKMP responder.
2. Phase 1 Outbound (Profile = SGa to SGb, Proto UDP, Src Port 500, Dst Port 500): This rule is needed to filter the Phase 1 information needed if traffic initiates ISAKMP Phase 1 negotiations. In this case the device is functioning as an ISAKMP initiator.
3. Phase 2 Inbound (Profile = SGb to SGa, Proto UDP, Src Port 500, Dst Port 500): This rule is needed to filter incoming Phase 2 traffic from the remote ISAKMP peer. This traffic is the result of the remote peer initiating a Phase 2 refresh or initial negotiation. A Phase 2 outbound rule is not needed since the outbound traffic (rule 5) always starts the negotiations if needed.
4. Traffic Into the Secure Tunnel (Profile = Subnet A to Subnet B): This rule is needed to put unprotected traffic into a secure tunnel. If the security association has not been negotiated, then the Phase 1 rule is also gathered and IKE starts Phase 1 and Phase 2. Once the SAs have been established, then packets matching this rule are given to IPSec for encapsulation and transmission.
5. Traffic From the Secure Tunnel (Profile = Subnet B to Subnet A): This rule is needed to ensure that packets that should have arrived in a secure tunnel did indeed arrive in a secure tunnel. If the packet was not decapsulated by

IPSec and encounters this rule, then the packet is dropped. This rule handles any traffic that is spoofed into the network.

One IPSec manual-keyed tunnel → Two IPSec rules

Example: Secure the traffic from subnet A to subnet B; the tunnel endpoints are SGa and SGb.

1. Traffic Into the Secure Tunnel (Profile = Subnet A to Subnet B): This rule is needed to put unprotected traffic into a secure tunnel. This is a statically configured tunnel so it is always available, and packets matching this rule are given directly to IPSec for encapsulation and transmission.
2. Traffic From the Secure Tunnel (Profile = Subnet B to Subnet A): This rule is needed to ensure that packets that should have arrived in a secure tunnel did indeed arrive in a secure tunnel. If the packet was not decapsulated by IPSec and encounters this rule, then the packet is dropped. This rule handles any traffic that is spoofed into the network.

You may view these rules using the policy feature's Talk 5 **list rule** command.

Configuration Examples

The following examples show how you can use the policy feature to configure the routers in a network. First, access the policy feature as shown:

```
* talk 6
Config>feature policy
IP Network Policy configuration
```

IPSec/ISAKMP Policy with QOS

You may enter policy information in either of two ways. The first way is to define the individual policy objects and then group them together. To use this method, first define the IPSec transforms, then the IPSec proposal (which refers to the IPSec transforms). Then define the IPSec action (which refers to the IPSec proposals), and so forth until you completely define the policy. Using Figure 20 on page 216 as a reference, this method starts at the right side of the policy objects and works its way to the left.

The second approach, which you may find easier, is to define the high-level policy options first, and as you are prompted, enter the definitions for the individual policy objects as you go along. A sample configuration procedure follows Figure 20 on page 216, and uses values that correspond to those in the figure. It uses the left-to-right method and starts with the **add policy** command.

If an object was defined previously that meets your needs, then you can reuse it instead of creating a new definition. For example, if a validity period for allTheTime was configured for a previous policy, then you may reuse it. The following procedure shows the entire process, but does not demonstrate the reuse of previously defined policy information. For an example of using previously defined information, see "IPSec/ISAKMP Only Policy" on page 224.

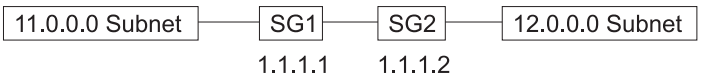


Figure 20. Configuring IPSec/ISAKMP with QOS

The policy configuration scenario described in the following text is from SG1's perspective. The policy statement is:

Secure the traffic from subnet 11 to subnet 12 with the tunnel endpoints being SG1 and SG2, and provide a QOS for the traffic in this tunnel by means of DiffServ GoldService

1. Add the policy.

```
Policy config>add policy
Enter a Name (1-29 characters) for this Policy []? examplePolicySecure11to12
Enter the priority of this policy (This number is used to
determine the policy to enforce in the event of policy conflicts) [5]? 10
```

2. No profiles are configured so you must define a new one.

```
List of Profiles:
0: New Profile

Enter number of the profile for this policy [0]?
```

3. New profile definition; in this case the traffic we are interested in is from subnet 11 to subnet 12.

```
Enter a Name (1-29 characters) for this Profile []? trafficFrom11NetTo12Net
Source Address Format (1:NetMask, 2:Range, 3:Single Addr) [1]?
Enter IPV4 Source Address [0.0.0.0]? 11.0.0.0
Enter IPV4 Source Mask [255.0.0.0]?
Destination Address Format (1:NetMask, 2:Range, 3:Single Addr) [1]?
Enter IPV4 Destination Address [0.0.0.0]? 12.0.0.0
Enter IPV4 Destination Mask [255.0.0.0]?

Protocol IDs:
     1)  TCP
     2)  UDP
     3)  All Protocols
     4)  Specify Range

Select the protocol to filter on (1-4) [3]?
Enter the Starting value for the Source Port [0]?
Enter the Ending value for the Source Port [65535]?
Enter the Starting value for the Destination Port [0]?
Enter the Ending value for the Destination Port [65535]?
Enter the Mask to be applied to the Received DS-byte [0]?
Enter the value to match against after the Mask has
been applied to the Received DS-byte [0]?
Configure local and remote ID's for ISAKMP? [No]:
Limit this profile to specific interface(s)? [No]:


Here is the Profile you specified...


Profile Name    = trafficFrom11NetTo12Net
sAddr:Mask=       11.0.0.0 : 255.0.0.0       sPort=    0 : 65535
dAddr:Mask=       12.0.0.0 : 255.0.0.0       dPort=    0 : 65535
proto     =              0 : 255
TOS       =            x00 : x00
Remote Grp=All Users
Is this correct? [Yes]:
```

4. Finished with the profile definition and have returned to the policy configuration menu.

```
List of Profiles:
0: New Profile
1: trafficFrom11NetTo12Net

Enter number of the profile for this policy [1]? 1
```

5. No validity periods are configured so you must define a new one.

```
List of Validity Periods:
0: New Validity Period

Enter number of the validity period for this policy [0]?
```

6. Validity period configuration questions; in this example the validity period is from 9 AM to 5 PM, Monday through Friday, every month of 1999.

```
Enter a Name (1-29 characters) for this Policy Valid Profile []?
              MonToFri-9am:5pm-1999
Enter the lifetime of this policy. Please input the
information in the following format:
yyyymmddhhmmss:yyyymmddhhmmss OR '*' denotes forever.
 [*]? 19990101000000:19991231000000
During which months should policies containing this profile
be valid.  Please input any sequence of months by typing in
the first three letters of each month with a space in between
each entry, or type ALL to signify year round.
 [ALL]?
During which days should policies containing this profile
be valid.  Please input any sequence of days by typing in
the first three letters of each day with a space in between
each entry, or type ALL to signify all week
 [ALL]? mon tue wed thu fri
Enter the starting time (hh:mm:ss or * denotes all day)
 [*]? 00:00:00
Enter the ending time (hh:mm:ss)
 [00:00:00]? 17:00:00

Here is the Policy Validity Profile you specified...

Validity Name   = MonToFri-9am:5pm-1999
Duration  = 19990101000000 : 19991231000000
Months    = ALL
Days      = MON TUE WED THU FRI
Hours     = 09:00:00 : 17:00:00
Is this correct? [Yes]:
```

7. Finished with the validity period definition and have returned to the policy configuration menu.

```
List of Validity Periods:
0: New Validity Period
1: MonToFri-9am:5pm-1999

Enter number of the validity period for this policy [1]? 1
Should this policy enforce an IPSEC action? [No]: yes
```

8. Should always define a new IPSec action because the tunnel endpoint will always be different. The exceptions to this are if there are multiple tunnels between the same two gateways, and in the wildcarded remote access configurations where the tunnel endpoint is unknown.

```
IPSEC Actions:
0: New IPSEC Action

Enter the Number of the IPSEC Action [0]?
```

9. IPSec action menu.

```
Enter a Name (1-29 characters) for this IPsec Action []? secure11NetTo12Net
List of IPsec Security Action types:
     1)  Block (block connection)
     2)  Permit

Select the Security Action type (1-2) [2]? 2
Should the traffic flow into a secure tunnel or in the clear:
     1) Clear
     2) Secure Tunnel
 [2]?
Enter Tunnel Start Point IPV4 Address
 [11.0.0.5]? 1.1.1.1
Enter Tunnel End Point IPV4 Address (0.0.0.0 for Remote Access)
 [0.0.0.0]? 1.1.1.2
Does this IPSEC tunnel flow within another IPSEC tunnel? [No]:
Percentage of SA lifesize/lifetime to use as the acceptable minimum [75]?

Security Association Refresh Threshold, in percent (1-100) [85]?
Options for DF Bit in outer header (tunnel mode):
     1)  Copy
     2)  Set
     3)  Clear
Enter choice (1-3) [1]?
Enable Replay prevention (1=enable, 2=disable) [2]?
Do you want to negotiate the security association at
system initialization(Y-N)? [No]:
You must choose the proposals to be sent/checked against during phase 2
negotiations.  Proposals should be entered in order of priority.
```

10. No IPSec proposals defined so you must define a new one. Note that once the IPSec proposal has been defined it can be reused across multiple IPSec actions.

```
List of IPSEC Proposals:
0: New Proposal

Enter the Number of the IPSEC Proposal [0]?
```

11. IPSec proposal configuration.

```
Enter a Name (1-29 characters) for this IPsec Proposal []? genP2Proposal
Does this proposal require Perfect Forward Secrecy?(Y-N)? [No]:
Do you wish to enter any AH transforms for this proposal? [No]:
Do you wish to enter any ESP transforms for this proposal? [No]: yes
```

12. No ESP transforms are configured so you must define a new one. Once the ESP transform has been defined it may be reused by any IPSec proposal.

```
List of ESP Transforms:
0: New Transform

Enter the Number of the ESP transform [0]? 0
```

13. IPSec transform configuration.

```
Enter a Name (1-29 characters) for this IPsec Transform []? esp3DESwSHA
List of Protocol IDs:
     1)  IPSEC AH
     2)  IPSEC ESP

Select the Protocol ID (1-2) [1]? 2
```

```
List of Encapsulation Modes:
     1)  Tunnel
     2)  Transport

Select the Encapsulation Mode(1-2) [1]? 1
List of IPsec Authentication Algorithms:
     0)  None
     1)  HMAC-MD5
     2)  HMAC_SHA

Select the ESP Authentication Algorithm (0-2) [2]? 2
List of ESP Cipher Algorithms:
     1)  ESP DES
     2)  ESP 3DES
     3)  ESP CDMF
     4)  ESP NULL

Select the ESP Cipher Algorithm (1-4) [1]? 2
Security Association Lifesize, in kilobytes (1024-65535) [50000]?
Security Association Lifetime, in seconds (120-65535) [3600]?

Here is the IPSec transform you specified...

Transform Name  = esp3DESwSHA
Type =ESP   Mode =Tunnel     LifeSize=   50000 LifeTime=    3600
Auth =SHA   Encr =3DES
Is this correct? [Yes]:
```

14. Return to the IPSec proposal menu.

```
List of ESP Transforms:
0: New Transform
1: esp3DESwSHA

Enter the Number of the ESP transform [1]?
Do you wish to add another ESP transform to this proposal? [Yes]: no

Here is the IPSec proposal you specified...

Name  = genP2Proposal
Pfs   = N
ESP Transforms:
esp3DESwSHA
Is this correct? [Yes]:
```

15. Return to the IPSec action menu.

```
List of IPSEC Proposals:
0: New Proposal
1: genP2Proposal

Enter the Number of the IPSEC Proposal [1]?
Are there any more Proposal definitions for this IPSEC Action? [No]:

Here is the IPSec Action you specified...

IPSECAction Name = secure11NetTo12Net
Tunnel Start:End         =        1.1.1.1 : 1.1.1.2
Tunnel In Tunnel         =             No
Min Percent of SA Life   =             75
Refresh Threshold        =             85 %
Autostart                =             No
DF Bit                   =           COPY
Replay Prevention        =       Disabled
IPSEC Proposals:
genP2Proposal
Is this correct? [Yes]:
```

16. Return to the policy menu.

```
IPSEC Actions:
0: New IPSEC Action
1: secure11NetTo12Net

Enter the Number of the IPSEC Action [1]? 1
```

17. You have specified a secure IPSec action type, so you must identify an ISAKMP action for the Phase 1 negotiations. None are defined, so you must enter a new one. In most cases, one ISAKMP action and proposal is sufficient for all of the security policies.

```
ISAKMP Actions:
0: New ISAKMP Action

Enter the Number of the ISAKMP Action [0]?
```

18. ISAKMP action configuration.

```
Enter a Name (1-29 characters) for this ISAKMP Action []? genPhase1Action

List of ISAKMP Exchange Modes:
     1)  Main
     2)  Aggressive

Enter Exchange Mode (1-2) [1]?
Percentage of SA lifesize/lifetime to use as the acceptable minimum [75]?

ISAKMP Connection Lifesize, in kilobytes (100-65535) [5000]?
ISAKMP Connection Lifetime, in seconds (120-65535) [30000]?
Do you want to negotiate the security association at
system initialization(Y-N)? [Yes]: no
You must choose the proposals to be sent/checked against during phase 1
negotiations.  Proposals should be entered in order of priority.
```

19. No ISAKMP proposals are configured, so you must create a new one.

```
List of ISAKMP Proposals:
0: New Proposal
```

20. ISAKMP proposal configuration.

```
Enter the Number of the ISAKMP Proposal [0]?
Enter a Name (1-29 characters) for this ISAKMP Proposal []? genP1Proposal

List of Authentication Methods:
     1)  Pre-Shared Key
     2)  RSA SIG

Select the authentication method (1-2) [1]? 2

List of Hashing Algorithms:
     1)  MD5
     2)  SHA

Select the hashing algorithm(1-2) [1]? 2

List of Cipher Algorithms:
     1)  DES
     2)  3DES

Select the Cipher Algorithm (1-2) [1]? 2
Security Association Lifesize, in kilobytes (100-65535) [1000]?
Security Association Lifetime, in seconds (120-65535) [15000]?

List of Diffie Hellman Groups:
     1)  Diffie Hellman Group 1
     2)  Diffie Hellman Group 2

Select the Diffie Hellman Group ID from this proposal (1-2) [1]?
```

```
Here is the ISAKMP Proposal you specified...

Name = genP1Proposal
AuthMethod = Pre-Shared Key
LifeSize   = 1000
LifeTime   = 15000
DHGroupID  = 1
Hash Algo  = SHA
Encr Algo  = 3DES CB
Is this correct? [Yes]:
```

21. Return to the ISAKMP action configuration.

```
List of ISAKMP Proposals:
0: New Proposal
1: genP1Proposal

Enter the Number of the ISAKMP Proposal [1]?
Are there any more Proposal definitions for this ISAKMP Action? [No]:

Here is the ISAKMP Action you specified...

ISAKMP Name     = genPhase1Action
Mode                     =            Main
Min Percent of SA Life   =              75
Conn LifeSize:LifeTime   =            5000 : 30000
Autostart                =              No
ISAKMP Proposals:
genP1Proposal
Is this correct? [Yes]:
```

22. Return to the policy configuration.

```
ISAKMP Actions:
0: New ISAKMP Action
1: genPhase1Action

Enter the Number of the ISAKMP Action [1]?
Do you wish to Map a DiffServ Action to this Policy? [No]: yes
```

23. Define the DiffServ GoldService action.

```
DiffServ Actions:
0: New DiffServ Action

Enter the Number of the DiffServ Action [0]?
```

24. DiffServ action configuration.

```
Enter a Name (1-29 characters) for this DiffServ Action []? GoldService
Enter the permission level for packets matching this DiffServ
Action (1. Permit, 2. Deny) [2]? 1
List of DiffServ Queues:
   1) Premium
   2) Assured/BE
Enter the Queue Number(1-2) for outgoing packets matching
this DiffServ Action [2]? 2
How do you want to specify the bandwidth allocated to this service?
Enter absolute kbps(1) or percentage of output bandwidth(2) [2]?
Enter the percentage of output bandwidth allocated to this service [10]? 40

Transmitted DS-byte mask [0]?
Transmitted DS-byte modify value [0]?

Here is the DiffServ Action you specified...

DiffServ Name   = GoldService                    Type =Permit
```

```
TOS mask:modify=x00:x00
Queue:BwShare  =Assured    : 40 %
Is this correct? [Yes]:
```

25. Return to the policy configuration.

```
DiffServ Actions:
0: New DiffServ Action
1: GoldService

Enter the Number of the DiffServ Action [1]? 1
Policy Enabled/Disabled (1. Enabled, 2. Disabled) [1]?

Here is the Policy you specified...

Policy Name     = examplePolicySecure11to12
State:Priority =Enabled    : 10
Profile        =trafficFrom10NetTo12Net
Valid Period   =MonToFri-9am:5pm-1999
IPSEC Action   =secure11NetTo12Net
ISAKMP Action  =genPhase1Action
DiffServ Action=GoldService
Is this correct? [Yes]:
```

26. If DiffServ or IPSec is not enabled, then you are alerted that before the policy can be enforced, you must enable DiffServ, IPSec, or both (DiffServ feature or IPSec feature).

```
You must enable and configure DiffServ in feature DS before
QOS can be ensured for this policy
```

27. The final step in this process is to add a USER profile definition for the remote ISAKMP peer. This step is not needed if the ISAKMP negotiations are to authenticate the peer with public certificates. However in the preceding example we chose pre-shared key as the authentication method, so we must identify the user and enter the pre-shared key that we expect the peer to use.

```
Policy config>add user
Choose from the following ways to identify a user:
1: IP Address
2: Fully Qualified Domain Name
3: User Fully Qualified Domain Name
4: Key ID (Any string)
Enter your choice(1-4) [1]?
Enter the IP Address that distinguishes this user
 [0.0.0.0]? 1.1.1.2
Group to include this user in []? peers
Authenticate user with 1:pre-shared key or 2: Public Certificate [1]?
Mode to enter key (1=ASCII, 2=HEX) [1]?
Enter the Pre-Shared Key (an even number of 2-128 ascii chars):
Enter the Pre-Shared Key again (10 characters) in ascii:

Here is the User Information you specified...

Name      = 1.1.1.2
Type      = IPV4 Addr
Group     =peers
Auth Mode =Pre-Shared Key
Key(Ascii)=exampleKey
Is this correct? [Yes]:
```

28. The policy configuration steps are now complete. If you want to configure DiffServ, IPSec, or any network or IP configuration, then you must do that before the IPSec tunnel will be functional. The following list command example

shows the configuration that was just completed. To activate these changes, either reload the device or enter the policy feature's Talk 5 **reset database** command.

```
Policy config>list all

Configured Policies....

Policy Name     = examplePolicySecure11to12
State:Priority =Enabled    : 10
Profile        =trafficFrom11NetTo12Net
Valid Period   =MonToFri-9am:5pm-1999
IPSEC Action   =secure11NetTo12Net
ISAKMP Action  =genPhase1Action
DiffServ Action=GoldService
--More--

Configured Profiles....

Profile Name    = trafficFrom11NetTo12Net
sAddr:Mask=       11.0.0.0 : 255.0.0.0       sPort=    0 : 65535
dAddr:Mask=       12.0.0.0 : 255.0.0.0       dPort=    0 : 65535
proto     =             0 : 255
TOS       =          x00 : x00
Remote Grp=All Users
--More--

Configured Validity Periods

Validity Name   = MonToFri-9am:5pm-1999
Duration  = 19990101000000 : 19991231000000
Months    = ALL
Days      = MON TUE WED THU FRI
Hours     = 09:00:00 : 17:00:00
--More--

Configured DiffServ Actions....

DiffServ Name   = GoldService                    Type =Permit

TOS mask:modify=x00:x00
Queue:BwShare  =Assured    : 40 %
--More--

Configured IPSEC Actions....

IPSECAction Name = secure11NetTo12Net
Tunnel Start:End         =        1.1.1.1 : 1.1.1.2
Tunnel In Tunnel         =             No
Min Percent of SA Life   =             75
Refresh Threshold        =             85 %
Autostart                =             No
DF Bit                   =           COPY
Replay Prevention        =       Disabled
IPSEC Proposals:
genP2Proposal
--More--

Configured IPSEC Proposals....

Name  = genP2Proposal
Pfs   = N
ESP Transforms:
esp3DESwSHA
--More--

Configured IPSEC Transforms....

Transform Name  = esp3DESwSHA
Type =ESP   Mode =Tunnel     LifeSize=   50000 LifeTime=    3600
Auth =SHA   Encr =3DES
--More--

Configured ISAKMP Actions....
```

```
ISAKMP Name     = genPhase1Action
Mode                    =            Main
Min Percent of SA Life  =              75
Conn LifeSize:LifeTime  =            5000 : 30000
Autostart               =              No
ISAKMP Proposals:
genP1Proposal
--More--

Configured ISAKMP Proposals....
Name = genP1Proposal
AuthMethod = Pre-Shared Key
LifeSize   = 1000
LifeTime   = 15000
DHGroupID  = 1
Hash Algo  = SHA
Encr Algo  = 3DES CB
--More--

Configured Policy Users....
Name      = 1.1.1.2
Type      = IPV4 Addr
Group     =peers
Auth Mode =Pre-Shared Key
Key(Ascii)=exampleKey
--More--

Configured Manual IPSEC Tunnels....

                              IPv4 Tunnels
-------------------------------------------------------------------------------

   ID         Name        Local IPv4 Addr  Rem IPv4 Addr    Mode    State
 ------  ---------------  ---------------  ---------------  -----  --------
```

IPSec/ISAKMP Only Policy

A sample configuration procedure, which follows Figure 21 and uses values that correspond to those in the figure, uses the left-to-right method and shows how to build on the previous sample procedure by reusing information that the previous one created.

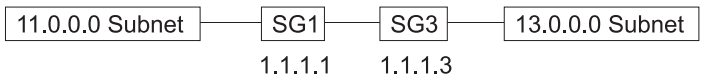


Figure 21. Configuring IPSec and Reusing a Previous Definition

The policy configuration scenario described in the following text is from SG1's perspective. The policy statement in this scenario is:

Secure the traffic from subnet 11 to subnet 13 (TCP traffic only) with the tunnel endpoints being SG1 and SG3, and provide no QOS.

1. Add the policy.

```
Policy config>add policy
Enter a Name (1-29 characters) for this Policy []? examplePolicySecure11to13
Enter the priority of this policy (This number is used to
determine the policy to enforce in the event of policy conflicts) [5]? 10
List of Profiles:
0: New Profile
1: trafficFrom10NetTo12Net

Enter number of the profile for this policy [1]? 0
Enter a Name (1-29 characters) for this Profile []? trafficFrom11NetTo13Net
```

```
Source Address Format (1:NetMask, 2:Range, 3:Single Addr) [1]?
Enter IPV4 Source Address [0.0.0.0]? 11.0.0.0
Enter IPV4 Source Mask [255.0.0.0]?
Destination Address Format (1:NetMask, 2:Range, 3:Single Addr) [1]?
Enter IPV4 Destination Address [0.0.0.0]? 13.0.0.0
Enter IPV4 Destination Mask [255.0.0.0]?

Protocol IDs:
     1)  TCP
     2)  UDP
     3)  All Protocols
     4)  Specify Range

Select the protocol to filter on (1-4) [3]? 1
Enter the Starting value for the Source Port [0]?
Enter the Ending value for the Source Port [65535]?
Enter the Starting value for the Destination Port [0]?
Enter the Ending value for the Destination Port [65535]?
Enter the Mask to be applied to the Received DS-byte [0]?
Enter the value to match against after the Mask has
been applied to the Received DS-byte [0]?
Configure local and remote ID's for ISAKMP? [No]:
Limit this profile to specific interface(s)? [No]:

Here is the Profile you specified...

Profile Name    = trafficFrom11NetTo13Net
sAddr:Mask=       11.0.0.0 : 255.0.0.0       sPort=    0 : 65535
dAddr:Mask=       13.0.0.0 : 255.0.0.0       dPort=    0 : 65535
proto     =              6 : 6
TOS       =            x00 : x00
Remote Grp=All Users
Is this correct? [Yes]:
List of Profiles:
0: New Profile
1: trafficFrom10NetTo12Net
2: trafficFrom11NetTo13Net

Enter number of the profile for this policy [1]? 2
```

2. Reuse the validity period.

```
List of Validity Periods:
0: New Validity Period
1: MonToFri-9am:5pm-1999

Enter number of the validity period for this policy [1]?
Should this policy enforce an IPSEC action? [No]: yes
IPSEC Actions:
0: New IPSEC Action
1: secure11NetTo12Net

Enter the Number of the IPSEC Action [1]? 0
Enter a Name (1-29 characters) for this IPsec Action []? secure11To13
List of IPsec Security Action types:
     1)  Block (block connection)
     2)  Permit

Select the Security Action type (1-2) [2]?
Should the traffic flow into a secure tunnel or in the clear:
     1) Clear
     2) Secure Tunnel
 [2]?
Enter Tunnel Start Point IPV4 Address
 [11.0.0.5]? 1.1.1.1
Enter Tunnel End Point IPV4 Address (0.0.0.0 for Remote Access)
```

```
 [0.0.0.0]? 1.1.1.3
Does this IPSEC tunnel flow within another IPSEC tunnel? [No]:
Percentage of SA lifesize/lifetime to use as the acceptable minimum [75]?

Security Association Refresh Threshold, in percent (1-100) [85]?
Options for DF Bit in outer header (tunnel mode):
     1)  Copy
     2)  Set
     3)  Clear
Enter choice (1-3) [1]?
Enable Replay prevention (1=enable, 2=disable) [2]?
Do you want to negotiate the security association at
system initialization(Y-N)? [No]:
You must choose the proposals to be sent/checked against during phase 2
negotiations.  Proposals should be entered in order of priority.
```

3. Reuse the IPSec proposal from the previously defined configuration.

```
List of IPSEC Proposals:
0: New Proposal
1: genP2Proposal

Enter the Number of the IPSEC Proposal [1]?
Are there any more Proposal definitions for this IPSEC Action? [No]:


Here is the IPSec Action you specified...


IPSECAction Name = secure11To13
Tunnel Start:End         =        1.1.1.1 : 1.1.1.3
Tunnel In Tunnel         =             No
Min Percent of SA Life   =             75
Refresh Threshold        =             85 %
Autostart                =             No
DF Bit                   =           COPY
Replay Prevention        =       Disabled
IPSEC Proposals:
genP2Proposal
Is this correct? [Yes]:
IPSEC Actions:
0: New IPSEC Action
1: secure11NetTo12Net
2: secure11To13

Enter the Number of the IPSEC Action [1]? 2
```

4. Reuse the ISAKMP action from the previous configuration.

```
ISAKMP Actions:
0: New ISAKMP Action
1: genPhase1Action

Enter the Number of the ISAKMP Action [1]?
Do you wish to Map a DiffServ Action to this Policy? [No]:
Policy Enabled/Disabled (1. Enabled, 2. Disabled) [1]?

Here is the Policy you specified...

Policy Name     = examplePolicySecure11to13
State:Priority =Enabled    : 10
Profile        =trafficFrom11NetTo13Net
Valid Period   =MonToFri-9am:5pm-1999
```

```
IPSEC Action   =secure11To13
ISAKMP Action  =genPhase1Action
Is this correct? [Yes]:
```

Drop All Public Traffic (Filter Rule)

This policy example shows how to configure a simple drop rule for the public interface that drops all traffic that is not secured through IPSec. This rule is very general and **must** have the lowest priority of any rule configured.

1. Add the policy.

```
Policy config>add policy
Enter a Name (1-29 characters) for this Policy []? dropAllPublicTraffic
Enter the priority of this policy (This number is used to
determine the policy to enforce in the event of policy conflicts) [5]?
List of Profiles:
0: New Profile
1: trafficFrom10NetTo12Net
2: trafficFrom11NetTo13Net

Enter number of the profile for this policy [1]? 0
```

2. Define a new profile that includes all traffic going in or out the public interface (1.1.1.1).

```
Enter a Name (1-29 characters) for this Profile []? allPublicTraffic
Source Address Format (1:NetMask, 2:Range, 3:Single Addr) [1]?
Enter IPV4 Source Address [0.0.0.0]?
Enter IPV4 Source Mask [0.0.0.0]?
Destination Address Format (1:NetMask, 2:Range, 3:Single Addr) [1]?
Enter IPV4 Destination Address [0.0.0.0]?
Enter IPV4 Destination Mask [0.0.0.0]?

Protocol IDs:
     1)  TCP
     2)  UDP
     3)  All Protocols
     4)  Specify Range

Select the protocol to filter on (1-4) [3]?
Enter the Starting value for the Source Port [0]?
Enter the Ending value for the Source Port [65535]?
Enter the Starting value for the Destination Port [0]?
Enter the Ending value for the Destination Port [65535]?
Enter the Mask to be applied to the Received DS-byte [0]?
Enter the value to match against after the Mask has
been applied to the Received DS-byte [0]?
Configure local and remote ID's for ISAKMP? [No]:
```

3. Since the source and destination (or both) information has been wild-carded out, you must specify the interfaces on which you expect this traffic to arrive and leave.

```
The Source and/or Destination Address information you specified
includes all addresses.  You must specify an Interface Pair
with this profile to further qualify what traffic you wish to filter
to this policy.  The interface pair should at least specify the
Limit this profile to specific interface(s)? [No]: yes
Interface Pair Groups:
```

```
0: New Ifc Pair
Number of Ifc Pair Group [1]? 0
```

4. Add an interface-pair for traffic going out over the public interface.

```
Enter a Group Name (1-29 characters) for this Interface Pair []? inOutPublic
Ingress Interface IP Address (255.255.255.255 = any ingress)
 [255.255.255.255]?
Egress Interface IP Address (255.255.255.255 = any egress)
 [255.255.255.255]? 1.1.1.1
Interface Pair Groups:
0: New Ifc Pair
1) Group Name: inOutPublic
  In:Out=255.255.255.255 : 1.1.1.1

Number of Ifc Pair Group [1]? 0
```

5. Add another interface-pair for traffic coming in over the public interface. Give it the same name as the previous interface pair to assign it to the same group.

```
Enter a Group Name (1-29 characters) for this Interface Pair []? inOutPublic
Ingress Interface IP Address (255.255.255.255 = any ingress)
 [255.255.255.255]? 1.1.1.1
Egress Interface IP Address (255.255.255.255 = any egress)
 [255.255.255.255]?
Interface Pair Groups:
0: New Ifc Pair
1) Group Name: inOutPublic
  In:Out=255.255.255.255 : 1.1.1.1
  In:Out=        1.1.1.1 : 255.255.255.255

Number of Ifc Pair Group [1]?

Here is the Profile you specified...

Profile Name    = allPublicTraffic
sAddr:Mask=        0.0.0.0 : 0.0.0.0         sPort=    0 : 65535
dAddr:Mask=        0.0.0.0 : 0.0.0.0         dPort=    0 : 65535
proto     =              0 : 255
TOS       =            x00 : x00
Remote Grp=All Users
1.  In:Out=255.255.255.255 : 1.1.1.1
2.  In:Out=        1.1.1.1 : 255.255.255.255
Is this correct? [Yes]:
List of Profiles:
0: New Profile
1: trafficFrom10NetTo12Net
2: trafficFrom11NetTo13Net
3: allPublicTraffic

Enter number of the profile for this policy [1]? 3
```

6. Add a new validity period that specifies `all the time`.

```
List of Validity Periods:
0: New Validity Period
1: MonToFri-9am:5pm-1999

Enter number of the validity period for this policy [1]? 0
Enter a Name (1-29 characters) for this Policy Valid Profile []? allTheTime
Enter the lifetime of this policy. Please input the
information in the following format:
yyyymmddhhmmss:yyyymmddhhmmss OR '*' denotes forever.
 [*]?
```

```
During which months should policies containing this profile
be valid.  Please input any sequence of months by typing in
the first three letters of each month with a space in between
each entry, or type ALL to signify year round.
 [ALL]?
During which days should policies containing this profile
be valid.  Please input any sequence of days by typing in
the first three letters of each day with a space in between
each entry, or type ALL to signify all week
 [ALL]?
Enter the starting time (hh:mm:ss or * denotes all day)
 [*]?

Here is the Policy Validity Profile you specified...

Validity Name   = allTheTime
Duration  = Forever
Months    = ALL
Days      = ALL
Hours     = All Day
Is this correct? [Yes]:
List of Validity Periods:
0: New Validity Period
1: MonToFri-9am:5pm-1999
2: allTheTime

Enter number of the validity period for this policy [1]? 2
Should this policy enforce an IPSEC action? [No]: yes
IPSEC Actions:
0: New IPSEC Action
1: secure11NetTo12Net
2: secure11To13
```

7. Add a new IPSec action to drop all traffic (filter action).

```
Enter the Number of the IPSEC Action [1]? 0
Enter a Name (1-29 characters) for this IPsec Action []? dropTraffic
List of IPsec Security Action types:
     1)  Block (block connection)
     2)  Permit

Select the Security Action type (1-2) [2]? 1

Here is the IPSec Action you specified...

IPSECAction Name = dropTraffic
Action   = Drop
Is this correct? [Yes]:
IPSEC Actions:
0: New IPSEC Action
1: secure11NetTo12Net
2: secure11To13
3: dropTraffic

Enter the Number of the IPSEC Action [1]? 3
Do you wish to Map a DiffServ Action to this Policy? [No]:
Policy Enabled/Disabled (1. Enabled, 2. Disabled) [1]?

Here is the Policy you specified...

Policy Name     = dropAllPublicTraffic
State:Priority =Enabled    : 5
Profile        =allPublicTraffic
Valid Period   =allTheTime
```

```
IPSEC Action   =dropTraffic
Is this correct? [Yes]:
```

Configuring and Enabling the LDAP Policy Search Engine

This example shows how to configure and enable the LDAP policy search engine. In this example there are two LDAP directories (a primary and a secondary) with IP addresses of 11.0.0.2 and 13.0.0.1 respectively. They are both listening on TCP port 389 and the device must bind to the LDAP server as cn=router, password myPassWord. The base entry in the directory tree for the router's policies is cn=RouterDeviceProfile,o=ibm,c=us.

Note: Currently both the primary and secondary LDAP servers must be listening on the same port and have the same authentication credentials for the router. The DeviceProfile must be the same for the router in both directory servers.

This example also shows how to set the default policy so that the LDAP communications are secured through IPSec. This example uses pre-shared key for the ISAKMP authentication, and SHA and 3DES for the authentication and encryption parameters for Phase 1 and Phase 2. The tunnel startpoint is 1.1.1.4 for the device performing the LDAP policy search, and the tunnel endpoints are 1.1.1.1 for the 11.0.0.1 LDAP server, and 1.1.1.3 for the 13.0.0.1 LDAP server.

1. Configure and enable the LDAP policy search engine, and list the results.

```
Policy config>set ldap primary-server 11.0.0.1
Policy config>set ldap secondary-server 13.0.0.1
Policy config>set ldap port 389
Policy config>set ldap bind-name cn=router
Policy config>set ldap bind-pw myPassWord
Policy config>set ldap anonymous-bind no
Policy config>set ldap policy-base cn=RouterDeviceProfile,o=ibm,c=us
Policy config>enable ldap policy-search
Policy config>list ldap
LDAP CONFIGURATION information:

Primary Server Address:              11.0.0.1
Secondary Server Address:            13.0.0.1


Search timeout value:                3 sec(s)
Retry interval on search failures:   1 min(s)
Server TCP port number:              389
Server Version number:               2


Bind Information:
Bind Anonymously:                    No
  Device Distinguished Name:         cn=router
  Device Password:                   myPassWord


Base DN for this device's policies:  cn=RouterDeviceProfile,o=ibm,c=us


Search policies from LDAP Directory: Enabled
```

2. Set the default policy

```
Policy config>set default-policy
List of default policy rules:
  1)  Accept and Forward all IP Traffic
```

```
  2)  Permit LDAP traffic, drop all other IP Traffic
  3)  Permit and Secure LDAP traffic, drop all other IP Traffic

Select the default policy rule to use during policy refresh periods [1]? 3
List of default error handling procedures:
  1)  Reset Policy Database to Default Rule
  2)  Flush any rules read from LDAP, load local rules

Select the error handling behavior for when loading Policy Database  [1]?

Please enter the set of Security Information for encrypting and
authenticating the LDAP traffic generated by the device when
retrieving policy information from the LDAP Server

Enter phase 1 ISAKMP negotiation parameters:

List of Diffie Hellman Groups:
     1)  Diffie Hellman Group 1
     2)  Diffie Hellman Group 2

Select the Diffie Hellman Group ID from this proposal (1-2) [1]?

List of Hashing Algorithms:
     1)  MD5
     2)  SHA

Select the hashing algorithm(1-2) [1]? 2

List of Cipher Algorithms:
     1)  DES
     2)  3DES

Select the Cipher Algorithm (1-2) [1]? 2
Authentication: (1)Pre-shared Key or (2)Certificate(RSA Sig) [2]? 1
Enter the Pre-Shared Key []? test

Enter phase 2 IPSEC negotiation parameters:
List of IPsec Authentication Algorithms:
     0)  None
     1)  HMAC-MD5
     2)  HMAC_SHA

Select the ESP Authentication Algorithm (0-2) [1]? 2
List of ESP Cipher Algorithms:
     1)  ESP DES
     2)  ESP 3DES
     3)  ESP CDMF
     4)  ESP NULL

Select the ESP Cipher Algorithm (1-4) [1]? 2
Tunnel Start IPV4 Address (Primary LDAP Server)
 [0.0.0.0]? 1.1.1.4
Tunnel End Point IPV4 Address (Primary LDAP Server)
 [0.0.0.0]? 1.1.1.1
Tunnel Start IPV4 Address (Secondary LDAP Server)
 [1.1.1.4]?
Tunnel End Point IPV4 Address (Secondary LDAP Server)
 [1.1.1.1]? 1.1.1.3
Policy config>list default-policy

Default Policy Rule:                  Drop All IP Traffic except secure LDAP
Default error handling procedure:     Reset Policy Database to Default Rule

Phase 1 ISAKMP negotiation parameters:
Diffie Hellman Group ID:              1
Hashing Algorithm:                    SHA
ISAKMP Cipher Algorithm:              ESP 3DES CBC
```

```
Per-shared key value:                  test

Phase 2 IPSEC negotiation parameters:
IPsec ESP Authentication Algorithm:    HMAC SHA
ESP Cipher Algorithm:                  3DES
Local Tunnel Addr (Primary Server):    1.1.1.4
Remote Tunnel Addr (Primary Server):   1.1.1.1
Local Tunnel Addr (Secondary Server):  1.1.1.4
Remote Tunnel Addr (Secondary Server): 1.1.1.3
```

At this point you are ready to manage the routers in your network with the policy feature. For detailed information about the commands used to configure the required policy parameters such as profiles, proposals, transforms, and actions, see “Policy Configuration Commands” on page 233, “LDAP Policy Server Configuration Commands” on page 246, and “Policy Monitoring Commands” on page 251.

Chapter 17. Configuring and Monitoring the Policy Feature

This chapter describes the LDAP and policy commands provided by the policy feature for configuring and operating the router devices in a network. It includes the following sections:

Accessing the Policy Configuration Prompt

To enter policy configuration commands:

1. Enter **talk 6** at the OPCON (*) prompt.
2. Enter **feature policy** at the Config> prompt.

The Policy config> prompt displays. You may now enter policy configuration commands.

Policy Configuration Commands

These commands enable you to configure the information contained in policies. Table 39 summarizes the policy configuration commands and the rest of this section describes them in detail. Enter these commands at the Policy config> prompt. You can either enter the command and options on one line, or enter only the command and respond to the prompts. To see a list of valid command options, enter the command with a question mark instead of options.

Table 39. Policy Configuration Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page xxix.
Add	Adds the information used to create a policy.
Change	Changes the information making up a policy.
Copy	Copies information from one policy into another.
Delete	Deletes information from a policy.
Disable	Disables a policy.
Enable	Enables a policy.
List	Displays the information in a policy.
Set	Specifies a policy to be used as the default.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page xxix.

Add

Use the **add** command to add information to a policy.

Syntax: add diffserv-action

interface-pair
ipsec-action
ipsec-manual-tunn
ipsec-proposal
ipsec-transform
isakmp-action
isakmp-proposal
policy
profile
rsvp-action
user
validity-period

Diffserv-action

Prompts you for information about which DiffServ-action selections apply.

Name The unique name of the DiffServ action for the policy.

permission-level

Specifies whether the router is to forward packets that match this DiffServ action.

1 Permit

2 Deny

Default value: 2

Queue-priority

The queue into which outgoing packets matching this DiffServ action are placed.

1 Premium (expedited forwarding)

2 Assured/Best Effort

Default value: 2

bwshare-type

The type of bandwidth share allocation.

1 Absolute (in Kbps)

2 Percentage (of total output bandwidth)

Default value: 2

bwshare

The bandwidth (in Kbps or as a percentage of output bandwidth) allocated to this service.

ds-bytemask

The mask to apply to transmitted ds bytes. This value designates which bits of a packet's TOS byte must be changed when the packet is transmitted. A zero in any bit position of this byte implies that the bit must not change.

Default value:

00

(do not change any bits)

ds-bytemodify

The marking of the IP TOS byte that should be applied to packets be forwarded by this device. Zeros in the mask imply that the corresponding bit will not change. A one implies that the bit will be marked with the bit value in the mark byte. The operation is: newTOSByte = (Mask^ & receivedTOSByte) | (Mask&Mark) The ^ ^ ∧ is a bit-based complement (Mask:Mark)

Example:

```
11111101:00000001
```

Using this example, a received value 0x07 would be sent with a value of 0x03

Default value: X'00' (do not change any bit)

interface-pair

The interface pair associates a profile with a specific interface or set of interfaces. By default the profile object does not restrict the policy from being applied to any one interface. If that is necessary, you may add interface pairs to accomplish it. The interface pair specifies the IP address of the interface on which the traffic is to arrive and the IP address of the interface on which the traffic is to leave.

The following example shows two interface pairs with the same name, representing traffic coming in on any interface and going out on the public interface, and conversely.

```
1) Group Name: inOutPublic
        In:Out=255.255.255.255 : 1.1.1.1
        In:Out=1.1.1.1 : 255.255.255.255
```

Name The name of the interface pair.

Ingress interface

IPv4 address of the input interface.

Default value: 255.255.255.255 (any)

Egress interface

IPv4 address of the output interface.

Default value: 255.255.255.255 (any)

IPSec-action

Prompts you for information for setting up the Phase 2 tunnel.

Name The name of the IPSec action.

Action type

The action to apply to packets matching the profile of a policy containing this action.

1 Block (block connection).

2 Permit (Permit packets matching this action.) If an IPSec

proposal does not exist, pass the packet; if an IPSec proposal exists, apply IPSec security processing to the packet.

Default value: 2

The following option is only available if you specify pass as the action type:

Traffic flow type
Type of traffic flow (secure tunnel or in the clear).

1 Clear

2 Secure Tunnel

Default value: 2

The following option is only available if you specify the traffic flow as secure:

Tunnel start point
IPv4 address of the tunnel start point.

Tunnel end point
IPv4 address of the tunnel end point. (0.0.0.0 for remote access)

Default value: 0.0.0.0

Tunnel-in-tunnel
Specifies whether the traffic being protected by this tunnel is to be further protected by another policy configured on this device.

Valid options: Yes or No

Default value: No

Percentage of SA lifesize/lifetime to accept
The minimum SA lifesize/lifetime (as a percentage) of the SA lifesize/lifetime. An SA lifesize/lifetime received with a value less than this is not accepted.

Default value: 75

SA refresh threshold
The percentage into the SA lifetime or lifesize value that the SA is to be refreshed automatically.

Default value: 85

DF-Bit-Setting
Specifies whether to copy the Don't Fragment bit from the original packet, and whether to set or clear it in the outer header of the IPSec packet if running in tunnel mode.

1 Copy

2 Set

3 Clear

Default value: 1

Replay-Prevention
Specifies whether IPSec is to enforce replay prevention for received

IPSec packets. In this mode IPSec ensures that the sequence numbers are valid and not received more than once.

1 Enable

2 Disable

Default value: 2

Negotiate SA Automatically
Specifies whether the Phase 2 SA is negotiated automatically at system initialization.

Yes or No

Default value: No

IPSec proposal
The name of the IPSec proposal (you may specify up to five proposals) to be sent or checked during Phase 2. The order in which you specify them determines their priority, with the first one being the highest.

IPSec-manual-tunn
Prompts you for information for manually setting up the Phase 2 tunnel.

Tunnel name
The name of the IPSec manual tunnel.

Tunnel lifetime
The tunnel lifetime (in minutes).

Default value: 46080

Encapsulation mode
The encapsulation mode to use.

tunn Tunnel mode

trans Transport mode

Default value: tunn

Policy The type of tunnel policy to use.

AH Authentication Header

ESP Encapsulating Security Payload

AH-ESP
For outbound packets, specifies that encryption runs before authentication.

ESP-AH
For outbound packets, specifies that authentication runs before encryption.

Default value: AH-ESP

Local IP address
The source IPv4 address.

Default value: 11.0.0.5

Local encryption SPI
The source security parameters index value.

Default value: 256

Local encryption algorithm
The source encryption algorithm.

Null No encryption.

CDMF Commercial Data Masking Facility.

DES-CBC
Data Encryption Standard and Cipher Block Chaining.

3DES Triple Data Encryption Standard.

Default value: DES-CBC

Local encryption key
A 16-character key.

Padding
Additional padding for local encryption.

Default value: 0

Local ESP authentication
Specifies whether local ESP authentication is to be used.

Yes or No

Default value: Yes

Remote IP address
The destination IPv4 address.

Default value: 0.0.0.0

Remote encryption SPI
The destination security parameters index value.

Default value: 256

Remote encryption algorithm
The destination encryption algorithm.

Null No encryption.

CDMF Commercial Data Masking Facility.

DES-CBC
Data Encryption Standard and Cipher Block Chaining.

3DES Triple Data Encryption Standard.

Default value: DES-CBC

Remote encryption key
A 16-character key.

Verify remote encryption padding.
Specifies whether to verify remote encryption padding.

Yes or No

Default value: No

Remote ESP authentication
Specifies whether remote ESP authentication is to be used.

Yes or No

Default value: Yes

DF bit Specifies how to process the Don't Fragment bit.

Copy Copies the DF bit.

Set Sets the DF bit on.

Clear Sets the DF bit off.

Default value: COPY

Enable tunnel
Specifies whether to enable the tunnel when it is created.

Yes or No

Default value: Yes

IPSec-proposal
Prompts you for information for creating an IPSec proposal.

IPSec proposal name
The name of the IPSec proposal.

Perfect forward secrecy
Specifies whether IKE is to be used, to prevent anyone from determining a current key from a previously compromised key.

Yes or No

Default value: No

Diffie Hellman Group ID
The type of Diffie Hellman group.

1 Diffie Hellman Group 1

2 Diffie Hellman Group 2

Default value: 1

AH transform
The name of the AH transform (you may specify up to five transforms) for this proposal. The order in which you specify them determines their priority, with the first one being the highest.

ESP transform
The name of the ESP transform (you may specify up to five proposals) for this proposal. The order in which you specify them determines their priority, with the first one being the highest.

IPSec-transform
Prompts you for information about IPSec transforms.

IPSec transform name
The name of the IPSec transform.

Protocol ID
The security protocol to use.

1 IPSec-AH

2 IPSec-ESP

Default value: 1

AH Authentication Algorithm
The AH authentication algorithm to use.

1 HMAC-MD5

2 HMAC-SHA

Default value: 1

Encapsulation mode
The encapsulation mode to use.

1 Tunnel

2 Transport

Default value: 1

ESP Authentication Algorithm
The ESP authentication algorithm to use.

0 None

1 HMAC-MD5

2 HMAC-SHA

Default value: 2

ESP cipher algorithm
The ESP cipher algorithm to use.

1 ESP DES

2 ESP 3DES

3 ESP CDMF

4 ESP Null (no encryption)

Default value: 1

SA lifesize
The lifesize (in Kb) of the SA for this proposal.

Default value: 50000

SA lifetime
The lifetime (in seconds) of the SA for this proposal.

Default value: 3600

ISAKMP-Action
Prompts you for information about which ISAKMP action to apply.

Name The name of the ISAKMP action.

Exchange mode
The type of exchange mode for Phase 1 negotiations.

1 Main

2 Aggressive

Default value: 1

Percentage of Minimum SA lifesize/lifetime
The minimum SA lifesize/lifetime (as a percentage) of the SA lifesize/lifetime. An SA lifesize/lifetime with a value less than this is not accepted.

Default value: 75

ISAKMP connection lifesize
The lifesize (in Kb) of the Phase 1 connection. Once the Phase 1 connection expires, the next time the Phase 2 SA must refresh, Phase 1 completely renegotiates before Phase 2 can start.

Default value: 5000

ISAKMP connection lifetime
The lifetime (in seconds) of the Phase 1 connection. Once the Phase 1 connection expires, the next time Phase 2 must refresh, Phase 1 starts over completely.

Default value: 5000

Negotiate SA automatically
Specifies whether the SA is negotiated automatically at system initialization.

Yes or No

Default value: No

ISAKMP proposal
The name of the ISAKMP proposal (you may specify up to five proposals) to be sent or checked during Phase 2 quick mode. The order in which you specify them determines their priority, with the first one being the highest.

ISAKMP-Proposal
Prompts you for the ISAKMP proposal information used in the ISAKMP negotiations.

ISAKMP proposal name
The name of the ISAKMP proposal.

Authentication method
The type of authentication to use during ISAKMP Phase 1 negotiations.

1 Pre-Shared Key

2 RSA SIG (certificate mode)

Default value: 1

Hash algorithm
The type of hash algorithm to use during Phase 1 negotiations.

1 MD5

2 SHA

Default value: 1

Cipher algorithm
The type of cipher algorithm to use during Phase 1 negotiations.

1 DES

2 3DES

Default value: 1

Diffie Hellman Group ID
The type of Diffie Hellman group to use during Phase 1 negotiations.

1 Diffie Hellman Group 1

2 Diffie Hellman Group 2

Default value: 1

SA lifesize
The lifesize (in Kb) of the SA for this proposal.

Default value: 50000

SA lifetime
The lifetime (in seconds) of the SA for this proposal.

Default value: 5000

Policy Prompts you for information about the policy configuration: Profile name (required), RSVP name (optional), DiffServ name (optional), IPSec name (optional), ISAKMP name (optional), and Validity Period Profile (optional). You must specify either DiffServ, IPSec, ISAKMP, or RSVP for the policy to be valid.

Default value: Valid all the time

Name The name of the policy configuration

Priority
Relative priority of this policy to other policies (the higher the number, the higher the priority). This is used to resolve conflicts if multiple policies apply to a packet.

Default value: 5

Profile
The name of a previously configured data traffic profile to use for this policy.

Validity period
The name of a previously configured validity period to use for this policy.

IPSec action
If this policy will enforce an IPSec action, the name of a previously configured IPSec action to use for this policy. If you specify a secure IPSec action, you must also specify an ISAKMP action.

ISAKMP action
The name of a previously configured ISAKMP action to use for this policy. If you specify an ISAKMP action, you must also specify an IPSec action.

Diffserv action
If you want to map a DiffServ action to this policy, the name of a previously configured DiffServ action.

RSVP action
The name of an RSVP action for this policy to enforce.

Profile

Prompts you for information for defining a set of selectors (conditionals) for a policy profile on which to perform actions.

name The name of the policy profile.

ipv4-src-address-format
The format of the IPv4 source address (range, netmask, single address).

ipv4-src-address
The IPv4 source address (low address if address format is *range*).

Default value: 0.0.0.0

ipv4-src-mask
The IPv4 source mask (high address if address format is *range*).

Default value: 255.0.0.0

ipv4-dest-address-format
The format of the IPv4 destination address (range, netmask, single address).

ipv4-dest-address
The IPv4 destination address (low address if address format is *range*).

Default value: 0.0.0.0

ipv4-dest-mask
The IPv4 destination mask (high address if address format is *range*).

Default value: 255.0.0.0

protocol-id
The protocol id on which to filter.

1 TCP

2 UDP

3 All protocols

4 Specify range

Default value: 3

src-port-start
The first port number of the source port number range.

Default value: 0

src-port-end
The last port number of the source port number range.

Default value: 65535

dest-port-start
The first port number of the destination port number range.

Default value: 0

dest-port-end
The last port number of the destination port number range.

Default value: 65535

src-id-type
The source ID type, which is sent to the remote. This value is used to determine which policy contains the ISAKMP information needed during ISAKMP Phase 1 negotiations. It is compared to the information in the identification payload of the ISAKMP packet. This information is needed if the remote peer must identify the device with a value other than IP address.

1 Local tunnel end point

2 Host fully qualified domain name

3 User fully qualified domain name

4 Key ID

any-user-access
Allow access for any user within the profile definition. If you specify No, then you are prompted for the name of the remote user group for this profile. This attribute is only required if you want to limit the access of remote access peers to a specific policy.

Yes or No

Default value: Yes

Received DS byte mask
The 8-bit mask to apply to an incoming packet's TOS byte.

Default value: 0

Received DS byte match
The 8-bit pattern to compare to the result of ANDing the incoming TOS byte with the Received DS byte mask value.

Default value: 0

Interface pairs
If this policy must restrict the traffic flows to specific interfaces, this is the name of the interface pair group.

RSVP-Action
Prompts you for information about which RSVP actions apply.

Name The name of the RSVP action.

Permission
Specifies the permission level for RSVP sessions that match this action.

1 Permit

2 Deny

Default value: 2

Max token rate
The maximum amount of bandwidth (in Kbps) that RSVP is to allocate for an individual flow.

Default value: 100

Max duration
The maximum amount of time (in seconds) that a flow can last (0 implies forever).

Default value: 600

RSVP-to-DS
Specifies whether to map RSVP flows that match this action to a configured DiffServ action. RSVP uses the information from the DiffServ action to mark the TOS byte for the next DiffServ-enabled upstream device. This is for use in a network in which packets leave an RSVP-enabled network into a DiffServ-enabled network.

Yes or No

Default value: No

VALIDITY-PERIOD
Prompts you for information about the period during which the policy is valid, and creates a policy profile.

Name The name of the validity period profile.

yyyymmddhhmmss:yyyymmddhhmmss
The period during which the policies containing this validity period profile are valid.

Example:

```
19980101000000:19981231000000
```

Months
The months during which the policies containing this validity period profile are valid. You can specify any sequence of months, using the first three letters of each month (for example, jan or dec), with the months separated by a spaces, or you can specify `all` to signify every month of the year.

Days The dates on which the policies containing this validity period profile are valid. You can specify any sequence of dates, using the first three letters of each day (for example, mon or fri), with the days separated by a spaces, or you can enter `all` to specify every day of the week.

Starting time
The time at which policies containing this validity period profile are valid. Specify this in the form hh:mm:ss or specify * if you want the policy to be valid all day.

Default value: *

Ending time
The time at which the validity of policies containing this validity period profile expires. Specify this in the form hh:mm:ss.

Default value: None

Change

Use the **change** command to change information in a policy object. See the description of the **add** command for the available objects.

Copy

Use the **copy** command to copy information from one policy object to another. See the description of the **add** command for the available objects. (The interface-pair, manual tunnel, and user options do not apply to the copy command.)

Delete

Use the **delete** command to delete information from a policy object. See the description of the **add** command for the available objects.

Disable

Use the **disable** command to disable a policy configuration.

Syntax: disable policy

Policy Prompts you for the name of the policy configuration to disable.

Enable

Use the **enable** command to enable a policy configuration.

Syntax: enable policy

Policy Prompts you for the name of the policy configuration to enable.

List

Use the **list** command to display any or all of the policy configuration information.

Syntax: list all
default-policy
ldap
refresh

All Displays all policy configuration information.

Default-policy
Displays the name of the default policy.

LDAP Displays the names of the defined LDAP configurations.

Refresh
Lists the policy refresh status (Enable or Disable) and the refresh interval time.

LDAP Policy Server Configuration Commands

The LDAP policy server configuration commands enable you to specify LDAP server options for retrieving policy information. Table 40 on page 247 summarizes the LDAP configuration commands, and the rest of this section describes them in detail. Enter them at the `Policy config>` prompt. You can either enter the command and options on one line, or enter only the command and respond to the prompts. To see a list of valid command options, enter the command with a question mark instead of options.

Table 40. LDAP Configuration Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Disable ldap	Disables LDAP configuration options.
Enable ldap	Enables LDAP configuration options.
Set ldap	Specifies LDAP configuration options.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Disable LDAP

Use the **disable ldap** command to disable LDAP policy search functions in the directory.

Syntax: disable ldap policy-search

policy-search
Disables LDAP from performing policy search functions in the directory.

Enable LDAP

Use the **enable ldap** command to enable LDAP policy search functions in the directory.

Syntax: enable ldap policy-search

policy-search
Enables LDAP for performing policy search functions in the directory.

Set Default-Policy

Use the **set default-policy** command to specify the policy options to use while the policy database is being refreshed. The command sets the error handling options and the default security needed for accessing the LDAP policy server.

Syntax: set default-policy
default-error-handling
default-security

default-error-handling
Specifies the error handling options to use while the policy database is being refreshed.

Note: The default error handling setting determines the behavior of the device if an error occurs while rebuilding the policy database. If an error occurs then you have the options for how the device is to behave. They are:

1. Reset policy database to default security.
2. Flush any rules read from LDAP, load local rules plus default security.

These settings are only valid if there was an error building the policy database. Either option inherits the default security of drop or pass when an error occurs. If you select option 2 then all traffic is dropped

or passed unless it matches a locally defined policy. If the policy database builds successfully then this option is not used.

default-security
Specifies the security options to use while the policy database is being refreshed.

Note: Once the policy database has been built successfully, the default behavior is defined as pass. This means that if a packet does not match any policy rule then it will be passed in the clear. If you want packets that do not match a rule to be dropped globally or just for certain interfaces, then you must define a policy to do that.

1 Accept and forward all IP traffic.

2 Permit LDAP traffic, drop all other IP traffic.

If you select this option, then you are prompted for the local IP addresses on the device on which the LDAP traffic is to be sent and received.

3 Permit and secure LDAP traffic, drop all other IP traffic.

If you select this option, then you are prompted for the following information:

DHGroupId
The Diffie-Hellman Group Id to use during the ISAKMP Phase 1 negotiations.

1 DH Group 1.

2 DH Group 2.

Phase1-Hash-Algorithm
The hash algorithm to use during the Phase 1 negotiations. The hash algorithm provides the authentication of the Phase 1 messages.

1 MD5.

2 SHA.

Phase1-Cipher-Algorithm
The cipher algorithm to use during Phase 1 negotiations. The cipher algorithm provides encryption protection for the Phase 1 negotiations.

1 DES

2 3DES

Phase1-Authentication-Method
The authentication method to use with the remote peer. This specifies how ISAKMP determines whether the remote peer is actually the correct device with which to be negotiating.

1 Pre-shared key

2 Certificate (RSA SIG)

Pre-Shared-Key-Value
If you have specified the pre-shared key Phase 1 authentication method, then you are prompted to enter the key value in ASCII.

Phase2-ESP-Authentication-Algorithm
ESP is the only IPSec protocol allowed for the default security. You are prompted for the authentication algorithm to use during Phase 2 ISAKMP negotiations.

0 None

1 HMAC-MD5

2 HMAC-SHA

Phase2-ESP-Cipher-Algorithm
ESP is the only IPSec protocol allowed for the default security. You are prompted for the encryption algorithm to use during Phase 2 ISAKMP negotiations.

1 ESP DES

2 ESP 3DES

3 ESP CDMF

4 ESP NULL

Primary-Tunnel-Start
The IP address on the device that is to be used for the IKE and IPSec traffic between the device and the security gateway protecting the primary LDAP server.

Primary-Tunnel-End
The IP address on the remote security gateway protecting the primary LDAP server that are to be used for the IKE and IPSec traffic.

Secondary-Tunnel-Start
The IP address on the device that is to be used for the IKE and IPSec traffic between the device and the security gateway protecting the secondary LDAP server.

Secondary-Tunnel-End
The IP address on the remote security gateway protecting the secondary LDAP server that are to be used for the IKE and IPSec traffic.

Set LDAP

Use the **set ldap** command to configure the LDAP operating parameters.

Syntax: set ldap
anonymous-bind
yes
no
bind-name <*name*>
bind-pw <*pw*>
policy-base <*string*>
primary <*ip-address*>

secondary <*ip-address*>

version <*value*>

anonymous-bind [Yes or No]
Specifies whether you want to bind to the LDAP directory anonymously or with the bind name and bind password you have specified.

Default value: Yes

bind-name <*name*>
Prompts you for information needed to bind to the LDAP server before a search of its directory can be performed. The *name* parameter specifies the distinguished name that the router uses to identify itself. If you do not enter this parameter, then the bind is issued as an anonymous request.

bind-pw <*pw*>
Prompts you for information needed to bind to the LDAP server before a search of its directory can be performed. The *pw* parameter is the password related to the distinguished name. If you do not enter this parameter, then the bind is issued as an anonymous request.

policy-base <*string*>
Prompts you to enter a character string that is used to define the scope of the search for policies in the router's SRAM and the LDAP server. For example, you can use this option to return policies that only apply to router A, or for NHD, or for IBM-US. The policy-base is the distinguished name of the DeviceProfile object in the LDAP server.

primary <*ip-address*>
Prompts you for the IPv4 address of the LDAP server from which to retrieve policies.

secondary <*ip-address*>
Prompts you for the IPv4 address of a backup LDAP server that is used if the default server cannot be reached.

version <*value*>
Prompts you for the LDAP version number supported by the LDAP server.

Default value: 2 (The only acceptable values are 2 or 3.)

Set Refresh

Use the **set refresh** command to enable or disable automatic refresh of the policy database once each day. If enabled then the policy database automatically refreshes once a day at the specified time. This enables all policy-enabled routers in the network to incorporate automatically any policy changes that have occurred in the LDAP directory. To reset this parameter, use the policy feature's Talk 5 **reset refresh** command.

Syntax: set refresh

enabled

yes

no

<*time*>

enabled [yes or no]
Specifies whether to perform the automatic refresh.

<*time*>

If you specify enabled yes, designates the time of day (in 24-hour format) at which the refresh is to occur.

Accessing the Policy Monitoring Prompt

The policy console portion of the policy feature enables you to view policies that are in the policy database and to enable or disable individual policies. To access the Policy monitoring environment type **talk 5** at the OPCON prompt (*):

```
* t 5
```

Then, enter the following command at the **+** prompt:

```
+ feature policy
Policy>
```

Policy Monitoring Commands

These commands enable you to view the profiles defined in the policy database and to enable or disable individual policies. Table 41 summarizes the policy monitoring commands and the rest of this section describes them. Enter the commands at the `Policy console>` prompt. You can either enter the command and options on one line, or enter only the command and respond to the prompts. To see a list of valid command options, enter the command with a question mark instead of options.

Table 41. Policy Monitoring Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Disable	Disables a policy that is loaded in the policy database.
Enable	Enables a policy that is loaded in the policy database.
Reset	Refreshes or resets policy-related criteria.
Search	Tests or debugs activity between the LDAP client and server.
Status	Displays information about the policy database.
List	Displays information about the LDAP configuration and the policies defined.
Test	Queries the policy engine and retrieves the rules that were selected
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Disable

Use the **disable** command to disable a policy that is currently loaded in the policy database. Any data packet that matches the criteria of a policy you disable will have default decisions applied to it.

Syntax: disable <*policy-name*>

Enable

Use the **enable** command to enable a policy that is currently loaded in the policy database. Any data packet that matches the criteria of a policy you enable will have the decisions configured for the policy applied to it.

Syntax: enable <*policy-name*>

Reset

Use the **reset** command to refresh or reset policy-related criteria.

Syntax: reset ldap-config
policy-database
refresh-time

ldap-config
Dynamically loads the LDAP configuration (as specified in the **set ldap** command) into memory. Any changes become active for the next search operation. This command also forces a reset of the policy database and inactivates the policy database refresh time.

policy-database
Refreshes the policy database. Stops all tunnels, Phase 1 and Phase 2 SAs, resets RSVP and DiffServ data structures, and flushes the policy database. Then policies are loaded from the LDAP server and an autostart is done. While the database is being rebuilt, no packets will be allowed in to or out of the router except for packets to and from the LDAP server.

refresh-time
Sets the time at which the policy database will be refreshed automatically on a daily basis. If you have disabled the refresh time, then the database will not be refreshed until the router is rebooted or restarted.

Search

Use the **search** command to test or debug activity between the LDAP client and server. You can perform searches against the directory and have the results of the searches displayed in talk 5.

Syntax: search *filter*
ipaddress

filter Specifies a filter value for the search operation.

ipaddress
Specifies the IP address of the server.

Status

Use the **status** command to display information about the policy database.

Syntax: status

status Displays the results of the most recent policy database refresh, the time that has elapsed since the refresh, and the time that the next refresh is scheduled.

Example:

```
Policy>status
Status of Last Search:       Failed
Time since last refresh:     4 seconds
Next Policy Refresh not scheduled
```

List

Use the **list** command to display information about LDAP configurations and policies.

Syntax: list default-policy
ldap
policy
refresh
rule
stats

default-policy
Lists the default policy used during policy database refreshes.

ldap Lists the LDAP configurations in SRAM.

policy

basic Lists policy components by logical policy name. You may select one policy or list all policies. The listing displays the names of the components of policies as they were entered in during configuration in Talk 6.

complete
Does the same as list policy basic, except that the listing displays a complete listing of all parameter values for each logical policy.

generated
Does the same as list policy basic, except that the listing displays the names of all the generated rules for each logical policy.

refresh
Lists the policy refresh status (Enable or Disable) and the refresh interval time.

rule Lists information about generated rules according to the following options:

basic Lists all the generated rules. You can select a rule from the list or list all rules. The listing displays the names of the components of the rules. The components are:

policy name

loaded from (LDAP or local)

state

priority

number of hits

profile

validity (followed by an action list consisting of the following)
IPSec (and, or)

ISAKMP (and, or)

DiffServ (and, or)

RSVP

complete
Does the same as rule basic, except that the listing displays the names of all the parameters for each component.

stats Lists the rules that have been hit and the number of hits. A rule can have multiple actions and not all actions are hit, so this options also indicates which action of the rule was hit, and the number of times.

Test

Use the **test** command to verify the behavior of the policy database. It allows you to enter a selector set, which queries the policy engine and retrieves the rules that match. You are prompted for the source and destination addresses, source and destination ports, the protocol ID, and the TOS value. If a rule is matched, then the command returns the name of the rule. Otherwise it indicates *No match found*.

Syntax: test forwarder

ISAKMP

IPSec

RSVP

forwarder
Simulates a database query from the IP forwarding engine and returns any policy decisions that would result from such a query. The type of policy returned could include DiffServ information, IKE Phase 1 and Phase 1 information, and IPSec manual tunnel IDs.

ISAKMP
Simulates a database query from IKE for Phase 1 policy information and returns any policy decisions that would result from such a query. If you use this option, you must set the source and destination addresses to the tunnel endpoint IP addresses, the protocol to 17, and the source and destination ports to 500.

IPSec Simulates a database query from IKE for Phase 2 policy information and returns any policy decisions that would result from such a query. If you use this option, you must set the source and destination addresses to the tunnel endpoint IP addresses, the protocol to 17, and the source and destination ports to 500.

RSVP Simulates a database query from RSVP and returns any RSVP policy decisions that would result from such a query.

Chapter 18. Using IP Security

This chapter explains how to use the IP Security feature and contains the following sections:

IP Security Overview

This section provides an overview of IP security capabilities for both IPv4 and IPv6.

Using Secure Tunnels

To protect IP packets sent to another host, router, or firewall, you may configure a secure tunnel for each IP route that must be secure. An IPSec tunnel is a two-way logical connection to the remote host, router, or firewall over which a local router sends protected IP packets. A secure tunnel is identified by parameters such as the addresses of the source host and destination host, port numbers, and tunnel ID.

With IPv4 you can define a negotiated tunnel by configuring a tunnel policy in the policy database, or you can create a manual tunnel using the Talk 6 **add tunnel** command as shown at "Configuring the Tunnel for Router A" on page 287. With IPv6, use the Talk 6 **add tunnel** command.

To establish a secure IPSec tunnel, a policy may specify the IP Authentication Header (AH) function (see "IP Authentication Header" on page 258), which attaches special authentication headers, and the IP Encapsulation Security Payload (ESP) function (see "IP Encapsulating Security Payload" on page 259), which encrypts the data. The policy establishes which of the following security measures are implemented for packets:

- AH algorithm and AH authentication keys (See "Configuring the Algorithms" on page 278 or "Configuring the Algorithms" on page 288 as appropriate.)
- ESP encryption algorithm and ESP encryption and decryption keys (See "Configuring the Algorithms" on page 278 or "Configuring the Algorithms" on page 288 as appropriate.)
- Security parameters indexes (SPIs) (See "Security Associations" on page 260.)

Note: For each secure tunnel, the sender and the receiver must select identical options.

IP Security Concepts

Packets sent using the Internet Protocol (IP) can be made secure by using the IP Security feature of the 2210.

Security, as defined by RFC 2401 - Security Architecture for the Internet Protocol, consists of the following functions:

Authentication
Knowing that the data received is the same as the data that was sent and that the claimed sender is, in fact, the actual sender.

Integrity
Ensuring that data is transmitted from source to destination without undetected alteration.

Confidentiality
Communicating so that the intended recipients know what was being sent but unintended parties cannot determine what was sent.

Non-repudiation
Communicating so that the receiver can prove that the sender did, in fact, send certain data even though the sender might later deny ever having sent it.

Note: In some countries, encryption support is not provided because of U.S. export regulations, and the encryption parameters are not displayed. However, the ESP-NULL algorithm is always available. For a definition of the ESP-NULL algorithm, see “ESP Encryption Algorithms” on page 259.

IP Security Terminology

The following terms are used when describing IPSec topics related to IPv4:

Authentication Header (AH)
A data area containing packet header information, which provides data origin authentication and data integrity and replay protection.

Certificate
An ASN.1 encode data item (according to ITU X.509 standands) that binds an end entity’s ID to its public key. (In this case, the end entity is the ISAKMP negotiation entity.) The end entity must register its ID and public key with a certificate authority (CA) by submitting a certificate request. The CA verifies the request, signs it, and issues it to the entity. ISAKMP uses the public key certificate during Phase 1 processing to authenticate the initial message exchanges that set up the master secret (cryptographic key) between routers.

Certificate Authority (CA)
A trusted authority that issues “signed” X.509 digital certificates that network users must use to exchange secure user data using ISAKMP. To participate in secure data exchanges with other ISAKMP-enabled parties, a router must register with a CA and obtain an X.509 digital certificate to be used in authentication.

Digital Signature
A data item containing a user’s encoded ID, which becomes part of an X.509 digital certificate. Users exchange certificates during Phase 1

negotiations to authenticate one another. The signature is generated by performing a public key operation on an input data area to be signed.

Encapsulating Security Payload (ESP)
An IPSec function that can encapsulate and encrypt a datagram so that its contents cannot be determined by anyone except the recipient. This comprises data integrity and replay protection. ESP also provides data origin authentication. It operates in the following modes: transport mode, which encrypts only the payload of the original datagram, leaving the addressing information visible to unauthorized parties, and tunnel mode, in which the entire original datagram, including the header, are encrypted. This conceals sensitive address information.

Internet Key Exchange (IKE)
A protocol derived from the ISAKMP and Oakley protocols, which is used by the Internet community to exchange cryptographic keys and authenticate the communicating parties.

ISAKMP
Internet Security Association and Key Management Protocol. This function automatically sets up security associations and manages packets' cryptographic keys for the duration of a data exchange.

Management Information Base (MIB)
A data block sent by a router in response to an inquiry from a central, trusted authority that has requested statistical information about router operations. The authority can detect problems in the network and contact a responsible party to take corrective action.

Oakley
The cryptographic key management protocol used by ISAKMP.

Perfect Forward Secrecy (PFS)
The level of data security obtained if Phase 2 negotiations derive new cryptographic keying information for each negotiation. ISAKMP accomplishes this by enabling the exchange of public Diffie Hellman values between parties. This security feature prevents anyone from determining a current cryptographic key from a previously compromised key.

Phase 1 Negotiations
The communication between a sender and receiver that establishes an ISAKMP security association and cryptographic keys that will protect the ISAKMP messages to be exchanged during Phase 2 negotiations. Phase 1 is processor-intensive, and typically is done infrequently, perhaps only daily or weekly.

Phase 2 Negotiations
The exchange of ISAKMP messages between a sender and receiver during which security associations and cryptographic keys are negotiated that will protect user data exchanges. These negotiations typically happen frequently, perhaps every two to three minutes, and are used to refresh cryptographic keys on a regular basis without user intervention.

Proxy A router that is assigned to operate in behalf of another network device.

Public Key Infrastructure (PKI)
The framework that a CA uses to bind the user's ID with its public key and distributes the bound public key in a way that ensures its security.

Quick Mode
The term used to describe the Phase 2 negotiations for non ISAKMP security associations.

Replay
The act of capturing a datagram and either attempting to determine its contents or mounting a denial-of-service attack by resending it repeatedly.

Security Association (SA)
A data area tying together information about a data packet, such as its cryptographic algorithm and key information, the identities of the participating parties, and so forth.

Transform
A named collection of information about a configuration of authentication and encryption selections.

IP Authentication Header

The Authentication Header (AH) is described in RFC 2402 IP Authentication Header. This header contains authentication data for the IP datagram.

For IPv4 using negotiated IPSec, the policy assigned to a datagram implements a cryptographic authentication function that relies upon the Internet Key Exchange (IKE) protocol and a public/private key pair. For IPv4 manual tunnels and for IPv6, the sender uses a cryptographic function that relies upon a secret authentication key. In either case, the cryptographic authentication function is applied to the contents of the datagram. You may specify AH alone or with ESP. See “Using AH and ESP” on page 259 for details.

AH Authentication Algorithms

A secure tunnel that uses the AH tunnel policy must use one of the following authentication algorithms:

- HMAC-MD5 IP Authentication with Replay Prevention
- HMAC-SHA-1 IP Authentication with Replay Prevention

These AH algorithms combine a keyed message authentication function using cryptographic hashing (hashed message authentication code, abbreviated as HMAC) with an optional replay prevention function. Replay prevention uses a sequence number contained in the AH to verify that a packet has not been received previously. Replay prevention protects the receiver from denial-of-service attacks, in which the same packet is sent repeatedly and the router becomes so busy processing the duplicate packets that it cannot process legitimate traffic. An authentication code is applied to a secret cryptographic key and the data, then to the output of the secret key and the output of the first operation. See Figure 22 on page 259 for an illustration of how this is done for HMAC-MD5.

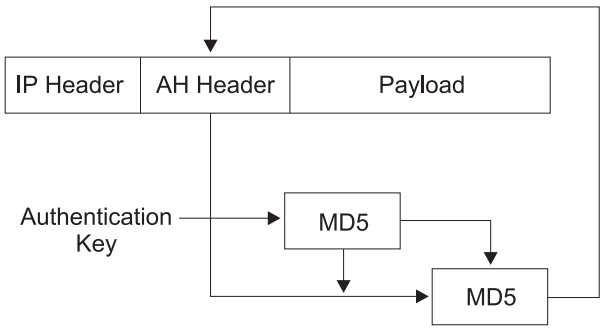


Figure 22. Creation of an HMAC MD5-Authenticated Message

IP Encapsulating Security Payload

IP Encapsulating Security Payload (ESP) is described in RFC 2406 IP Encapsulating Security Payload. ESP encrypts part or all of the IP packet to provide confidentiality in addition to authentication (optional) and integrity. However, if you select the ESP-NULL algorithm, ESP performs only authentication and integrity checking. You may specify ESP alone or with AH. See "Using AH and ESP" for details.

ESP Authentication Algorithms

The algorithms available for ESP authentication are the same as those for AH, previously shown at "AH Authentication Algorithms" on page 258.

ESP Encryption Algorithms

A secure tunnel that uses the ESP encryption policy must either use one of the following encryption algorithms or the ESP-NULL algorithm:

- Data Encryption Standard in Cipher Block Chaining Mode (DES-CBC)
- Commercial Data Masking Facility (CDMF)
- Triple DES (3DES)

Note: Except for ESP-NULL, the ESP encryption algorithms are subject to U.S. export laws. If your 2210 does not allow you to use some or all of these algorithms, sale of those algorithms may be prohibited in your country. Check with your IBM representative for more information.

The ESP-NULL algorithm does not encrypt the cleartext data and is available in all countries. It enables ESP authentication and integrity checking only, not encryption. If you use ESP-NULL, you ***must*** use one of the ESP authentication algorithms.

Using AH and ESP

A secure tunnel may use one of the following authentication/encryption selections: AH, ESP, AH-ESP, or ESP-AH. If you want a combination of AH and ESP, the following statements apply:

- The policy AH-ESP specifies that for outbound packets, encryption runs before authentication. In this case, in the destination router the AH authentication function runs first, checking inbound packets, and only packets that pass authentication are forwarded to ESP for decryption.

- The policy ESP-AH specifies that for outbound packets, authentication runs before encryption. In this case, in the destination router the ESP function first decrypts inbound packets, and only packets that are decrypted successfully are forwarded to AH authentication.

Security Associations

A Security Association (SA) is a simplex “connection” that affords security services to the traffic carried by it. Security services are afforded to an SA by the use of AH or ESP, but not both. If both AH and ESP protection are applied to a traffic stream, then two (or more) SAs are created to afford protection to the traffic stream. To secure typical bidirectional communication between two hosts or between two security gateways, two SAs (one in each direction) are required.

Tunnel Mode and Transport Mode

The operational mode (either tunnel or transport) determines how IPSec handles IP packets. Tunnel mode is the default, and is required if the router is acting as a security gateway. It protects data on a single segment of a path through a network. Transport mode is allowed only when the router is acting as a host, and protects data end-to-end, along a complete path.

AH and Operational Modes

In tunnel mode, the AH is placed in front of the IP packet and a new IP header is created and placed in front of the AH. The IP header of the packet being tunnelled (inner header) carries the ultimate source and destination addresses of the packet. The new IP header (outer header) can contain the addresses of security gateways, which are the tunnel endpoints. The AH protects the entire new packet, both the new IP header and the IP packet being tunnelled, except for the mutable fields in the new IP header.

In transport mode, the AH is inserted after the IP header and before the header of an upper-layer protocol, such as TCP or UDP. In this mode, AH authenticates the upper-layer protocol header and the contents of the IP packet, except for the mutable fields in the IP header (such as time-to-live [TTL], checksum, fragment flag, fragment offset, and type of service [TOS]).

Figure 23 on page 261 shows the format of AH-protected datagrams.

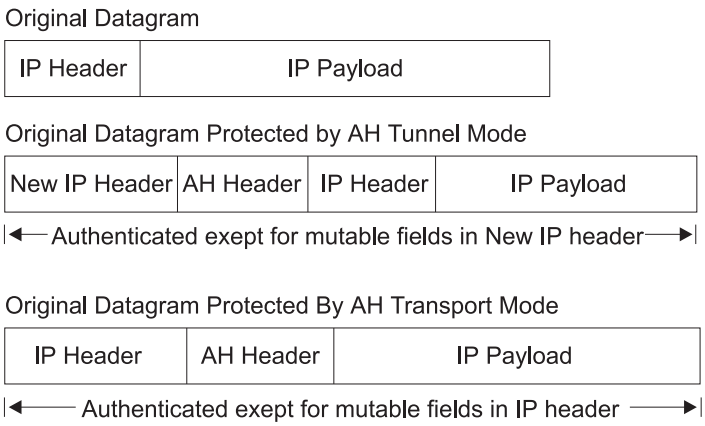


Figure 23. AH-Protected Datagram Format

ESP and Operational Modes

In tunnel mode, the payload data contains the entire IP packet, and a new IP header is created and placed in front of the ESP header. The IP header of the packet being tunnelled (inner header) contains the ultimate source and destination addresses of the packet, while the new IP header (outer header) contains the addresses of security gateways. ESP encrypts the tunnelled IP packet. If you use ESP authentication, the ESP header, the tunnelled IP packet, and the ESP trailer are authenticated.

In transport mode, the payload data contains encrypted upper-layer protocol data, such as TCP or UDP data. If you use authentication, the ESP header, the upper-layer protocol data, and the ESP trailer are authenticated.

Figure 24 shows the format of ESP-protected datagrams.

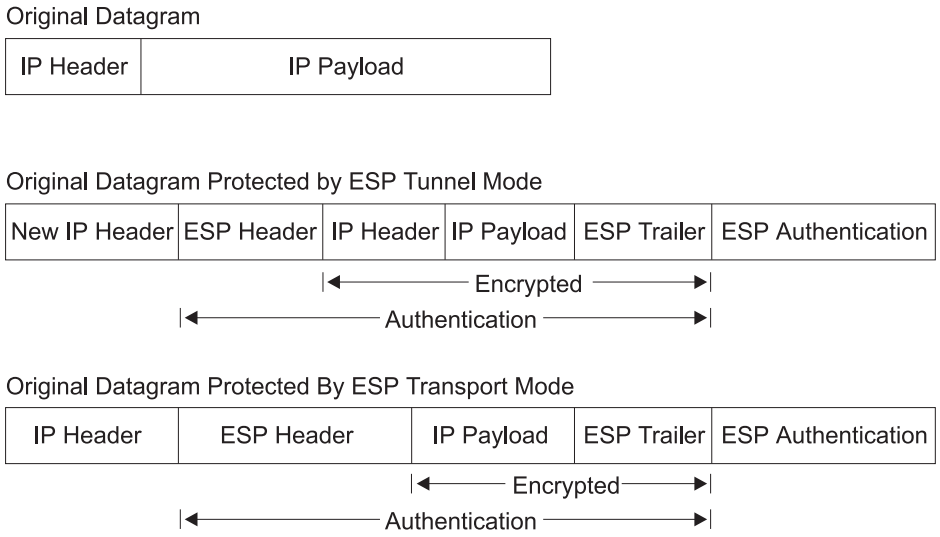


Figure 24. ESP-Protected Datagram Format

Nesting AH and ESP

You may nest one protocol within another instance of itself or the other protocol. Figure 25 shows the effects of nesting an ESP-protected datagram within an AH tunnel.

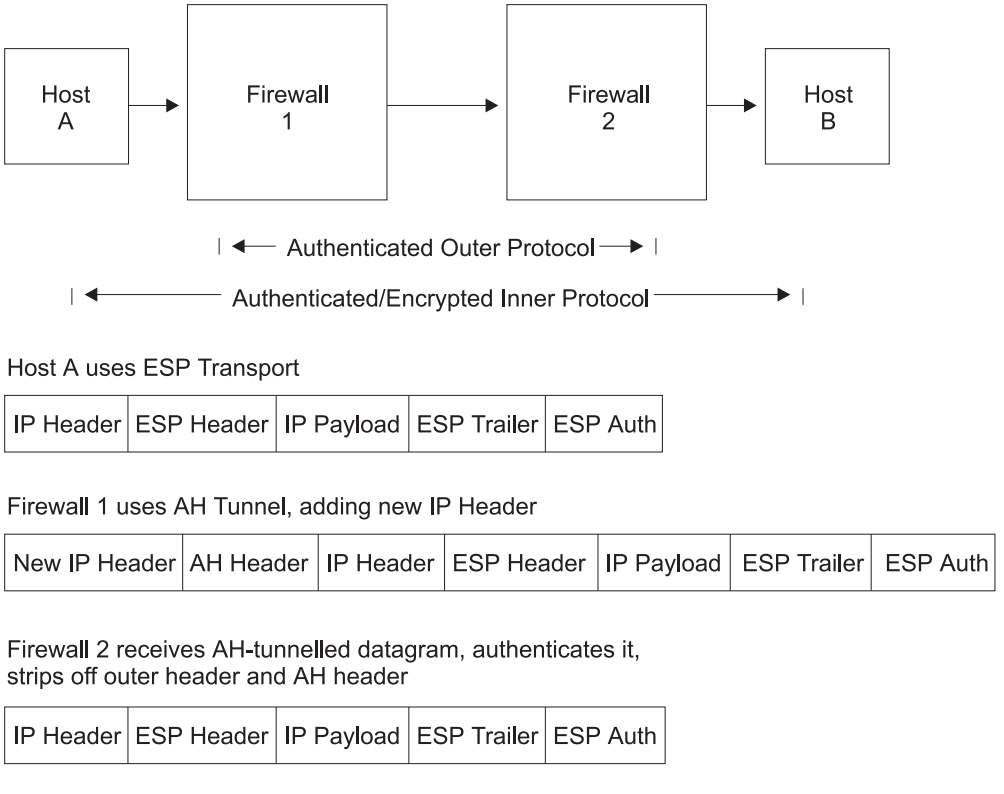


Figure 25. Nesting ESP Within an AH Tunnel

Using IP Security with L2TP Packets

With IPv4, you may also use IPSec to protect L2TP packets. After creating an L2TP tunnel by encapsulating an L2TP frame inside a UDP packet, you may encapsulate the UDP packet inside an IP packet whose source and destination addresses define the tunnel's end points. Then you can apply AH, ESP, and ISAKMP protocols to the IP packet. Figure 26 shows an IP-encapsulated L2TP packet including PPP and its payload protocol for transmission across the Internet.

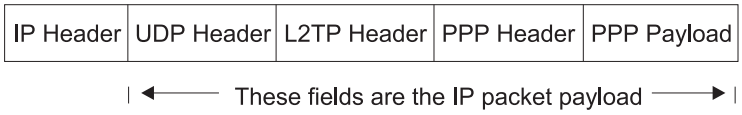


Figure 26. IPSec-Protected L2TP Packet

Tunnel-in-Tunnel Mode

For greater security, in addition to the security features already discussed, you may encapsulate the packets of a traffic stream twice and transmit them first through one IPSec tunnel and then through another (tunnel-in-tunnel).

Note: The use of multiple encryption (using tunnel-in-tunnel mode when encryption is preformed for both tunnels) within the router is restricted by U.S.A. Government export regulations. It is only supported in software loads that are under strict export control (software loads that support RC4 with 128 bit keys and Triple DES).

With IPv4, a rule in the policy database designates a packet for encapsulation (inner) for the first tunnel, and before the packet is sent, the rule causes the packet to be submitted to a second tunnel for a second encapsulation (outer). With IPv6, a packet filter access control rule identifies a packet for encapsulation (inner) for the first tunnel, and before the packet is sent, a second rule causes the packet to be submitted to a second tunnel for a second encapsulation (outer).

The two IPSec tunnels originate in the same router and the remote ends of the tunnels are at the same physical location, but on different machines. The remote end of the first tunnel can be either a secure gateway or a host; the remote end of the second tunnel *must* be a secure gateway router. Because the tunnels have different destinations, they must have different remote IP addresses. Both tunnels used for tunnel-in-tunnel must be configured for tunnel mode, and extra padding is not allowed on the second tunnel.

After it has been encapsulated twice, the packet is transmitted through the second (outer) tunnel. At the end of that tunnel, the outer encapsulation is removed and the packet is forwarded to the first tunnel (inner), based on information in the header created by the first tunnel encapsulation. At the end of this tunnel, the inner encapsulation is removed and the packet is forwarded to its final destination.

Path Maximum Transmission Unit Discovery

For both IPv4 and IPv6, IPSec supports Path Maximum Transmission Unit (PMTU) Discovery if the 2210 is acting as a security gateway. Support of PMTU Discovery is a concern if a packet cannot be fragmented. With IPv4, a packet cannot be fragmented if the Don't Fragment (DF) bit is set. With IPv6, a packet cannot be fragmented by intermediate routers. In these situations, if the packet does not fit on a link in the path from one end of the secure tunnel to the other, a "packet too big" ICMP error message is sent to the packet originator.

Because the router is acting as a security gateway, the error packet is returned to the originating router instead of the true originator of the packet. The receiving router must pass the reported MTU back to the true originator, who can reduce the packet size so that it will reach the final destination. Support for PMTU Discovery is discussed in RFC 2401 - Security Architecture for the Internet Protocol.

IPv4 provides the following options for the DF bit setting in the outer header of the tunnelled packet:

1. Copy from the inner header
2. Always set
3. Always clear

These options are available when configuring secure tunnel-in-tunnel mode, for example, using the policy feature **add ipsec-manual-tunn** (IPv4) or the Talk 6 **add tunnel** (IPv6) command. The DF bit is handled according to the option selected except under the following conditions:

- The tunnel MTU is equal to the minimum MTU.
- The incoming packet size is less than or equal to the minimum MTU.

- The encapsulated packet size would be greater than the minimum MTU.

In these circumstances, for IPv4, the DF bit is not set, regardless of the configuration, and the secure packet may be fragmented as needed on the path to the receiver. For IPv6, the packet is fragmented as needed as it leaves the security gateway so that it fits on the PMTU for the tunnel. This special action is needed because the incoming packet is already less than or equal to the minimum MTU, so the originating host will not decrease the size any further. If fragmentation were not allowed, this packet would never reach its final destination

Because changes in the network topology or configuration can change the PMTU, the PMTU value must be aged out periodically and reset to the maximum. The aging timer value defaults to 10 minutes and can be configured with the Talk 6 **set path** command. Setting the aging parameter to 0 disables PMTU aging.

Diagram of a Network with an IP Security Tunnel

Figure 27 shows an example of a network with two IPSec tunnels that connect router A (with IPSec) to router B (with both IPSec and Network Address Translation for IPv4).

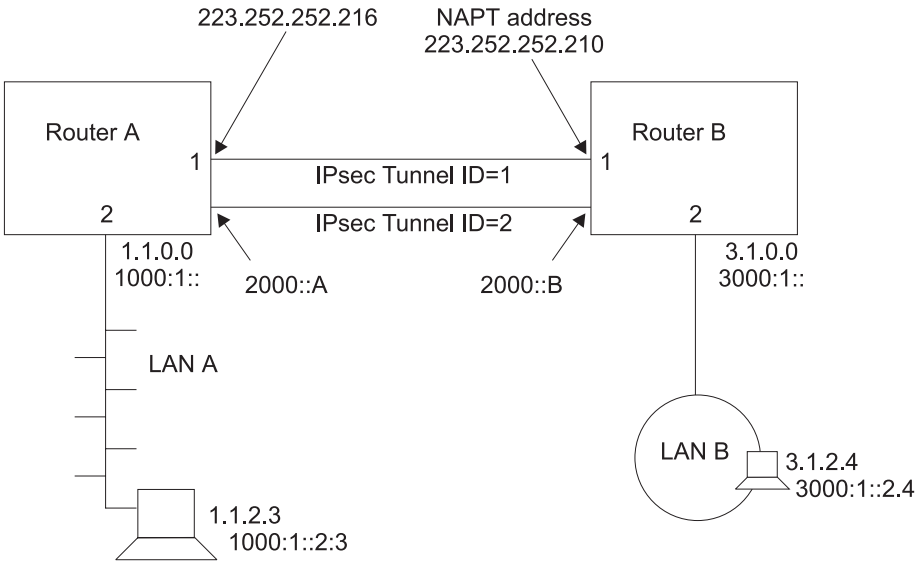


Figure 27. Network with IPSec and NAT

In this network, an IPSec tunnel with IPSec tunnel ID 1 has been configured from IPv4 address 223.252.252.216 in router A to IPv4 address 223.252.252.210 in router B. Router A is configured for IPSec. Router B is configured for both IPSec and NAT.

Also in this network, an IPSec tunnel with IPSec tunnel ID 2 has been configured from IPv6 address 2000::A in Router A to IPv6 address 2000::B in Router B.

With IPv4, to configure this network for IKE, follow the steps starting at “Configuring Internet Key Exchange (IPv4)” on page 273. For IPv4 with manual IPSec, follow the steps starting at “Configuring a Manual Tunnel (IPv4)” on page 286. For IPv6, follow the steps starting at “Configuring a Manual Tunnel (IPv6)” on page 289.

Note: Even if you do not plan to use NAT in your network, the description of configuring router B can help you understand the relationships between the parameters at each end of the IPSec tunnel more clearly.

Using the Internet Key Exchange

This section explains how you can use the Internet Key Exchange (IKE) to automate the definition and creation of IPSec security associations (SAs). IKE is a standard supported by the IETF (RFC 2409), which provides a standard way for IPSec-enabled products from the same or different vendors to communicate about their security requirements.

IKE provides a framework by means of which the following security requirements are met:

Authentication of the remote negotiating entity (IKE peer)
Through the use of either a pre-shared key or a digital certificate, IKE authenticates the identity of the entity you are communicating with by making the entity prove it is who it claims to be.

Creation of identical keying material in both peers
By using the Diffie-Hellman public key/private key mechanism, IKE provides for the exchange of the public key component and for the independent generation of identical keys by each peer.

Provide protection for the negotiation of IPSec security associations
Through a two-phase process, described in the following topic, IKE provides for the creation of security associations that are used solely to protect the negotiation of IPSec *tunnels*, and for the actual negotiation and creation of *security associations* that IPSec uses to protect user data.

Internet Key Exchange Phases

IKE defines two distinct negotiation exchanges: Phase 1 and Phase 2. Phase 1 sets up a secure tunnel between the two IKE peers, which will provide protection for the subsequent IPSec tunnel negotiations. The following actions occur during Phase 1 in the order shown:

1. The characteristics of the Phase 1 security association are negotiated and agreed upon by the IKE peers. These characteristics include the encryption algorithm that will be used to encrypt *the IKE communications*, the hash algorithm to be used, the authentication method, and the Diffie-Hellman group to be used when generating keys.
2. The Diffie-Hellman keys are generated and the public portions are exchanged with the IKE peer. These keys are used to generate encryption keys that will encrypt both the Phase 1 negotiations and will also allow the generation of keys that will be used by IPSec tunnels.
3. The IKE peer is authenticated using one of two supported methods—pre-shared key mode and signature mode.

   In pre-shared key mode, both IKE peers, by means of a previous off-line process, have exchanged a key, and this is used during Phase 1 to authenticate the peer. You configure the pre-shared key using the policy feature's **add user** command.

   In signature mode, a signed X.509 digital certificate is used to provide keys that are used to encrypt and decrypt the payloads of Phase 1 messages.Successful signing and verifying comprises authentication of the peer. For a detailed

discussion of signature mode and the use of X.509 digital certificates, see "Using Public Key Infrastructure" on page 267.

Phase 1 negotiations can take place using either of two exchange modes:

- Main mode uses six messages to perform the Phase 1 negotiations and encrypts the identities of the negotiating peers.
- Aggressive mode uses three messages to perform the Phase 1 negotiations. The peers exchange unprotected identities in the first two messages.

Negotiating an IP Security Tunnel

The processing discussed in this topic occurs when a router prepares to send a packet whose attributes match those defined in a rule in a policy database. Negotiating a tunnel occurs in two phases. During Phase 1, the sending router initiates communication by transmitting the first message of a six-message exchange, which establishes the security options to be used during Phase 2. The receiver responds and the two parties negotiate the ISAKMP security association (SA) characteristics, the authentication and encryption algorithms to be used, and they authenticate each other's identity. During Phase 2, the parties exchange a total of three messages to negotiate the SAs and keys to be used to protect IP datagrams sent between the two. Phase 1 proceeds as follows:

1. Message 1: The sender proposes how the communication activity will take place—the authentication method (for example, digital signatures), the authentication algorithm (for example, HMAC-MD5), and the encryption algorithm (for example, DES-CBC) to be used.
2. Message 2: The receiver indicates to the sender which, if any, of the security options it will support.
3. Message 3: The sender transmits its Diffie Hellman public value and a random value from which encryption keys will be created.
4. Message 4: The receiver transmits its own Diffie Hellman public value and a random value from which encryption keys will be created. At this point, both parties create public and private keys and key-related information to be used in ISAKMP message exchanges.
5. Message 5: The sender transmits a digital signature and may include an X.509 digital certificate signed by a trusted certificate authority (CA). If the sender does not include a valid certificate, the receiver must use the LDAP protocol to obtain a certificate from either a trusted CA, a secure DNS server, a secure local cache that maps previously used certificates to their respective ID values, or may request a certificate from the sender, who must immediately send it.
6. Message 6: After verifying the sender's digital signature, the receiver transmits the same kind of identifying information about itself to the sender.

At this point, both parties have authenticated themselves to the other, agreed on the characteristics of the SA, and have derived keys and key-related information for handling ISAKMP SAs. Now the parties enter Phase 2 to negotiate the non ISAKMP SAs and keys, which will be used to protect IP datagrams exchanged between them. Phase 2 proceeds as follows:

1. Message 1: The sender proposes a non ISAKMP SA by transmitting an AH or ESP algorithm selection, and also includes other security-related information.
2. Message 2: The receiver indicates to the sender which proposal it has selected, and also includes security-related information.
3. Message 3: The sender transmits a hash record of several items to indicate to the receiver that it is ready to proceed using the negotiated security protocols.

When the receiver verifies the information, the link is complete and the parties can begin to exchange protected data streams.

Using Public Key Infrastructure

This section explains how to use the public key infrastructure (PKI). Through PKI, IKE supports public key signature mode for authenticating IKE entities. Although this release supports pre-shared key mode, which does not require PKI support, this mode contains an inherent disadvantage. For authentication, it requires that you configure each IKE entity with the pre-shared key of each of its peers. This severely limits the scalability of IKE operations. Public key-based signature or public encryption mode provides much better scalability. In this release, the X.509 digital certificate is used in signature mode IKE Phase 1 negotiations to authenticate IKE entities.

You assign an identity to each IKE entity that you want to participate in IKE negotiations by specifying a unique value in the ISAKMP ID field when you configure its user policy profile. Each IKE entity authenticates its identity with its peers.

PKI is currently being defined and developed to support public key operation. In PKI, an X.509 digital certificate binds an entity's public key to its claimed identity. An IKE entity can extract the public key contained in a certificate. It can then perform a public key operation to authenticate the identity of a peer that is participating in an IKE negotiation. A public key is used for IKE signature mode. In this mode, the signer uses its private key to sign the digital signature. The receiver extracts the signer's public key from the certificate and uses it to verify the signature. The digital certificate function provides a scalable way for one IKE entity to authenticate the identity of another IKE entity.

Configuring PKI

This release assumes that both IKE entities in a negotiation use the same CA. Before starting IKE negotiations using the signature, you must configure PKI for the router. You must also generate the router private key and router certificate, and have downloaded the root CA's certificate. The following steps explain how to configure PKI:

1. Generate the key pair and request the certificate.

   Because public key operation involves a key pair (signature mode uses the private key to sign and the public key to verify), you must generate a key pair for the router. For a certificate request, you must send the generated public key to the CA to be put into an X.509 digital certificate. Then every potential IKE peer can extract this public key from the CA-issued certificate. The private key resides in the router and is kept secret, known only to the router.

   In this version, you may issue a **certificate request** command, which does the following:

   a. Generates a key pair, whose key length you may specify as either 512, 768, or 1024 bits. The generated private key stays in cache.

   b. Requests that you enter information to include in the certificate request (for example, the router ID in the form of the IP address, domain name, or email name.

   c. Creates a certificate request (in PKCS#10 format) containing the generated public key and the information you have entered.

d. TFTPs the certificate request to a host machine.

2. Issue the certificate (outside the router)

   The CA receives the PKCS#10 certificate request. The CA may manually verify the request and issue a certificate. The certificate contains the router public key and the information that you entered. The CA signs the certificate using its private key, thus it becomes trusted digital information as long as you trust the signing CA. The certificate is now ready to be used in IKE negotiations. (This processing is outside the scope of the router operation and is not discussed in further detail in this book.)

3. Download the router certificate

   Once the CA has issued the certificate, PKI can download it into the router. Depending on how the CA publishes the certificate, PKI can use either TFTP or LDAP to do the download.

   Note that the private key and the public key in the router certificate must match in order to perform public key operation such as digital signature. When PKI downloads the certificate into the router, the private key that was generated with the public key must be in the router key cache. The downloaded certificate is useless if it loses its matching private key. This means that from the time you issue the certificate request to the time the certificate downloads, you **must not** restart or reload the router, clear cache, or issue a new certificate request. Any of these operations destroy the private key in the router running cache.

4. Download the CA certificate

   To verify the IKE peer's certificate, PKI must obtain the peer's root CA certificate. This release supports single level CA operation, which means that the IKE entities must be assigned to the same CA. Each IKE entity (in this case, each router) must download the CA's certificate (using either TFPT or LDAP) to verify that the certificate received from the peer is valid.

5. Save and reload the certificate

   After the router has obtained the certificate, its matching private key, and the CA's certificate, you can start IKE negotiation. Since a certificate is typically valid for months or years, you may want to save the certificate and the private key in SRAM so that you do not have to issue a certificate request and do a download each time you reload or restart the router. This version provides the **cert save** and **cert load** commands to save or retrieve the certificate and private key in SRAM.

   Note that the router certificate and private key must be processed as a pair (for example, they are always saved or retrieved from SRAM together.

Use Talk 6 commands to configure and list both TFTP and LDAP server information as shown in the following examples:

Example: Add Server (T6)

```
Config>f ipsec
IP Security feature user configuration
IPsec config>pki
PKI config>add server
Name ? (max 65 chars) []? test
Enter server IP Address []? 8.8.8.8
Transport type (Choices: TFTP/LDAP)  [TFTP]?
PKI config>
```

Example: List Server Configuration (T6)

```
PKI config>li server

1)  Name: SERVER1
    Type: TFTP
```

```
    IP addr: 8.8.8.8


2)  Name: TEST
    Type: TFTP
    IP addr: 8.8.8.8
```

Example: List Root Certificate (T6)

```
PKI config>li cert

Root CA  certificate:
       SRAM    Name:   R1
       Subject Name:   /c=US/o=ibm/ou=nhd
       Issuer  Name:   /c=US/o=ibm/ou=nhd
           Validity:   1998/12/19 -- 2018/12/19
  Default Root Cert:   No

       SRAM    Name:   R2
       Subject Name:   /c=US/o=ibm/ou=nhd
       Issuer  Name:   /c=US/o=ibm/ou=nhd
           Validity:   1998/12/19 -- 2018/12/19
  Default Root Cert:   Yes



Router   Certificate:
       SRAM    Name:   B1
       Subject Name:   /c=CA/o=Entrust Technologies/ou=PartnerCA/cn=ibm3
       Issuer  Name:   /c=CA/o=Entrust Technologies/ou=PartnerCA
   Subject alt Name:   1.1.1.1
         Key Usuage:   Sign & Encipherment
           Validity:   1998/10/29 -- 2001/10/29
       Default Cert:   No

       SRAM    Name:   B2
       Subject Name:   /c=CA/o=Entrust Technologies/ou=PartnerCA/cn=ibm3
       Issuer  Name:   /c=CA/o=Entrust Technologies/ou=PartnerCA
   Subject alt Name:   1.1.1.1
         Key Usuage:   Sign & Encipherment
           Validity:   1998/10/29 -- 2001/10/29
       Default Cert:   Yes

       SRAM    Name:   B3
       Subject Name:   /c=CA/o=Entrust Technologies/ou=PartnerCA/cn=ibm3
       Issuer  Name:   /c=CA/o=Entrust Technologies/ou=PartnerCA
   Subject alt Name:   1.1.1.1
         Key Usuage:   Sign & Encipherment
           Validity:   1998/10/29 -- 2001/10/29
       Default Cert:   No

       SRAM    Name:   YYY
       Subject Name:   /c=CA/o=Entrust Technologies/ou=PartnerCA/cn=ibm3
       Issuer  Name:   /c=CA/o=Entrust Technologies/ou=PartnerCA
   Subject alt Name:   1.1.1.1
         Key Usuage:   Sign & Encipherment
           Validity:   1998/10/29 -- 2001/10/29
       Default Cert:   No
```

Example: Certificate Request (T5)

```
PKI Console>cert-req
Enter the following part for the subject name
   Country Name(Max 16 characters) []? us
   Organization Name(Max 32 characters) []? IBM
   Organization Unit Name(Max 32 characters) []? NHD
   Common Name(Max 32 characters) []? router1
Key modulus size
 [512]?
Certificate subject-alt-name type:
   1--IPv4 Address
   2--User FQDN
   3--FQDN
Select choice [1]?
Enter an IPv4 addr) []? 12.1.1.1
Generating a key pair. This may take some time. Please wait ...
```

```
PKCS10 message successfully generated
Enter tftp server IP Address []? 8.8.8.8
Remote file name (max 63 chars) [/tmp/tftp_pkcs10_file]?
Memory transfer starting.
.Memory transfer completed - successfully.
Certificate request TFTP to remote host successfully.
Private Key Alias [ROUTER_KEY]? local
Generated private key LOCAL stored into cache
```

Example: List Router Certificate (T5)

```
PKI Console>li cert
Router   certificate
      Serial Number:   909343811
      Subject  Name:   /c=CA/o=Entrust Technologies/ou=PartnerCA/cn=ibm3
      Issuer   Name:   /c=CA/o=Entrust Technologies/ou=PartnerCA
   Subject alt Name:   1.1.1.1
         Key Usuage:   Sign & Encipherment
           Validity:   1998/10/29 -- 2001/10/29

Root CA  certificate
      Serial Number:   914034740
      Subject  Name:   /c=US/o=ibm/ou=nhd
      Issuer   Name:   /c=US/o=ibm/ou=nhd
           Validity:   1998/12/19 -- 2018/12/19
```

Example: Cert Save (T5)

```
PKI Console>cert-save
Enter type of certificate to be stored into SRAM:
     1)Root certificate;
     2)Box  certificate with private key;
Select the certificate type (1-2) [2]?
SRAM Name for certificate and private key []? yyy
Load as default router certificate at initialization?? [No]:
Private key YYY written into SRAM
Both Certificate and private key saved into SRAM successfully
PKI Console>
```

Example: Cert Load (T5)

```
PKI Console>cert-load
Enter type of certificate to be stored into SRAM:
     1)Root certificate;
     2)Box  certificate with private key;
Select the certificate type (1-2) [2]?
Name []? yyy
Box certificate and private key saved into cache successfully
PKI Console>
```

Using Manual IP Security (IPv4)

The IP security feature contained in IPv4 for the 2210, in conjunction with the policy feature and other IPSec-related processes, provides authentication, integrity, confidentiality, and non-repudiation. To implement IPSec manually, you preconfigure a policy containing a subset of IPSec options in a policy database to define the manual tunnel's profile and validity period. You may also preconfigure the full set of IPSec options (policy) in the database so that when a policy-enabled router prepares to send an IPSec packet, it dynamically negotiates and establishes IPSec options with the destination router, based on the policy's contents. To define a manual tunnel, see "Configuring Manual IP Security (IPv4)" on page 277. For an explanation of the policy options, see "Chapter 16. Using the Policy Feature" on page 203.

Using Manual IP Security (IPv6)

The IP security feature contained in IPv6 for the 2210 provides authentication, integrity, and confidentiality. To define a manual tunnel, see “Configuring Manual IP Security (IPv6)” on page 288.

Chapter 19. Configuring and Monitoring IP Security

This chapter describes how to configure and monitor IP security and how to use the IP security monitoring commands. For IPv4, “Chapter 16. Using the Policy Feature” on page 203 and “Chapter 17. Configuring and Monitoring the Policy Feature” on page 233 provide additional information about configuring and monitoring IP security policies. This chapter contains the following sections:

Note: If you create an IPSec tunnel to transport TN3270, APPN-ISR, or APPN-HPR traffic and you plan to prioritize that traffic using BRS, you need to use the IPv4 precedence bit setting feature of BRS. See “Using IP Version 4 Precedence Bit Processing for SNA Traffic in IP Secure Tunnels and Secondary Fragments” on page 9 for more information.

Configuring Internet Key Exchange (IPv4)

This topic explains how to configure Internet Key Exchange (IKE).

Before establishing an IPSec tunnel, you must:

1. Configure the attributes of packets that will use the tunnel and the resulting actions to be taken (the policy).
2. Configure the encryption and authentication options that you want.

For details about doing these tasks, see “Chapter 16. Using the Policy Feature” on page 203, “Chapter 17. Configuring and Monitoring the Policy Feature” on page 233 and “Configuring Public Key Infrastructure (IPv4)” on page 274.

Configuring Public Key Infrastructure (IPv4)

This topic explains how to configure the Public Key Infrastructure (PKI) with IPv4.

Before establishing an IPSec tunnel, you must:

1. Create a public/private cryptographic key pair and obtain a digital certificate from a trusted Certificate Authority (CA). See “Obtaining a Certificate” for details.
2. Decide which IPSec algorithms, SAs, and other options you want to use for the routers whose policies you are configuring. See “Negotiating an IP Security Tunnel” on page 266 and the subsequent topics for details.
3. Configure IKE and the policy database. See “Configuring Internet Key Exchange (IPv4)” on page 273, “Chapter 16. Using the Policy Feature” on page 203, and “Chapter 17. Configuring and Monitoring the Policy Feature” on page 233 for details.

Obtaining a Certificate

Before establishing an IPSec tunnel, you must select and register with a trusted Certificate Authority (CA) as described at “Using Public Key Infrastructure” on page 267. The CA returns a signed X.509 digital certificate, which allows you to identify and authenticate yourself to other parties in the network. The certificate consists of an encoded digital ID (signature) and a public/private cryptographic key pair. Do the following:

1. Identify a CA and obtain its server address.
2. Configure the certificate repository retrieval options using either the PKI Talk 6 **add ldapserver** or **add tftpserver** command as described at “Public Key Infrastructure Configuration Commands” on page 275.
3. Create a public/private key pair using the PKI Talk 5 **certificate request** command as described at “Public Key Infrastructure Monitoring Commands” on page 296. You may do this either in the router or remotely, for example, acting as the Virtual Private Network (VPN) administrator, in which case you must encrypt and securely transfer the key pair into the router.
4. Submit an initial certificate request to the CA using the PKI Talk 5 **certificate request** command as described at “Public Key Infrastructure Monitoring Commands” on page 296. The request is sent in a PKCS#10 message through either email or FTP. The CA binds the key pair into the certificate, signs it with the CA’s private key, and either stores it in a central (LDAP or FTP) repository or returns it to you in a PKCS#7 message. Typically, a certificate is valid for several months or longer, then is renewed. This identifies which parties in a network can still be trusted.
5. Save the certificate into a router’s SRAM using the PKI Talk 5 **certificate save** command as described at “Public Key Infrastructure Monitoring Commands” on page 296.

Notes:

1. To display a list of certificate records in SRAM, use the PKI Talk 6 **list certificate** command as described at “Public Key Infrastructure Configuration Commands” on page 275.

2. To delete certificate records from SRAM, use the PKI Talk 6 **delete certificate** command as described at “Public Key Infrastructure Configuration Commands”.
3. To eliminate the need to resubmit a certificate request during future IPSec negotiations, use the PKI Talk 5 **certificate load** command as described at “Public Key Infrastructure Monitoring Commands” on page 296 to load the received certificate in cache.

Public Key Infrastructure Configuration Commands

Add

Use the PKI Talk 6 **add** command to configure the certificate repository server and its location.

Syntax:

add server

server Specifies that the add operation is for a server.

Example 1: Adding a server

```
PKI config>add server
Name ? (max 65 chars) []? myldap
Enter server IP Address []? 8.8.8.9
Transport type (Choices: TFTP/LDAP)  [TFTP]? ldap
LDAP search timeout value [3]?
LDAP retry interval (mins) [1]?
LDAP server port number [389]?
LDAP version [2]?
Bind to the server anonymously? [No]:
Enter your bind DN: []? c=us o=ibm
Enter your bind PW: []? testldap
```

Change

Use the PKI Talk 6 **change** command to change the certificate repository server and its location.

Syntax:

change server

server Specifies that the add operation is for a server.

Example 1: Changing a server

```
PKI config>change server
Name []? myldap
Enter server IP Address []? 8.8.8.7
Server type will continue to be LDAP
LDAP search timeout value [3]?
LDAP retry interval (mins) [1]?
LDAP server port number [389]?
LDAP version [2]?
Enter your bind DN: [c=us o=ibm]?
Enter your bind PW: [testldap]?
```

Delete

Use the PKI Talk 6 **delete** command to delete a certificate record or a private key record from a router's SRAM, or to delete a server.

Syntax:

delete certificate
private-key
server

certificate
Specifies that the delete operation is for one or more certificate records.

all Specifies that all certificate records are to be deleted.

id Specifies the ID of the certificate record to be deleted.

Example 1: deleting a certificate

```
PKI config>delete certificate
Cert Name []? test
Enter the type of the certificate:
Choices:  1-Root CA Cert, 2-Router Cert
Enter (1-2): [2]?
Box Certificate [TEST] deleted successfully
Corresponding private Key [TEST] deleted successfully
```

Example 2: Deleting private keys

```
PKI config>delete private-keys
Private Key Name []? test
Private Key [TEST] deleted successfully
Corresponding box certificate [TEST] deleted successfully
```

Example 3: Deleting server records

```
PKI config>delete server
Name []? myldap
Server MYLDAP deleted successfully
```

private-key
Specifies that the delete operation is for one or more private key records.

server Specifies that the delete operation is for a server.

List

Use the PKI Talk 6 **list** command to list certificate or key records in a router's SRAM.

Syntax:

list certificates
private-keys
servers

certificates
Specifies that the list operation is for the certificate records.

private-keys
Specifies that the list operation is for the private key records.

servers
Specifies that the list operation is for the server records.

Example 1: Listing certificates

```
PKI config>list certificates

Root CA  certificate:
       SRAM    Name:   B
       Subject Name:   /c=US/o=ibm/ou=nhd
       Issuer  Name:   /c=US/o=ibm/ou=nhd
          Validity:   1998/12/19 2:2:21 -- 2018/12/19 2:32:21
  Default Root Cert:   Yes



Router   Certificate:
       SRAM    Name:   W
       Subject Name:   /c=US/o=ibm/ou=nhd/cn=testip
       Issuer  Name:   /c=US/o=ibm/ou=nhd
   Subject alt Name:   1.1.1.1
         Key Usuage:   Sign & Encipherment
           Validity:   1999/1/19 23:24:27 -- 2002/1/19 23:54:27
       Default Cert:   No
```

Example 2: Listing private keys

```
PKI config>list private-keys
Private Keys In SRAM:

1) Name W
```

Example 3: Listing server records

```
PKI config>list servers
1)  Name: SERVER1
    Type: LDAP
    IP addr: 1.1.1.2
        LDAP search timeout (secs): 10
        LDAP retry interval (mins): 3
        LDAP server port number: 390
        LDAP version: 2
        Anonymous bind ?: y


2)  Name: TEST
    Type: TFTP
    IP addr: 8.8.8.8
```

Configuring Manual IP Security (IPv4)

This section describes the configuration options available for manual IPSec with IPv4. All IPSec functions apply to IPv4.

Do the following steps to configure an IPSec manual tunnel:

1. Create the IPSec tunnel.
2. Reset IPSec.
3. Configure policy for the manual tunnel (profile, validity, policy)
4. Reset Policy.

Configuring the Algorithms

You may configure tunnel policies with the algorithms shown in Table 42.

Table 42. Algorithms Configured with Various Tunnel Policies

Tunnel Policy	Algorithms
AH, AH-ESP, or ESP-AH	• Local AH Authentication Algorithm—Required • Remote AH Authentication Algorithm—Optional
ESP, AH-ESP, or ESP-AH	• Local Encryption Algorithm—Required • Remote Encryption Algorithm—Optional • Local ESP Authentication Algorithm—Optional • Remote ESP Authentication Algorithm—Optional **Note:** If your software load does not include encryption, you will not see encryption-related parameters.

A tunnel policy uses a local algorithm on outbound packets and a remote algorithm on inbound packets. The local algorithm for the router at the near end of a tunnel must match the remote algorithm for the router at the far end of the tunnel. The values for the remote algorithms are optional and they default to the value of the corresponding local algorithms. The local ESP authentication algorithm is optional because ESP authentication is optional.

Configuring Encryption Keys

For each local algorithm you configure, you must also configure a key that is identical to the key for the corresponding algorithm in the remote host. See the description of keys for the **add tunnel** command at “Manual IP Security Configuration Commands”.

Accessing the IP Security Configuration Environment

To access the IP Security configuration environment, enter **t 6** at the OPCON prompt (*), then enter the following sequence of commands at the `Config>` prompt:

```
Config> feature ipsec
IP Security feature user configuration
IPsec config>ipv4
IPV4-IPsec config>
```

Manual IP Security Configuration Commands

This section describes the IP security configuration commands. Enter these commands at the `IPV4-IPsec config>` prompt.

Table 43. IP Security Configuration Commands Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.

Table 43. IP Security Configuration Commands Summary (continued)

Command	Function
Add tunnel	Adds a secure tunnel.
Change tunnel	Changes a secure tunnel configuration parameter values.
Delete tunnel	Deletes a secure tunnel.
Disable	Disables all IP Security processing in a secure manner (packets that match the packet filters are dropped), disables all IP Security processing in a nonsecure manner (packets that match the packet filters are passed), or disables a secure tunnel.
Enable	Enables all IP Security processing, or enables a secure tunnel.
List	Lists information about global IP Security information, or information about defined tunnels.
Set	Sets various IPSec options.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Add Tunnel

Use the **add tunnel** command to add the parameters to define an IPSec tunnel.

Syntax:

add tunnel...

tunnel-name

Optional parameter to label the tunnel. It must be unique within the 2210.

Valid values: up to 15 characters; first character must be a letter; no blanks can be used.

Default value: none

lifetime

Time in minutes that the tunnel can be active. The value 0 indicates that the tunnel lifetime never expires.

Valid Values: 0 - 525600 (0 = no expiration; 525600 = 365 days)

Default Value: 46080 (32 days)

encapsulation-mode

The manner in which the IP packet is encapsulated. In tunnel mode, the entire IP packet is encapsulated and a new IP header is created; in transport mode, the IP header is not encapsulated. If one end of the secure tunnel is a router, then tunnel mode ***must*** be used, according to the Internet Engineering Task Force (IETF) security architecture draft.

Valid Values: tunnel (*TUNN*) or translate (*TRANS*)

Default Value: tunnel (*TUNN*)

tunnel-policy

One of the four choices that define the tunnel policy: IP Authentication Header (AH), IP Encapsulating Security Payload (ESP), or combinations of these protocols (AH-ESP and ESP-AH). In AH-ESP, ESP encryption is run first on the outbound packets; in ESP-AH, AH authentication is run first on the outbound packets. Some parameters are unique either to ESP or AH. The encryption parameters are configured only if ESP, AH-ESP, or ESP-AH is selected; the authentication parameters are configured only if AH, AH-ESP, or ESP with authentication is selected.

Valid Values: AH, ESP, AH-ESP, ESP-AH

Default Value: AH-ESP

local-IP-address
IP address for this end of the tunnel.

Valid Values: a valid IP address that has been configured either for an interface or as the internal address of the 2210.

Default Value: one of the IP addresses configured for the router

local-spi
A security association is a one-way security connection that uses AH or ESP to protect connection traffic. The security parameters index (SPI) is an arbitrary 32-bit value that uniquely identifies one of the two security associations (inbound or outbound) associated with this secure tunnel. This parameter, which is required, identifies the SPI expected in this tunnel for inbound packets received at the local end of the tunnel. This value cannot match the local SPI of another tunnel with the same local IP address. Regardless of the tunnel policy (ESP, AH, AH-ESP, or ESP-AH), only one local SPI is configured for inbound traffic for one IP secure tunnel.

Valid Values: any 32-bit value greater than 255

Default Value: 256

local-encryption-algorithm
The encryption algorithm used for ESP on outbound packets sent from the local router, which is required when configuring ESP. In some countries, some or all of these algorithms may be unavailable because of U.S. export rules. This encryption algorithm must match the remote encryption algorithm.

The ESP-NULL algorithm prevents ESP from performing encryption. This algorithm is available in all countries. If ESP-NULL is selected, ESP must be activated for authentication by selecting one of the authentication algorithms HMAC-MD5 or HMAC-SHA-1.

Valid Values: DES-CBC, CDMF, 3DES, or ESP-NULL

Default Value: DES-CBC

local-encryption-key
The key or keys used with the local ESP encryption algorithm. They must match the corresponding keys that are configured in the opposite end of the secure tunnel. This key is not configured when the ESP-NULL encryption algorithm is selected.

Valid Values:

- For DES-CBC: 16 hex characters (0 - 9, a - f, A - F)
- For CDMF: 16 hex characters (0 - 9, a - f, A - F)
- For 3DES: three separate keys, none of which is the same, each one 16 hex characters (0 - 9, a - f, A - F)

Default Value: none

padding-for-local-encryption
Size in bytes of additional padding that is added to outbound ESP packets. Additional padding may be used to disguise the size of the IP packets being encrypted when the encryption algorithm results in an encrypted packet that is the same size as the original packet. ESP padding values must be a multiple of 8. If a value that is not divisible by 8 is configured, that value is rounded up to the next value that is divisible by 8.

When the encryption algorithm is ESP-NULL, padding is not necessary because the ESP-NULL algorithm adds one byte to the original packet size. If padding for local encryption is configured, the value is ignored.

Valid Values: 0 - 120

Default Value: 0

local-ESP-authentication
Selects local ESP authentication, if desired. Authentication is required if the encryption algorithm is ESP-NULL.

Valid Values: Yes or No

Default Value: Yes

local-authentication-algorithm
The authentication algorithm used on outbound packets. This is an optional parameter for ESP and will not be required unless you select ESP authentication. For AH , AH-ESP, or ESP-AH, this parameter is required. The authentication algorithm used must match the remote authentication algorithm used at the far end of the IPSec tunnel.

Valid Values: HMAC-MD5 or HMAC-SHA

Default Value: HMAC-MD5

local-authentication-key
The key used with the local authentication algorithm. It must match the equivalent key that is configured in the opposite end of the IPSec tunnel. It is required if the policy is AH, AH-ESP, or ESP-AH, or if the policy is ESP and the local ESP authentication algorithm has been configured.

Valid Values:

- for HMAC-MD5: 32 hex characters (0 - 9, a - f, A - F)
- for HMAC-SHA: 40 hex characters (0 - 9, a -f, A - F)

Default Value: none

remote-IP-address
IP address for the remote end of the tunnel. This is a required parameter.

Valid Values: a valid IP address

Default Value: none

remote-spi
A security association is a one-way security connection that uses AH or ESP to protect connection traffic. The security parameters index (SPI) is an arbitrary 32-bit value that uniquely identifies one of the two security associations (inbound or outbound) associated with this secure tunnel. This parameter, which is required, identifies the SPI expected in ESP or AH for outbound packets destined for the remote host. This value cannot match the remote SPI of another tunnel with the same remote IP address. Regardless of the tunnel policy (ESP, AH, AH-ESP, or ESP-AH), only one local SPI is configured for outbound traffic for one IPSec tunnel.

Valid Values: any 32-bit value greater than 255

Default Value: 256

remote-encryption-algorithm
The decryption algorithm used on inbound packets received from the remote host. It must match the local encryption algorithm.

The ESP-NULL algorithm prevents ESP from performing encryption. If ESP-NULL is selected, ESP must be activated for authentication by selecting one of the authentication algorithms HMAC-MD5 or HMAC-SHA-1.

Valid Values: DES-CBC, CDMF, 3DES, or ESP-NULL

Default Value: value of the local encryption algorithm

remote-encryption-key
The key or keys used with the remote ESP encryption algorithm. They must match the equivalent keys that are configured in the opposite end of the secure tunnel. This key is not configured when the ESP-NULL encryption algorithm is selected.

Valid Values:

- For DES-CBC: 16 hex characters (0 - 9, a - f, A - F)
- For CDMF: 16 hex characters (0 - 9, a - f, A - F)
- For 3DES: three separate keys, none of which matches, each 16 characters in hex (0 - 9, a - f, A - F)

Default Value: none

verification-of-remote-encryption-padding
Determines whether the size of the encryption padding on received packets should be verified.

Valid Values: Yes or No

Default Value: No

padding-for-remote-encryption
Size in bytes of additional padding that is expected in received ESP packets. This parameter is required and valid only if the value of *verification-of-remote-encryption-padding* is Yes. ESP padding values must be a multiple of 8. If a value that is not divisible by 8 is configured, that value will be rounded up to the next value that is divisible by 8.

Valid Values: 0 - 120

Default Value: 0

remote-ESP-authentication
Selects remote ESP authentication for inbound packets, if desired.

Valid Values: Yes or No

Default Value: Yes

remote-authentication-algorithm
The authentication algorithm used for inbound packets. This is an optional parameter for ESP and will not be required unless you select ESP authentication. For AH or combinations of AH and ESP (AH-ESP or ESP-AH), this parameter is required. The authentication algorithm used must match the local authentication algorithm used at the far end of the IPSec tunnel.

Valid Values: HMAC-MD5 or HMAC-SHA

Default Value: HMAC-MD5

remote-authentication-key
The key used with the remote authentication algorithm. It must match the equivalent key that is configured in the opposite end of the secure tunnel. It

is required in AH, AH-ESP and ESP-AH and in ESP if the remote ESP authentication algorithm has been configured.

Valid Values:

- for HMAC-MD5: 32 hex characters (0 - 9, a - f, A - F)
- for HMAC-SHA: 40 hex characters (0 - 9, a -f, A - F)

Default Value: none

enable-replay-prevention
Specifies whether replay prevention is enabled. If replay prevention is enabled, the sequence numbers in the IP security headers are monitored to prevent duplicate packets from being processed by the tunnel receiver. The use of replay prevention is not recommended because the tunnel security association must be deactivated when a sender's sequence number counter reaches its limit. When this happens, manual intervention is required to restart the existing security association or create a new one.

In addition, if replay prevention is enabled and you reset IPSec using the **reset ipsec** command, you must make sure that IPSec is also reset on the router at the other end of the IPSec tunnel. This is necessary to re-initialize the sequence number at both ends of the tunnel. If IPSec is reset on one end of the tunnel and not on the other, it is possible that routers at each end of the tunnel will drop packets due to sequence number mismatch.

Valid Values: Yes or No

Default Value: No

DF-bit Specifies the handling of the Don't Fragment (DF) bit in the outer header for tunnel mode secure tunnels. This bit can be set in IPv4 headings to specify that the packet cannot be fragmented. The DF-bit parameter tells the 2210 how it should handle the DF bit on incoming packets - whether to copy the value of the DF-bit found in the inner header to the outer header, or whether to set or clear the bit in the outer header.

If the DF bit is set and the packet cannot be fragmented, IPSec uses the Path MTU (PMTU) Discovery function. See "Path Maximum Transmission Unit Discovery" on page 263 for more information.

Valid Values: Copy, Set, Clear

Default Value: Copy

enable-tunnel
Specifies whether this tunnel is enabled. The enabled tunnel will not filter packets until a packet filter has been configured to define the interface over which this IPSec tunnel will operate and IP has been reset or restarted on the 2210. You can use the **reset ip** command to reset IP.

Valid Values: Yes or No

Default Value: Yes

Change Tunnel

Use the **change tunnel** command to change an IPSec tunnel parameter previously configured by the **add tunnel** command.

Syntax:

change tunnel...	See the **add tunnel** command for a list of the parameters that can be changed.

Delete Tunnel

Use the Talk 6 **delete tunnel** command to delete an IPSec tunnel.

Syntax:

delete tunnel	*tunnel-id*
	tunnel-name
	all

tunnel-id
Specifies the identifier of the IPSec tunnel to be deleted.

Valid Values: 1 - 65535

Default Value: 1

tunnel-name
Specifies the name of the IPSec tunnel to be deleted.

Valid Values: any configured tunnel name

Default Value: none

all Specifies that all IPSec tunnels on this interface are to be deleted.

Disable

Use the **disable** command to disable the IPSec tunnel or to disable all IPSec tunnels either in a secure manner (packets that match the IPSec filters are dropped) or an insecure manner (packets that match the IPSec filters are passed).

Syntax:

disable	ipsec drop
	ipsec pass
	tunnel ...

ipsec drop
Disables IP security on the router in a secure manner. All IPSec tunnels will be disabled, but the secure tunnel information in packet filter rules is used to identify packets that match IPSec tunnel packet filters. The matching packets are dropped.

ipsec pass
Disables IP security on the router in a non-secure manner. All IPSec tunnels will be disabled. Packets that match IPSec tunnel packet filters are forwarded as ordinary traffic.

tunnel *tunnel-id tunnel-name* **all**
Disables IP security on a specified tunnel or on all tunnels.

tunnel-id
Specifies the identifier of the secure tunnel to be disabled.

Valid Values: 1 - 65535

Default Value: 1

tunnel-name
Specifies the name of the secure tunnel to be disabled.

Valid Values: any configured tunnel name

Default Value: none

all All tunnels.

Enable

Use the **enable** command to enable the IP Security protocol on all interfaces or a single tunnel. You must enable IPSec globally on the router before the individually enabled IPSec tunnels become active.

Syntax:

enable ipsec
tunnel ...

ipsec Enables IP security throughout the router.

tunnel *tunnel-id tunnel-name* **all**
Enables IP security on a specified tunnel or on all tunnels.

tunnel-id
Specifies the identifier of the secure tunnel to be enabled.

Valid Values: 1 - 65535

Default Value: 1

tunnel-name
Specifies the name of the secure tunnel to be enabled.

Valid Values: any configured tunnel name

Default Value: none

all All tunnels.

List

Use the **list** command to display the current IP Security configuration. Global tunnels include all tunnels in the router, both active and defined. All tunnels include all tunnels configured on this interface, both active and defined. Active tunnels are those that are currently active; defined tunnels are defined but not active. For IPv4, the selected certificates in a router's SRAM are also listed.

Syntax:

list ... all
status
tunnel
active *tunnel-id tunnel-name* all
defined *tunnel-id tunnel-name* all

Example 1: Listing all IPSec tunnels

```
IPsec config>list all

IPsec is ENABLED

IPsec Path MTU Aging Timer is 20 minutes

Defined Manual Tunnels:

   ID         Name         Local IP Addr   Remote IP Addr   Mode    State
 ------  ---------------  ---------------  ---------------  -----  --------
      1  test                    1.1.1.1          2.1.1.1  TUNN   Enabled
      2  test2                   1.1.1.1          1.1.1.3  TRANS  Enabled

Tunnel Cache:

 ID     Local IP Addr   Remote IP Addr   Mode   Policy  Tunnel Expiration
-----  ---------------  ---------------  -----  ------  ------------------
    2          1.1.1.1          1.1.1.3  TRANS  ESP     ******************
    1          1.1.1.1          2.1.1.1  TUNN   AH      ******************
```

Example 2: Listing an IPSec tunnel with the ESP policy and the ESP-NULL algorithm

```
IPsec config>li tun 1000

Tunnel  Name             Mode   Policy   Life   Replay  Rcv  IPsec  State
ID                                              Prev    Win  Vers
------  ---------------  -----  ------  ------  ------  ---  -----  --------
1000    t1000            TUNN   ESP      46080    No    ---   V2    Enabled

Handling of DF bit in outer header:  COPY

Local Information:

      IP Address: 10.11.12.10
  Authentication:  SPI: -----    Algorithm: ----------
      Encryption:  SPI:  1234    Encryption Algorithm: NULL
                                 Extra Pad:   0
                                 ESP Authentication Algorithm: HMAC-MD5
Remote Information:

      IP Address: 10.11.12.11
  Authentication:  SPI: -----    Algorithm: ----------
      Encryption:  SPI:  1234    Encryption Algorithm: NULL
                                 Verify Pad?:  No
                                 ESP Authentication Algorithm: HMAC-MD5
```

Set

Use the **set** command to control the tunnel PMTU value.

Syntax:

set path-mtu-age-timer

path-mtu-age-timer

Specifies the time (in minutes) that will elapse before the 2210 restores the tunnel PMTU value to the maximum.

Default Value: 10 (0 means disabled)

Configuring a Manual Tunnel (IPv4)

This topic provides information about configuring a manual IPv4 tunnel for the network shown in Figure 27 on page 264.

Configuring the Tunnel for Router A

The following example shows how to configure an IPSec manual tunnel for router A in the network shown in Figure 27 on page 264 using IPv4.

```
Config> feature ipsec
IP Security feature user configuration
IPsec config>ipv4
IPV4-IPsec config>add tunnel
Adding tunnel 1
Tunnel Name (optional)? tunnelone
Tunnel Lifetime, in minutes (0-525600)[46080]?
Tunnel Encapsulation Mode (TUNN or TRANS) [TUNN]?
Tunnel Policy (AH, ESP, AH-ESP, ESP-AH) [AH-ESP]? AH
Local IP Address [1.1.1.1]? 223.252.252.216
Local Authentication SPI (256-65535)[256]?
Local Authentication Algorithm (HMAC-MD5, HMAC-SHA)[HMAC-MD5]?
Local Authentication Key (32 characters) in Hex (0-9,a-f,A-F):
Enter Local Authentication Key again (32 characters) in Hex (0-9,a-f,A-F):
Remote IP Address [0.0.0.0]? 223.252.252.210
Remote Authentication SPI (1-65535) [256]?
Remote Authentication Algorithm (HMAC-MD5, HMAC-SHA)[HMAC-MD5]?
Remote Authentication Key (32 characters) in Hex (0-9,a-f,A-F):
Enter Remote Authentication Key again (32 characters) in Hex (0-9,a-f,A-F):
Enable replay prevention? [No]:
Copy, set, or clear DF bit in outer header (COPY,SET,CLEAR) [COPY]?
Do you wish to enable this tunnel? [Yes]:
IPV4-Ipsec config>
```

As you can see from this example, you are prompted for the parameters that you need to provide. The configuration of an ESP, AH-ESP, or ESP-AH secure tunnel calls for similar parameters.

Note: The values of the keys are not displayed when they are entered. Therefore, they are not visible in this example. If the keys for HMAC-MD5 authentication were visible, you would see 32 hexadecimal characters. For example, a key could have the value: X'1234567890ABCDEF1234567890ABCDEF'.

Configuring the Tunnel for Router B

Within router B, you must configure the same IPSec manual tunnel that was configured for router A, IPSec tunnel 1. The local IP address of this tunnel in router B is 223.252.252.210 and the remote IP address is 223.252.252.216. All other IPSec tunnel parameters must match the parameters that were configured for router A.

Example: Manually Configuring an IP Security Tunnel with ESP

Note that you are prompted to set the DF bit when the tunnel is in tunnel mode and the tunnel policy is ESP. This example shows only the configuration of the IPSec tunnel, not of the packet filters.

```
IPV4-IPsec config>add tunnel
Adding tunnel 2
Tunnel Name (optional)? tunneltwo
Tunnel Lifetime, in minutes (0-525600) [46080]?
Tunnel Encapsulation Mode (TUNN or TRANS) [TUNN]?
Tunnel Policy (AH,ESP,AH-ESP,ESP-AH) [AH-ESP]? ESP
Local IP Address [1.1.1.1]?
Local Encryption SPI (256-65535) [256]?
Local Encryption Algorithm (DES-CBC,CDMF,3DES, NULL) [DES-CBC]?
Do you wish to change the Local Encryption Key? [No]:
Additional Padding for Local Encryption (0-120) [0]?
Do you wish to use local ESP authentication? [Yes]:
Remote IP Address [0.0.0.0]?
Remote Encryption SPI (1-65535) [256]?
```

```
Remote Encryption Algorithm (DES-CBC,CDMF) [DES-CBC]?
Do you wish to change the Remote Encryption Key? [No]:
Do you wish to perform verification of remote encryption padding? [No]:
Do you wish to use remote ESP authentication? [No]:
Copy, set or clear DF bit in outer header (COPY,SET,CLEAR) [COPY]?
Do you wish to enable this tunnel? [Yes]:
IPV4-IPsec config>
```

Example: Manually Configuring an IP Security Tunnel with ESP and ESP-NULL

Note that authentication is required.

```
IPV4-IPsec config>add tunnel
Adding tunnel 3
Tunnel Name (optional)? tunnel3
Tunnel Lifetime, in minutes (0-525600) [46080]?
Tunnel Encapsulation Mode (TUNN or TRANS) [TUNN]?
Tunnel Policy (AH,ESP,AH-ESP,ESP-AH) [AH-ESP]? ESP
Local IP Address [1.1.1.1]?
Local Encryption SPI (256-65535) [256]? 1234
Local Encryption Algorithm (DES-CBC,CDMF,3DES,NULL) [DES-CBC]? null
Additional Padding for Local Encryption (0-120) [0]?
Local ESP Authentication Algorithm (HMAC-MD5,HMAC-SHA) [HMAC-MD5]?
Local ESP Authentication Key (32 characters) in Hex (0-9,a-f,A-F):
Enter Local ESP Authentication Key again (32 characters) in Hex (0-9,a-f,A-F):
Remote IP Address [0.0.0.0]? 10.11.12.11
Remote Encryption SPI (1-65535) [1234]?
Remote Encryption Algorithm (DES-CBC,CDMF,3DES,NULL) [NULL]?
Do you wish to perform verification of remote encryption padding? [No]:
Remote ESP Authentication Algorithm (HMAC-MD5,HMAC-SHA) [HMAC-MD5]?
Remote ESP Authentication Key (32 characters) in Hex (0-9,a-f,A-F):
Enter Remote ESP Authentication Key again (32 characters) in Hex (0-9,a-f,A-F):
Enable replay prevention? [No]:
Copy, set or clear DF bit in outer header (COPY,SET,CLEAR) [COPY]?
Do you wish to enable this tunnel? [Yes]:
IPV4-IPsec config>
```

Configuring Manual IP Security (IPv6)

This section describes the configuration options available for manual IPSec with IPv6. All IPSec functions apply to IPv6. Observe the following changes to the IPSec configuration questions when you are configuring IPSec for IPv6:

- Enter addresses in IPv6 address format (for example, 8:0:9:8::1).
- You are not asked for the DF bit setting.

Do the following steps to configure an IPSec manual tunnel:

1. Create the IPSec tunnel.
2. Reset IPSec.
3. Configure filter rules.
4. Reset IPV6.

Configuring the Algorithms

You may configure tunnel policies with the algorithms shown in Table 44 on page 289.

Table 44. Algorithms Configured with Various Tunnel Policies

Tunnel Policy	Algorithms
AH, AH-ESP, or ESP-AH	• Local AH Authentication Algorithm—Required • Remote AH Authentication Algorithm—Optional
ESP, AH-ESP, or ESP-AH	• Local Encryption Algorithm—Required • Remote Encryption Algorithm—Optional • Local ESP Authentication Algorithm—Optional • Remote ESP Authentication Algorithm—Optional **Note:** If your software load does not include encryption, you will not see encryption-related parameters.

A tunnel policy uses a local algorithm on outbound packets and a remote algorithm on inbound packets. The local algorithm for the router at the near end of a tunnel must match the remote algorithm for the router at the far end of the tunnel. The values for the remote algorithms are optional and they default to the value of the corresponding local algorithms. The local ESP authentication algorithm is optional because ESP authentication is optional.

Configuring Encryption Keys

For each algorithm you configure, you must also configure a key that is identical to the key for the corresponding algorithm in the remote host. See the description of keys for the **add tunnel** command at “Manual IP Security Configuration Commands” on page 278.

Accessing the IP Security Configuration Environment

To access the IP Security configuration environment, enter **t 6** at the OPCON prompt (*), then enter the following sequence of commands at the `Config>` prompt:

```
Config> feature ipsec
IP Security feature user configuration
IPsec config>ipv6
IPV6-IPsec config>
```

Manual IP Security Configuration Commands

See “Manual IP Security Configuration Commands” on page 278 for a description of the IP Security configuration commands available for IPv6. The commands for IPv6 are the same as those used for IPv4 unless indicated otherwise. Enter the commands at the `IPV6-IPsec config>` prompt.

Configuring a Manual Tunnel (IPv6)

Refer to the example network in Figure 27 on page 264 while reading this topic. IPSec tunnel 1 has an endpoint on interface 1 in router A. Router A will be configured for IPSec. Do the following steps to configure router A manually:

1. Create the IPSec tunnel.

2. Create one outbound packet filter on the router interface that is the endpoint of the IPSec tunnel.
3. Create access control rules for the packet filters.
4. Reset IPSec.
5. Reset IPv6.

Creating the IP Security Tunnel for Router A

The following example shows how to create IPSec tunnel 1 for router A. The following example shows how to create IPSec tunnel 1 for router A.

```
Config> feature ipsec
IP Security feature user configuration
IPsec config> ipv6
IPV6-IPsec config> add tunnel
IPsec Tunnel ID (1 - 65535) [1]
Tunnel Name (optional)? tunnelone
Tunnel Lifetime, in minutes (0-525600)[46080]?
Tunnel Encapsulation Mode (TUNN or TRANS) [TUNN]?
Tunnel Policy (AH, ESP, AH-ESP, ESP-AH) [AH-ESP]? AH
Local IP Address [1000:1::1]? 2000::A
Local Authentication SPI (256-65535)[256]?
Local Authentication Algorithm (HMAC-MD5, HMAC-SHA)[HMAC-MD5]?
Local Authentication Key (32 characters) in Hex (0-9,a-f,A-F):
Enter Local Authentication Key again (32 characters) in Hex (0-9,a-f,A-F):
Remote IP Address [0::0]? 2000::B
Remote Authentication SPI (1-65535) [256]?
Remote Authentication Algorithm (HMAC-MD5, HMAC-SHA)[HMAC-MD5]?
Remote Authentication Key (32 characters) in Hex (0-9,a-f,A-F):
Enter Remote Authentication Key again (32 characters) in Hex (0-9,a-f,A-F):
Enable replay prevention? [No]:
Do you wish to enable this tunnel? [Yes]:
IPV6-IPsec config>
```

As you can see from this example, you are prompted for the parameters that you need to provide. The configuration of an ESP, AH-ESP, or ESP-AH secure tunnel calls for similar parameters.

Note: The values of the keys are not displayed when they are entered. Therefore, they are not visible in this example. If the keys for HMAC-MD5 authentication were visible, you would see 32 hex characters. For example, a key could have a value such as X'1234567890ABCDEF1234567890ABCDEF'.

Configuring Packet Filters for Router A

After you have created the IPSec tunnel for router A, you must set up one IP packet filter. The creation of the packet filter *out-router-A* is shown in the following example. Refer to the sections IPv6 Filtering and Access Control in the chapter Using IPv6 in *Protocol Configuration and Monitoring Reference Volume 1* for more information about configuring IPv6 packet filters and access control rules.

```
*talk 6
Config> Protocol IPv6
Internet protocol user configuration
IPv6 Config> set access-control on
IPv6 Config> add packet-filter
Packet-filter name [ ]? out-router-A
Filter incoming or outgoing traffic? [IN]? OUT
Which interface is this filter for [0]? 1
IPv6 Config> update packet-filter
Packet-filter name [ ]? out-router-A
Packet-filter 'out-router-A' Config>
```

Configuring Packet Filter Access Control Rules for Router A

The next step is to configure the packet filter access control rules. Create two access control rules on the outbound packet filter *out-router-A*.

The access control rules on the outbound packet filter perform these functions:

- One access control rule defines the range of the source and destination addresses of the packets to be passed into the IPSec tunnel.
- The other access control rule allows IPSec traffic to pass through the packet filter.

Configure the first access control rule for packet filter *out-router-A*. This access control rule passes packets from network 1000:1:: to the destination network 3000:1:: attached to Router B.

```
IPv6 Config> update packet-filter
Packet-filter name [ ]? out-router-A
Packet-filter 'out-router-A' Config> add access
Enter type [E]? IS
Internet source [0::0]? 1000:1::
Prefix Length [64]? 64
Internet destination [0::0]? 3000:1::
Prefix Length [64]? 64
Enter IPsec Tunnel ID [1]? 2
Packet-filter 'out-router-A' Config>
```

The second access control rule for *out-router-A* allows secured packets to pass between the two ends of the IPSec tunnel.

```
Packet-filter 'out-router-A' Config> add access
Enter type [E]? I
Internet source [0::0]? 2000::A
Prefix Length [64]? 64
Internet destination [0::0]? 2000::B
Prefix Length [64]? 64
Packet-filter 'out-router-A' Config>
```

As with the other packet filters, you may want to configure a wildcard access control rule for *out-router-A* to pass traffic that does not match any access control rules.

Resetting IP Security and IP on Router A

After you finish configuring the policy, use the Talk 5 **reset ipsec** command to reload SRAM with the new IPSec configuration. The **reset ipsec** command does not affect any IP configuration. Then, use the Talk 5 **reset ipv6** command to dynamically reset IPv6 within the router. Alternatively, to reset each component, you can restart the router. You must either reset IPSec and IPv6 or restart the router to ensure that the filter rules are reloaded. Otherwise, your configuration may not be correctly supported on the interface. See “Chapter 19. Configuring and Monitoring IP Security” on page 273 and the **reset ipv6** command in *Protocol Configuration and Monitoring Reference Volume 2* for more information.

As shown in Figure 27 on page 264, IPSec tunnel 2 has an endpoint on interface 1 in Router B. Do the following steps to configure router B manually.

1. Create the IPSec tunnel.
2. Create one outbound filter on the router interface that is the endpoint of the IPSec tunnel.

3. Create access control rules for the packet filters.
4. Reset IPSec.
5. Reset IPv6.

Creating the IP Security Tunnel for Router B

Within router B, the same IPSec tunnel that was created for router A, IPSec tunnel 2, must be created. The local IP address of this tunnel in router B is 2000::B and the remote IP address is 2000::A. All other IPSec tunnel parameters must match the parameters that were specified for router A.

Configuring Packet Filters for Router B

As you did for router A, configure an outbound packet filter (*out-router-B*) on interface 1, which is the interface in router B that is the endpoint of IPSec tunnel 1.

Configuring Packet-Filter Access Control Rules for Router B

Configure an access control rule on *out-router-B* to pass outbound packets from network 3000:1:: to IPSec for processing and transmission through IPSec tunnel 2. This access control rule is type I and S.

```
Packet-filter name [ ]? out-router-B
Packet-filter 'out-router-B' Config> add access
Enter type [E]? IS
Internet source [0::0]? 3000:1::
Prefix Length [64]? 64
Internet destination [0::0]? 1000:1::
Prefix Length [64]? 64
Enter IPsec Tunnel ID [1]? 2
Packet-filter 'out-router-B' Config>
```

Now, for *out-router-B*, create an inclusive access control rule to let packets that have been processed by IPSec pass through IPSec tunnel 2.

```
Packet-filter 'out-router-B' Config> add access
Enter type [E]? I
Internet source [0::0]? 2000::B
Prefix Length [64]? 64
Internet destination [0::0]? 2000::A
Prefix Length [64]? 64
Packet-filter 'out-router-B' Config>
```

For *out-router-B*, create an inclusive wildcard access control rule if you wish to pass rather than drop packets that do not match either of the two access control rules, for example, traffic not destined for IPSec tunnel 2.

Resetting IP Security and IPv6 on Router B

Before the IPSec function will work and the filters are activated, you must reset IPSec and IPv6. Use the talk 5 **reset IPSec** command to reset IPSec and IPv6. See “Resetting IP Security and IP on Router A” on page 291 for information about resetting IPSec. After you reset IPSec, use the talk 5 **reset IPv6** command to reset IPv6. Alternatively, to reset each component, you can restart the router.

Example: Configuring an IP Security Tunnel with ESP

Note that this example shows only the configuration of the IPSec tunnel, not of the packet filters.

```
IPV6-IPsec config>add tun
Tunnel ID or Tunnel Name [ ]? 2
Tunnel Lifetime, in minutes (0-525600) [46080]?
Tunnel Encapsulation Mode (TUNN or TRANS) [TUNN]?
Tunnel Policy (AH,ESP,AH-ESP,ESP-AH) [ESP]?
Local IP Address [0::0]? 2000::A
Local Encryption SPI (256-65535) [256]?
Local Encryption Algorithm (DES-CBC,CDMF,3DES, NULL) [DES-CBC]?
Do you wish to change the Local Encryption Key? (Yes or [No]):
Additional Padding for Local Encryption (0-120) [0]?
Do you wish to use local ESP authentication? [Yes]:
Remote IP Address [0::0]? 2000::B
Remote Encryption SPI (1-65535) [256]?
Remote Encryption Algorithm (DES-CBC,CDMF) [DES-CBC]?
Do you wish to change the Remote Encryption Key? (Yes or [No]):
Do you wish to perform verification of remote encryption padding? [No]:
Do you wish to use remote ESP authentication? [No][No]:
Do you wish to enable this tunnel? [Yes]:
IPV6-IPsec config>
```

Example: Configuring an IP Security Tunnel with ESP and ESP-NULL

Note that authentication is required.

```
IPV6-IPsec config>add tun
Tunnel ID or Tunnel Name [ ]? 2
Tunnel Lifetime, in minutes (0-525600) [46080]?
Tunnel Encapsulation Mode (TUNN or TRANS) [TUNN]?
Tunnel Policy (AH,ESP,AH-ESP,ESP-AH) [ESP]?
Local IP Address [0::0]? 2000::A
Local Encryption SPI (256-65535) [256]?
Local Encryption Algorithm (DES-CBC,CDMF,3DES,NULL) [DES-CBC]? null
Additional Padding for Local Encryption (0-120) [0]?
Local ESP Authentication Algorithm (HMAC-MD5,HMAC-SHA) [HMAC-MD5]?
Local ESP Authentication Key (32 characters) in Hex (0-9,a-f,A-F):
Enter Local ESP Authentication Key again (32 characters) in Hex (0-9,a-f,A-F):
Remote IP Address [0::0]? 2000::B
Remote Encryption SPI (1-65535) [1234]?
Remote Encryption Algorithm (DES-CBC,CDMF,3DES,NULL) [NULL]?
Do you wish to perform verification of remote encryption padding? [No]:
Remote ESP Authentication Algorithm (HMAC-MD5,HMAC-SHA) [HMAC-MD5]?
Remote ESP Authentication Key (32 characters) in Hex (0-9,a-f,A-F):
Enter Remote ESP Authentication Key again (32 characters) in Hex (0-9,a-f,A-F):
Enable replay prevention? [No]:
Do you wish to enable this tunnel? [Yes]:
IPV6-IPsec config>
```

Monitoring Manual IP Security (IPv4)

This section explains how to monitor manual IPSec with IPv4. It describes how to access the Internet Key Exchange environment and the available commands.

Accessing the Internet Key Exchange Environment

This section explains how to use the Internet Key Protocol (IKE) with IPv4.

To access the IP Security IKE monitoring environment, enter the following sequence of commands at the **+** prompt:

```
+ feature ipsec
IPSP>ike
IKE>
```

Internet Key Exchange Monitoring Commands

This section describes the IKE monitoring commands.

Table 45. IKE Monitoring Commands Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Delete	Dynamically deletes a specific tunnel’s ISAKMP Phase 1 SAs, or all Phase 1 SAs.
List	Lists information about a specific tunnel’s Phase 1 SAs or all Phase 1 SAs.
Stats	Displays statistics for a tunnel.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Delete

Use the IKE **delete** command to dynamically delete a Phase 1 SA for a tunnel or all Phase 1 SAs.

Syntax:

delete tunnel

all

tunnel Specifies that a Phase 1 SA is to be deleted for a specific tunnel.

all Specifies that all Phase 1 SAs are to be deleted.

Example: Deleting a Tunnel

```
PKI config>delete tunnel
Peer address [10.0.0.3]?
```

List

Use the IKE **list** command to display information about a specific tunnel’s Phase 1 SAs, or all SAs.

Syntax:

list tunnel

all

tunnel Specifies that information is to be displayed for a specific tunnel’s SAs.

all Specifies that information is to be displayed for all SAs.

Example: Listing Information for all SAs

```
IKE>list all

Phase 1 ISAKMP Tunnels for IPv4:
```

```
-------------------------------------------------------------
 Peer Address    I/R  Mode  Auto    State         Auth
---------------  ---  ----  ----  ----------  ------------
       10.0.0.3   R   Aggr   N    QM_IDLE       pre-shared


IKE>LIST TUNNEL 10.0.0.3

                     Peer IKE address: 10.0.0.3
                    Local IKE address: 10.0.0.1
                                 Role: Responder
                             Exchange: Aggr
                            Autostart: No
                         Oakley State: QM_IDLE
                  Authentication Method: Pre-shared Key
                   Encryption algorithm: des3
                          Hash function: md5
                   Diffie-Hellman group: 1
                      Refresh threshold: 85
                        Lifetime (secs): 15000
```

Stats

Use the IKE **stats** command to display tunnel statistics.

Syntax:

stats tunnel

tunnel Displays statistical information about a tunnel's SAs.

Valid Values: any configured tunnel-name or tunnel-id.

Example: Displaying a Tunnel's SA Statistics

```
IKE>stats

Peer address [10.0.0.3]?

    Peer IP address......:        10.0.0.3
    Active time (secs)...:             187

                                        In              Out
                                        ---             ---
    Octets..............:            1229             1248
    Packets.............:              14               16
    Drop pkts...........:               0                1
    Notifys.............:               6                0
    Deletes.............:               0                0
    Phase 2 Proposals....:              16               18
    Invalid Proposals....:               0
    Rejected Proposals...:               0                0
```

Accessing the Public Key Infrastructure Environment (IPv4)

This section explains how to use the Public Key Infrastructure (PKI) with IPv4.

To access the IP Security PKI monitoring environment, enter the following sequence of commands at the **+** prompt:

```
+ feature ipsec
IPSP>pki
PKI>
```

Public Key Infrastructure Monitoring Commands

This section describes the Public Key Infrastructure (PKI) monitoring commands.

Table 46. PKI Monitoring Commands Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Cert-load	Loads a certificate into a router’s SRAM.
Cert-req	Submits a certificate request to a CA.
Cert-save	Saves a certificate into cache for possible future use.
List certificate	Lists information about a certificate.
List configured-servers	Displays information about the configured servers.
Load certificate	Loads a record containing the certificate from SRAM into the run time cache.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Cert-load

Use the PKI **cert-load** command to load a record containing the certificate and private key from SRAM into the run time certificate cache.

Syntax:

cert-load

Example: Loading a Certificate Record from SRAM into Cache

```
Enter type of certificate to be stored into SRAM:
     1)Root certificate;
     2)Box  certificate with private key;
Select the certificate type (1-2) [2]?
Name []? test
mystr=1.1.1.1
Box certificate and private key saved into cache successfully
```

Cert-req

Use the PKI **cert-req** command to request a certificate from a CA.

Syntax:

cert-req

Example: Requesting a Certificate from a CA

```
Enter the following part for the subject name
   Country Name(Max 16 characters) []? us
   Organization Name(Max 32 characters) []? ibm
   Organization Unit Name(Max 32 characters) []? nhd
   Common Name(Max 32 characters) []?
Key modulus size (512|768|1024)
 [512]?
Certificate subject-alt-name type:
   1--IPv4 Address
   2--User FQDN
   3--FQDN
Select choice [1]?
Enter an IPv4 addr) []? 1.1.1.1
Generating a key pair. This may take some time. Please wait ...
PKCS10 message successfully generated
Enter tftp server IP Address []? test
```

```
Bad address, try again
Enter tftp server IP Address []? 8.8.8.8
Remote file name (max 63 chars) [/tmp/tftp_pkcs10_file]?
Certificate request TFTP to remote host successfully.
```

Cert-save

Use the PKI **cert-save** command to save a record containing the certificate and private key into SRAM.

Syntax:

cert-save

Example: Saving a Certificate Record into SRAM

```
Enter type of certificate to be stored into SRAM:
     1)Root certificate;
     2)Box  certificate with private key;
Select the certificate type (1-2) [2]?
SRAM Name for certificate and private key []? test
Load as default router certificate at initialization? [No]:
Private key TEST written into SRAM
Both Certificate and private key saved into SRAM successfully
```

List Certificate

Use the PKI **list certificate** command to display information about an X.509 digital certificate.

Syntax:

list certificate

Example: Listing certificate information

```
Router   certificate
      Serial Number:   914034877
      Subject  Name:   /c=US/o=ibm/ou=nhd/cn=testip
      Issuer   Name:   /c=US/o=ibm/ou=nhd
   Subject alt Name:   1.1.1.1
         Key Usuage:   Sign & Encipherment
           Validity:   1999/1/19 23:24:27 -- 2002/1/19 23:54:27
```

List Configured-servers

Use the PKI **list configured-servers** command to display information about the configured servers.

Syntax:

list configured-servers

Example: Listing Information about Configured Servers

```
1)  Name: SERVER1
    Type: LDAP
    IP addr: 0.0.0.0
        LDAP search timeout (secs): 0
        LDAP retry interval (mins): 0
        LDAP server port number: 0
        LDAP version: 0
        LDAP version: 0
        Anonymous bind ?: y
```

```
2)  Name: TEST
    Type: TFTP
    IP addr: 9.9.9.9


3)  Name: TFTP
    Type: TFTP
    IP addr: 2.2.2.2
```

Load Certificate

Use the PKI **load certificate** command to load a certificate from SRAM into the run time cache.

Syntax:

load certificate

Example: Loading a Certificate into Cache

```
Enter the type of the certificate:
Choices:  1-Root CA Cert, 2-Router Cert
Enter (1-2): [2]?
Encoding format:
Choices: 1-DER 2-PEM
Enter (1-2): [1]?
Server info name []? test
Remote file name on tftp server (max 63 chars) [/tmp/default_file]? /tmp/test.cert

Attempting to load certificate file. Please wait ...
Router Certificate  loaded into run-time cache
```

Accessing the IP Security Monitoring Environment (IPv4)

To access the IPv4 IP Security monitoring environment type **t 5** at the OPCON prompt (*):

```
* t 5
```

Then, enter the following sequence of commands at the **+** prompt:

```
+ feature ipsec
IPSP>ipv4
IPV4-IPsec>
```

IP Security Monitoring Commands (IPv4)

This section describes the IP Security monitoring commands.

Table 47. IP Security Monitoring Commands Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Change tunnel	Dynamically changes a secure tunnel configuration parameter values.
Delete tunnel	Dynamically deletes a secure tunnel.

Table 47. IP Security Monitoring Commands Summary (continued)

Command	Function
Disable	Dynamically disables all IP Security processing in a secure manner (matching packets are dropped), disables all IP Security processing in a nonsecure manner (matching packets are forwarded), or disables a particular secure tunnel.
Enable	Dynamically enables all IP Security processing, or enables a secure tunnel.
List	Lists global information about IP Security, about active and defined tunnels.
Reset	Resets IP Security or resets a secure tunnel. This command reloads the configuration that was created in Talk 6. Resetting will override the values of parameters configured using Talk 5 with those that were configured using Talk 6.
Set	Dynamically sets the Path MTU (PMTU) aging timer.
Stats	Displays statistics for all tunnels or for an active tunnel.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Change Tunnel

Dynamically changes a secure tunnel.

Syntax:

change tunnel ...

See the description of the **add tunnel** command under “Manual IP Security Configuration Commands” on page 278 for a description of the parameters.

Delete Tunnel

Use the **delete** command to dynamically delete a secure tunnel or all secure tunnels.

Syntax:

delete tunnel

tunnel-id

tunnel-name

all

tunnel-id
Specifies the identifier of the IPSec tunnel to be deleted.

Valid Values: 1 - 65535

Default Value: 1

tunnel-name
Specifies the name of the IPSec tunnel to be deleted.

Valid Values: any configured tunnel name

Default Value: none

all Specifies that all IPSec tunnels on this interface are to be deleted.

Disable

Use the **disable** command to dynamically disable the IP Security protocol on all interfaces or a single tunnel.

Syntax:

disable ipsec drop
ipsec pass
tunnel ...

ipsec drop
Disables IP security on the router in a secure manner. All IPSec tunnels will be disabled, but the secure tunnel information in packet filter rules is used to identify packets that match IPSec tunnel packet filters. The matching packets are dropped.

ipsec pass
Disables IP security on the router in a non-secure manner. All IPSec tunnels will be disabled. Packets that match IPSec tunnel packet filters are forwarded as ordinary traffic.

tunnel *tunnel-id* **all**
Disables IP security on a specified tunnel or on all tunnels.

tunnel-id
Specifies the identifier of the secure tunnel to be disabled.

Valid Values: 1 - 65535

Default Value: 1

all All tunnels.

Enable

Use the **enable** command to dynamically enable the IP Security protocol on all interfaces or a single tunnel. You must enable IPSec globally on the router before the individually enabled IPSec tunnels become active.

Note: IPSec cannot be dynamically enabled if the router was restarted with IPSec disabled.

Syntax:

enable ipsec
tunnel ...

ipsec Enables IP security throughout the router.

tunnel *tunnel-id* **| all**

tunnel-id
Specifies the identifier of the secure tunnel to be enabled.

Valid Values: 1 - 65535

Default Value: 1

all All tunnels.

List

Use the **list** command to display the current IP Security configuration. Global tunnels include all tunnels in the router, both active and defined. All tunnels include all tunnels configured on this interface, both active and defined. Active tunnels are those that are currently active; defined tunnels are defined but not active.

Syntax:

list ... all
global
tunnel
active *tunnel-id tunnel-name* all
defined *tunnel-id tunnel-name* all

Example 1: Listing all active tunnels

```
IPV6-IPsec>li tunnel ?
ACTIVE
DEFINED
IPsec>li tunnel active
Enter the Tunnel ID, Tunnel Name, or 'ALL' [ALL]? all

Tunnel Cache:

 ID     Local IP Addr   Remote IP Addr   Mode   Policy  Tunnel Expiration
-----  ---------------  ---------------  -----  ------  ------------------
    2          1.1.1.1          1.1.1.3  TRANS  ESP     ****************
    1          1.1.1.1          2.1.1.1  TUNN   AH      ****************
```

Example 2: Listing one active tunnel that has received a “packet too big” message

```
IPV6-IPsec>li tun act 1

Tunnel      Name       Mode  Policy  Life  Replay       Tunnel       PMTU
  ID                                       Prev      Expiration
------ --------------- ----- ------ ------ ------ ------------------ -----
     1 tofran2         TUNN  AH      46080   No   10:49  May  8 1998 1420 [1]

Local Information:

      IP Address: 2001:1::6101 [2]
      Authentication:  SPI:   257    Algorithm: HMAC-MD5
      Encryption:  SPI: -----    Encryption Algorithm: --------
                                 Extra Pad: ---
                                 ESP Authentication Algorithm: ----------
Remote Information:

      IP Address: 2001.1..86
  Authentication:  SPI:   257    Algorithm: HMAC-MD5
      Encryption:  SPI: -----    Encryption Algorithm: --------
                                 Verify Pad?: ---
                                 ESP Authentication Algorithm: ----------
```

1 PMTU is displayed as n/a if no packet too big has been received.

2 This is an IPv6 address. If the IP version is IPv4, a message is displayed that defines the handling of the DF bit: COPY, SET, or CLEAR.

Example 3: Listing all tunnels

```
IPV6-IPsec>li all

IPsec is ENABLED

IPsec Path MTU Aging Timer is 30 minutes

Defined Manual Tunnels for IPv4:
--------------------------------------------------------------------------------
   ID         Name         Local IP Addr   Remote IP Addr   Mode    State
 ------  ---------------  ---------------  ---------------  -----  --------

Defined Manual Tunnels for IPv6:
--------------------------------------------------------------------------------
ID=       1  Name= tofran2           Mode= TUNN    State= Enabled
Local IP address=  2001:1::6101
Remote IP address= 2001:1::86


Tunnel Cache for IPv4:
--------------------------------------------------------------------------------
 ID     Local IP Addr   Remote IP Addr   Mode   Policy  Tunnel Expiration
-----  ---------------  ---------------  -----  ------  ------------------

Tunnel Cache for IPv6:
--------------------------------------------------------------------------------
ID=     1  Mode= TUNN   Policy= AH      Expiration= 10:49  May  8 1998
Local IP Address= 2001:1::6101
Remote IP Address= 2001:1::86
```

Reset

Use the **reset** command to dynamically reset IP security on the router or on a single tunnel. After you reset IPSec or the tunnels, be sure to use the **reset IP** command to reset the IP configuration. This is necessary to reload the access control information, such as packet filters and their access control rules. If you do not reset IP, the packet filters and access control rules may not support your new IPSec configuration.

Rebooting the router is an alternative to using the **reset** commands. However, rebooting the router takes it off the network for a time, whereas the **reset** commands interrupt only IP functions.

Syntax:

reset ipsec

tunnel *tunnel-id tunnel-name* all

ipsec Resets IP security on the 2210. IP security is temporarily disabled and then restarted. While IP security is disabled, any packets that are normally handled by IPSec tunnels are dropped until the reset is complete. Resetting IP security does not affect other functions on the 2210. This command activates the IP security configuration that was created using Talk 6. The Talk 6 IP security configuration overwrites the Talk 5 configuration.

tunnel Resets IP security on a specified tunnel. If the tunnel is disabled at the time of reset, the tunnel configuration is rebuilt from the SRAM configuration, but the tunnel remains disabled after the reset.

tunnel-id

Specifies the identifier of the secure tunnel to be reset.

Valid Values: 1 - 65535

Default Value: 1

tunnel-name
Specifies the name of the secure tunnel to be reset.

Valid Values: any configured tunnel name

Default Value: none

all All tunnels.

Set

Dynamically sets the Path MTU (PMTU) aging timer.

Syntax:

set path

path This parameter defines the time in minutes that will elapse before the 2210 sets the tunnel MTU back to the maximum.

Default Value: 10 (0 means disabled)

Stats

Use the **stats** command to display statistics about a specific tunnel or all tunnels. For example, the **stats** command shows packets sent and received.

Syntax:

stats *tunnel-id*

tunnel-name

all

tunnel-id
Specifies the identifier of the secure tunnel.

Valid Values: 1 - 65535

Default Value: 1

tunnel-name
Specifies the name of a secure tunnel that has been configured.

Valid Values: any configured tunnel name

Default Value: none

all Displays statistics about all tunnels configured on the 2210.

Example:

```
IPV6-IPsec>stats
Enter the Tunnel ID, Tunnel Name, or 'ALL' [ALL]? all

                          Global IPSec Statistics
Received:
  total pkts   AH packets   ESP packets   total bytes    AH bytes     ESP bytes
  ----------   ----------   -----------   -----------   ----------   ----------
           0            0             0             0            0            0

Sent:
  total pkts   AH packets   ESP packets   total bytes    AH bytes     ESP bytes
  ----------   ----------   -----------   -----------   ----------   ----------
           0            0             0             0            0            0

Receive Packet Errors:
  total errs    AH errors   AH bad seq   ESP errors   ESP bad seq
  ----------   ----------   ----------   ----------   -----------
```

```
          0           0           0           0           0

Send Packet Errors:
  total errs    AH errors   ESP errors
  ----------   ----------   ----------
          0           0           0
```

Monitoring Manual IP Security (IPv6)

This section explains how to monitor manual IPSec with IPv6. It describes how to access the IP security environment and the available commands.

Accessing the IP Security Monitoring Environment

To access the IP Security monitoring environment type **t 5** at the OPCON prompt (*):

```
* t 5
```

Then, enter the following sequence of commands at the **+** prompt:

```
+ feature ipsec
IPSP>ipv6
IPV6-IPsec>
```

IP Security Monitoring Commands (IPv6)

The IP Security monitoring commands for IPv6 are the same as those used for IPv4 unless indicated otherwise. See “IP Security Monitoring Commands (IPv4)” on page 298 for a description of the commands. Enter the commands at the `IPV6-IPsec>` prompt.

Chapter 20. Using the Differentiated Services Feature

This chapter describes how to use the Differentiated Services (DiffServ) feature so that a router can provide preferred service to appropriate IP data packets. Based on information in the IP header, the router classifies packets by matching them with predefined configurations in the policy database (created with the policy feature). See “Chapter 16. Using the Policy Feature” on page 203 for details. As a result, some packets may receive preferred service. This chapter consists of the following sections:

Overview of Differentiated Services

Most forwarding devices installed in an IP network today deliver standard best-effort service to data packets on a first-come, first-served basis. This delivery method is adequate for most traffic, but new applications are emerging that require faster and earlier transmission of certain packets.

The Differential Services (DiffServ) feature provides different levels of service to IP packets when a router processes them for transmission. DiffServ provides some packets with preferred service by reserving system resources (buffers) and link resources (bandwidth) for them. A DiffServ classifier function determines the type of service given to IP packets by examining various fields in the IP header, for example, ranges of IP source and destination addresses and port numbers, protocol type, and incoming TOS byte. To accomplish this in a scalable way, individual flows are aggregated into streams. Streams are the entities through which DiffServ manages access to buffers and bandwidth. Figure 28 shows how DiffServ processes the packets of a stream.

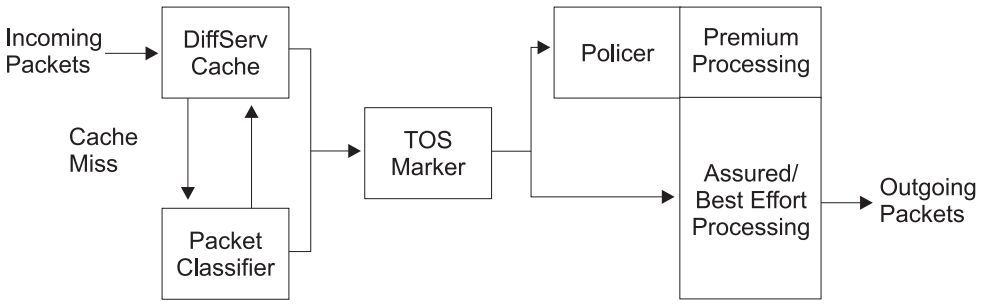


Figure 28. DiffServ Data Packet Path

In addition to the traditional best-effort service, DiffServ provides the following types of service:

Expedited Forwarding (EF)

Expedited forwarding service represents the DiffServ implementation of premium service and both terms are used interchangeably in the following text. This service guarantees a specific transmission rate and lower delay than either assured forwarding or best effort service. If excess traffic

develops, DiffServ drops the excess traffic. The premium queue provides EF service and is shown in Figure 29 as the EF queue.

Assured Forwarding (AF)
Assured forwarding service represents the DiffServ implementation of assured service and both terms are used interchangeably in the following text. This service guarantees a specific transmission rate, but no delay guarantee. If idle resources exist, DiffServ can send excess traffic at a higher rate. The AF/BE queue provides AF service and is shown in Figure 29.

Best Effort (BE)
This is the standard best-effort service, which does not provide service or delay guarantees. You must strike a balance between reserving resources for EF and AF services, and leaving enough resources free so that best effort traffic receives adequate service. The AF/BE queue provides BE service and is shown in Figure 29.

Local routers create and send control packets, so you must also leave enough resources free so that they receive adequate service.

DiffServ is currently implemented on PPP and Frame Relay links, and can be used by the RSVP subsystem. Figure 28 on page 305 shows how packets of a stream are processed. When a router receives the first packet of a flow (assuming that it is designated for premium service), no indication of its service category exists in the fast path cache, so the packet is processed by the slow path. DiffServ invokes a search of the policy database to obtain the packet-handling criteria (policy). The policy-defined action is saved in the fast-path cache. When the router receives a subsequent packet of this flow, it finds that an entry in the fast-path cache for the flow already exists, so its policy-defined action is applied and the packet takes the fast path. Thus, subsequent packets from this flow receive premium service.

Figure 29 shows the relationship between the policer, buffer management, the queues, and the scheduler—some of the basic components that provide different quality of service levels.

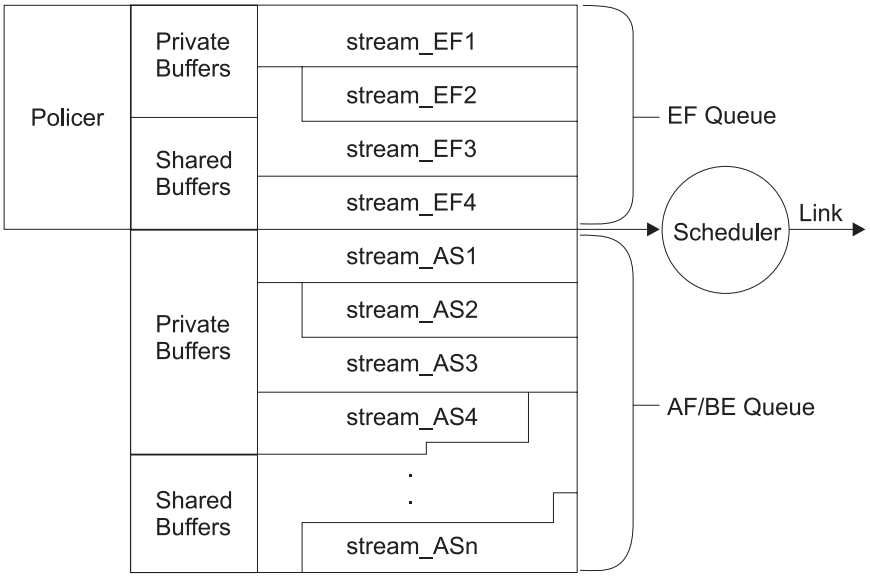


Figure 29. Relationship of Buffers, Queues, and Scheduler

The expedited forwarding (EF) and assured forwarding (AF) services have different characteristics, which are supported by three functions in the router: (1) The policer, (2) buffer management, and (3) the scheduler. These functions provide more sophisticated traffic control than is available in a traditional BE router device.

If traffic requires EF processing, any excess traffic must be dropped. A DiffServ *policer* function uses a token bucket to examine EF traffic and to determine whether a packet is in excess. If it is, the policer drops the packet.

If traffic is for AF, BE, or is EF traffic that the policer has allowed, the rate-based *buffer management* function processes it. This function allocates buffers from either a private pool for the interface, or from a common shared pool for all the interfaces. Use the Talk 6 **set receive-buffers** configuration command (see the set receive-buffers command in *Software User's Guide*) to specify the total amount of physical buffer space available to an interface. Use the DiffServ Talk 6 **set interface** command to set the egress buffer size for the premium and assured queues. This is the buffer space that DiffServ manages. (DiffServ manages two separate pools—one for the premium (EF) queue and one for the assured forwarding (AF) queue. Ensure that the buffer space you specify reflects the actual amount of buffer space available in the system.) Buffer management determines whether buffers from its interface's private pool are available for the packet. If there are, it accepts and enqueues the packet. If they are not, it attempts to allocate buffer space from the shared pool and if it can, it enqueues the packet. If no shared buffer space is available, buffer management drops the packet.

The *scheduler* function examines the queues on a regular basis, dequeues enqueued packets, and sends them to the interface adapter for transmission. It is a self-clocked fair-queuing scheduler, which is a variation of weighted fair queuing. You may configure the scheduler weights and specify the frequency at which the scheduler examines the queues.

Note: It is possible to configure DiffServ options such that the resources are overcommitted or overbooked, that is, the traffic conditioner controls are configured as though there were more bandwidth or buffering than is actually available. DiffServ does not support overbooking.

If a DiffServ stream becomes idle (no packets have been sent on the stream for some time), the system reclaims the resources so other streams can use them. If the stream reactivates, the resources are returned to it. If the resources are no longer available because of overbooking, then DiffServ attempts periodically to reallocate the resources.

After you have used the policy feature to configure appropriate policies, the first step in implementing DiffServ is to use the DiffServ **set** command to configure the options that define the system resources available to DiffServ. Then use the **enable ds** command to enable the DiffServ feature, and the **enable interface** command to enable the egress interface.

Differentiated Services Terminology

The following terms are used when discussing DiffServ:

DiffServ Cache
This cache contains the traffic and service profile of the most recently active IP flows being serviced by the router.

Flow A sequence of packets with the same source address and port, IP protocol, and destination address and port.

Stream
An aggregation of flows.

Virtual Interface (VIF)
For Frame Relay links, each DLCI connection is considered to be a virtual interface.

Configuring Differentiated Services

The following procedures provide a high-level description of how to configure DiffServ to provide preferred service for selected packets. First, access the DiffServ feature:

1. At the * prompt, enter **talk 6**.
2. At the Config> prompt, enter **feature ds**. This displays the DS config> prompt and opens the configuration dialog.

```
* talk 6
Config>feature ds
DS config>
```

3. Enable the DiffServ feature on a router:

```
DS config>enable ds
DiffServ enabled
```

4. Enable and set the interface parameters:

```
DS config>set interface
Enter Diffserv Interface number [0]? 2
Set Premium Queue Bandwidth (%) (1 - 99) [20]?
    Assured Queue Bandwidth (%) = 80
Configure Advanced setting (y/n)? [No]: no
Accept input (y/n)? [Yes]:
```

Note: If you specify no to the Configure Advanced setting prompt, then default parameters for Premium Queue and Assured/BE queue will be used.

```
Configure Advanced setting (y/n)? [No]: yes
Set Premium Queue Weight (%) (20 - 99) [90]?
    Assured Queue Weight (%) = 10
EGRESS BufSize for Premium Queue (in bytes) (550 - 16500) [5500]?
Max EGRESS QoS Allocation for Premium Queue (%) (1 - 99) [95]?
EGRESS BufSize for Assured/BE Queue (in bytes) (5500 - 140800) [27500]?
Max EGRESS QoS Allocation for Assured/BE Queue (%) (1 - 99) [80]?
```

In this example, 20 percent of line bandwidth, and 90 percent of scheduler weight are given to the EF queue. The egress buffer size for the EF queue is 5500 (in bytes), out of which 95 percent is allocatable to QOS streams. The egress buffer size for the AF/BE queue is 27500 (in bytes), out of which 80 percent is allocatable to QOS streams.

5. When you have finished enabling DiffServ on routers and setting interface parameters, enter **Ctrl-P** to return to the * prompt.

After enabling DiffServ and setting interface parameters, you must restart or reload the device to activate DiffServ. For more details on specifying DiffServ commands, see “Chapter 21. Configuring and Monitoring the Differentiated Services Feature” on page 311.

Chapter 21. Configuring and Monitoring the Differentiated Services Feature

This chapter describes the commands provided by the Differentiated Services (DiffServ) feature for configuring routers and interfaces to provide preferred service for selected data packets. It includes the following sections:

Accessing the Differentiated Services Configuration Prompt

To enter DiffServ configuration commands:

1. Enter **talk 6** at the OPCON (*) prompt.
2. Enter **feature ds** at the Config> prompt.

The DS Config> prompt displays. You may now enter DiffServ configuration commands.

Differentiated Services Configuration Commands

These commands enable you to configure the DiffServ options, which designate preferred service for selected data packets. Table 48 summarizes the DiffServ configuration commands, and the rest of this section describes them in detail. Enter the commands at the DS Config> prompt. Either enter the command and options on one line, or enter only the command and then respond to the prompts. To see a list of valid command options, enter the command with a question mark instead of options.

Table 48. DiffServ Configuration Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page xxix.
Delete	Deletes a DiffServ configuration record from a router's SRAM.
Disable	Disables DiffServ either in a router or on a specific egress interface.
Enable	Enables DiffServ either in a router or on a specific egress interface.
List	Displays information about a router's DiffServ system and interface-related settings.
Set	Specifies a router's DiffServ-related settings.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page xxix.

Delete

Use the **delete** command to delete a DiffServ system configuration record or interface record from a router's SRAM.

Syntax: delete ds

interface

ds Deletes the router's DiffServ system configuration record.

Example:

```
DS Config> delete ds
Diffserv system config record deleted
```

interface Prompts you for the interface number to delete.

Example:

```
DS Config> delete interface
Enter Diffserv Interface number to delete [0]? 3
Diffserv interface config record deleted
```

Disable

Use the **disable** command to disable the DiffServ function either in a router or on a specific egress interface.

Syntax: disable ds

interface

ds Disables the router's DiffServ function.

Example:

```
DS Config> disable ds
DiffServe feature disabled
```

interface Prompts you for the number of the interface to disable.

Example:

```
DS Config> disable interface
Enter Interface number [0]? 2
DiffServe interface disabled
```

Enable

Use the **enable** command to enable the DiffServ function either in a router or on a specific egress interface.

Syntax: enable ds

interface

ds Enables the router's DiffServ function.

Example:

```
DS Config> enable ds
DiffServe feature enabled
```

interface Prompts you for the number of the interface to enable.

Example:

```
DS Config> enable interface
Enter Interface number [0]? 2
DiffServe interface enabled
```

Note: DiffServ can be enabled only on PPP and Frame Relay links.

List

Use the **list** command to display information about a router's DiffServ system and interface-related settings.

Syntax: list
- all
- ds
- interface

all Displays information about a router's DiffServ and interface configurations.

ds Displays a router's DiffServ configuration.

Example:

```
DS Config> list ds

System Parameters:

        DiffServ:               ENABLED
        Packet_size:            550
        Min BE_Alloc (%):       10
        Min CTL Alloc (%):      5
        Number_of_Q:            2
```

interface Displays the interfaces in a router, their DiffServ enable/disable status, and the parameters for each interface and queue.

Example:

```
DS Config> list interface
-------------------------------------------------------------------------------
                              ---------- Premium -------   --------- Assured --------
Net If     Status  NumQ Bwdth  Wght  OutBuf  MaxQos Bwdth  Wght  OutBuf  MaxQos
Num                       (%)   (%) (bytes)     (%)   (%)   (%) (bytes)     (%)
-------------------------------------------------------------------------------
2   PPP   Enabled     2    20    90    5500      95    80    10   27500      80
3   PPP   Enabled     2    20    90    5500      95    80    10   55000      80
```

Set

Use the **set** command to set a router's DiffServ system and interface-related parameters.

Syntax: set
- be-alloc-min
- ctl-alloc-min
- interface
- pkt-size

be-alloc-min Specifies the minimum percentage of total output buffer space to allocate to best effort service.

Default value: 10

Example:

```
DS Config> set be-alloc-min
Enter Minimum percent output BW allocated to BE service (10 - 50) [10]?
```

ctl-alloc-min Specifies the minimum percentage of total output buffer space to allocate to network control service.

Default value: 5

Example:

```
DS Config> set ctl-alloc-min
Enter Minimum percent output BW allocated to CTL service (5 - 20) [5]?
```

interface Specifies the interface to enable for DiffServ and prompts you for interface-specific parameters.

Queue bandwidth
Specifies the percentage of the output link to be used for the premium queue. The remaining percentage is used for the assured queue value.

Default value: 20

Queue weight
Specifies the percentage of time that the scheduler monitors the premium queue. The remaining percentage is used for the assured queue value. The queue weight is defaulted to 90 percent so that the scheduler reacts quickly to EF traffic.

Default value: 90

Egress buffer size
Specifies the amount of data (in bytes) that can be queued on the premium queue and the assured queue.

For the premium queue, this parameter controls the amount of data (in bytes) that can be queued on the premium queue. Too large a value for this parameter could cause a high queuing delay for the premium traffic. For example, if this is set to 25 Kbytes and the output link speed is 1.5 Mbps (T1 speed), then there is a potential queuing delay of 133 msec (25000 bytes * 8 bits/byte)/1500000 bps, or .133 sec (133 milliseconds). Too small a value for this parameter could make it impossible to buffer small bursts. For example, if this is set to 2 Kb, it implies that there will not be sufficient buffering for a 2-packet burst of 1500-byte packets (because they require 3000 bytes of buffer space).

As a compromise between these two extremes, the default setting is 5500 bytes, which is ten times the default packet size of 550.

Default value: 5500 (premium queue)

For the assured queue, this parameter controls the amount of data (in bytes) that can be queued on the assured queue. The considerations for this parameter value are the same as for the premium queue, except that the traffic in the assured queue does not have very strict delay requirements. Rather, it is more likely that assured queue traffic will consist of TCP flows, which are bursty in nature. Because of this, enough buffer space must be defined to accommodate bursts from several flows.

A value of 27500, which is fifty times the default packet size of 550.

Default value: 27500 (assured queue)

Egress QoS allocation
Specifies the amount of the egress buffer size value (as a percentage) that all the DiffServ streams can reserve. The remaining percentage is used for the minimum size of the shared pool.

Default value: 95 (premium queue)

Default value: 80 (assured queue)

Example:

```
DS Config> set interface
Enter Diffserv Interface number [0]? 2

DiffServ Interface enabled

Set Premium Queue Bandwidth (%) (1 - 99) [20]?
    Assured Queue Bandwidth (%) = 80

Configure Advanced setting (y/n)? [No]: y

Set Premium Queue Weight (%) (20 - 99) [90]?
    Assured Queue Weight (%) = 10

EGRESS BufSize for Premium Queue (in bytes) (550 - 16500) [5500]?
Max EGRESS QoS Allocation for Premium Queue (%) (1 - 99) [95]?

EGRESS BufSize for Assured/BE Queue (in bytes) (5500 - 140800) [27500]?
Max EGRESS QoS Allocation for Assured/BE Queue (%) (1 - 99) [80]?

        DiffServ Interface: ENABLED
PREMIUM Queue Bandwidth (%) =                          20
PREMIUM Queue Weight (%) =                             80
PREMIUM Queue EGRESS BufSize in bytes =              5500
PREMIUM Queue Max EGRESS QoS allocation (%) =          95
ASSURED/BE Queue Bandwidth (%) =                       80
ASSURED/BE Queue Weight (%) =                          20
ASSURED/BE Queue EGRESS BufSize in bytes =          27500
ASSURED/BE Queue Max EGRESS QoS allocation (%) =       80
Accept input (y/n)? [Yes]:
```

pkt-size Specifies the average packet size of the traffic flow (in bytes). This enables DiffServ to determine the available buffer space on the ingress and egress interfaces. If this is changed, the router must be restarted and the DiffServ **set interface** command values should be reviewed and changed if necessary.

Default value: 550

Example:

```
DS Config> set pkt-size
Average packet size (64 - 64000) [550]?
```

Accessing the Differentiated Services Monitoring Environment

The console portion of the DiffServ feature enables you to view and manage DiffServ-related settings. To access the DiffServ monitoring environment enter **talk 5** at the OPCON prompt (*):

```
* t 5
```

Then, enter the following command at the **+** prompt:

```
+ feature ds
DS Console>
```

Differentiated Services Monitoring Commands

These commands enable you to view DiffServ-related settings. Table 49 summarizes the DiffServ monitoring commands and the rest of this section describes them. Enter the commands at the DS Console> prompt. Either enter the command and options on one line, or enter only the command and then respond to the prompts. To see a list of valid command options, enter the command with a question mark instead of options.

Table 49. DiffServ Monitoring Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Clear	Clears statistics for a stream between a specific ingress and egress interface pair.
DScache	Clears or displays information in a router’s DiffServ cache.
List	Displays information about a router’s DiffServ system and interface-related settings.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Clear

Use the **clear** command to clear statistics for a stream between a specific ingress and egress interface pair.

Syntax: clear stream-stats

Example:

```
DS Console> clear stream-stats
Incoming Network number : 0
Outgoing Network number : 2
IN Net 0 DiffServ not enabled.
Net 0->2 stream stats cleared at sysclock 85327 Second.
```

DScache

Use the **dscache** command to clear or display information in a router’s DiffServ cache.

Syntax: dscache actions
clear
nexthop
order
stats

actions Displays the actions to be taken for packets sent from the specified IP source to the specified IP destination, and the DiffServ stream ID, if any.

Example:

```
DS Console> dscache actions
Source Address to list []?
Destination Address to list []?
Source          Destination     Pro ProtocolInf Net TosIn/Out Action StrmID
10.1.100.1      9.1.140.1         1 T:x08 C:x00   0 x00->x15  PASS   85
9.1.140.1       10.1.100.1        1 T:x00 C:x00   1 x00->x15  PASS   null
```

clear Specifies clearing of the entire DiffServ cache.

nexthop Displays the nexthop IP address.

Example:

```
DS Console> dscache nexthop
Source Address to list []? 5.0.13.248
Destination Address to list []? 5.0.11.249
Source          Destination     Pro ProtocolInf Net  Tos  NextHop
5.0.13.248      5.0.11.249       17  1031> 1031   0  x00  5.0.61.7      (PPP/1)
5.0.13.248      5.0.11.249       17  1032> 1032   0  x00  5.0.61.7      (PPP/1)
5.0.13.248      5.0.11.249       17  1033> 1033   0  x00  5.0.67.1      (PPP/1)
```

order Displays the order in which the packets have arrived.

Example:

```
DS Console> dscache order
Order Source          Destination     Pro ProtocolInf Net  Tos
   1  5.0.16.246      5.0.13.248        1 T:x03 C:x03   2  x00
   2  5.0.13.248      5.0.16.246       17  4000> 5678   0  x00
   3  5.0.16.246      5.0.13.244        1 T:x03 C:x03   1  x00
   4  5.0.13.248      5.0.15.243       17   123>  123   0  x00
```

stats Displays statistics for packets sent from the specified IP source to the specified IP destination.

Example:

```
DS Console> dscache stats
Source Address to list []? 5.0.13.248
Destination Address to list []? 5.0.11.249
Source          Destination     Pro ProtocolInf Net  Tos    RxPkts    RxBytes
5.0.13.248      5.0.11.249       17  1031> 1031   0  x00       432     444096
5.0.13.248      5.0.11.249       17  1032> 1032   0  x00       432     444096
5.0.13.248      5.0.11.249       17  1033> 1033   0  x00       437     459516
```

List

Use the **list** command to display information about a router's DiffServ system and interface-related settings.

Syntax: list interface
queue
stream
vifs

interface Lists the interfaces in a router, their DiffServ enable/disable status, their ingress buffer allocations, and other information.

Net Displays the interface number.

Status
Displays the DiffServ status.

KB/s Displays the link speed in Kb per second.

VirtTime
Displays the virtual time used by the scheduler (indicates n/a for non DiffServ links, indicates 0 if no packets are in progress).

InMax Displays the maximum buffer size configured for assured forwarding.

InCurr Displays the amount of buffer space currently being used for the input stream. The buffers contain packets in progress.

InShar
Displays the amount of shared buffer space available for this egress interface.

InMaxA
Displays the maximum amount of buffer space that can be allocated to all QoS streams in aggregate.

InCurA
Displays the amount of allocated buffer space available for use by the input stream.

NumI Displays the number of input streams.

NumO Displays the number of output streams.

Example:

```
DS Console> list interface
DiffServ interfaces:
Net Status   KB/s  VirtTime InMax  InCurr InShar InMaxA InCurA NumI NumO
--- -------- ----- -------- ------ ------ ------ ------ ------ ---- ----
  0 Disabled  1250      n/a  55000    550  49775  44000   5225   22  n/a
  1 Disabled  1250      n/a  27500      0  27500  22000      0   20  n/a
  2 Enabled    256        0  27500      0  27500  22000      0   20    3
  3 Enabled    256        0  55000      0  55000  44000      0   20    3
  4 Disabled     0      n/a 550000      0 550000 550000      0   20  n/a
  5 Disabled     0      n/a 550000      0 550000 550000      0   20  n/a
  6 Disabled     0      n/a 550000      0 550000 550000      0   20  n/a
  7 Disabled     0      n/a 550000      0 550000 550000      0   20  n/a
  8 Disabled  2000      n/a  27500      0  27500  22000      0   20  n/a
  9 Disabled     0      n/a 550000      0 550000 550000      0   20  n/a
```

queue Displays the weights assigned to the DiffServ egress queues, and the buffer allocation status of the egress interfaces.

Queued packets
Displays the number of packets currently queued (0 indicates that no packets are currently queued).

Svc Tag
Displays the next virtual time that this queue should receive service.

Weight
Displays the scheduler weight of this queue.

out_max_alloc
Displays the maximum amount of buffer space that can be allocated to a DiffServ stream.

out_curr_alloc
Displays the current amount of buffer space allocated.

out_max_buff
Displays the maximum amount of buffer space for this queue.

out_curr_buff
Displays the amount of currently allocated buffer space being used for packets.

out_share_buff
Displays the amount of buffer space currently in the shared pool.

Example:

```
DS Console> list queue
OUT Network number : 1

Premium Queue:
        Queued packets: 0
        Svc Tag:        4294967295
        Weight: 20
        out_max_alloc:  5225 (Bytes)
        out_curr_alloc: 0 (Bytes)
        out_max_buff:   5500 (Bytes)
        out_curr_buff:  0 (Bytes)
        out_share_buff: 5500 (Bytes)


Assured Queue:
        Queued packets: 0
        Svc Tag:        4294967295
        Weight: 80
        out_max_alloc:  22000 (Bytes)
        out_curr_alloc: 4125 (Bytes)
        out_max_buff:   27500 (Bytes)
        out_curr_buff:  0 (Bytes)
        out_share_buff: 23375 (Bytes)
```

stream Displays information about streams.

Id Stream identification number

t Stream type

- **D** DiffServ stream
- **B** Best effort stream
- **C** Network control stream
- **R** RSVP stream

I/o q Queue type

- **q1** Premium queue
- **q2** Assured/BE queue

allo/cur(K)
Total buffer space (in kilobytes) allocated by this stream.

tot pkt
Total packets received for transmission by this stream.

tot Kby
Total kilobytes received for transmission by this stream.

pkt snt
Total packets sent by this stream.

Kby snt
Total kilobytes sent by this stream.

ovr snt
Number of packets sent using shared buffers.

buf drp
Number of packets dropped from this stream because no buffer space was available.

policed
Number of packets dropped by the policer on the premium queue.

Example:

```
DS Console> list stream
Incoming Network number : 0
Outgoing Network number : 2
At interface 0, 22 in-streams; clock=904 sec.
Streams from net 0 to net 2:
 Id   t I/o q  allo/cur(K)  tot pkt  tot Kby  pkt snt  Kby snt  ovr snt  buf drp policed
 ---- - -----  -----------  -------  -------  -------  -------  -------  ------- -------
(policy name)
 85   D   in    5.2/   0.0    82384    43828    48653    25883        0        0
          o-q1   5.2/   1.1                     48653    25883        0        0     732

(-)
 55   B   in    0.0/   0.0        0        0        0        0        0        0
          o-q2   2.8/   0.0                      263       21        0        0       0

(-)
 44   C   in    0.0/   0.0        0        0        0        0        0        0
          o-q2   1.4/   0.0                       79        6        0        0       0
```

vifs Displays information about Frame Relay virtual interfaces.

Example:

```
DS Console> list vifs 1

DiffServ virtual interface for dlci:  17
     Status: Inactive - no packets queued for transmission
     CIR: 64000 (bits/sec)
     Virtual Time: 0
     Service Tag:  0

DiffServ virtual interface for dlci:  16
     Status: Inactive - no packets queued for transmission
     CIR: 64000 (bits/sec)
     Virtual Time: 0
     Service Tag:  0
```

Chapter 22. Using Layer 2 Tunneling (L2TP, PPTP, L2F)

Layer 2 Tunneling (L2T) consists of L2TP, L2F, and PPTP tunneling protocols.

Layer 2 Tunneling Protocol (L2TP) is an IETF standards track protocol for tunneling of PPP across a packet network such as UDP/IP. L2TP is connection oriented.

Layer 2 Forwarding (L2F) and Point to Point Tunneling Protocol (PPTP) are IETF informational protocols for tunneling of PPP across an IP network.

Note: Layer 2 Tunneling is not supported on the 2210 Models 1S4 and 1U4.

Overview of L2TP

L2TP allows many separate and autonomous protocol domains to share a common access infrastructure including modems, Access Servers, and ISDN routers. L2TP permits the tunneling of the PPP link layer, for example, HDLC and asynchronous HDLC. Using these tunnels, it is possible to disassociate the location of the contacted dial-up server from the location that provides access to the network.

Traditionally, dial-up network service on the Internet is provided for registered IP addresses only. L2TP defines a new class of virtual dial-up application that allows multiple protocols and unregistered IP addresses on the Internet. This class of network application is useful for supporting privately addressed IP, IPX, and AppleTalk dial-ups through PPP across an existing Internet infrastructure.

The support of these multiprotocol virtual dial-up applications is beneficial to end users, enterprises, and Internet service providers because it allows the sharing of significant investments in access and core infrastructure and allows end users to use local calls when accessing the services.

L2TP also enables the secure use of existing investments in non-IP protocol applications within the existing Internet infrastructure.

Figure 30 shows a sample L2TP network using ISDN. The network could use any media type between the L2TP Network Access Concentrator (LAC) and the L2TP Network Server (LNS).

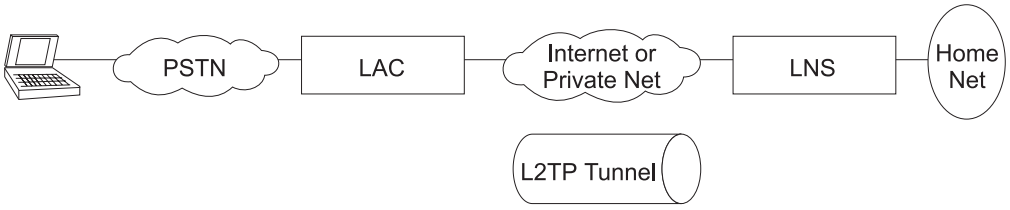


Figure 30. Sample L2TP Network

L2TP Terms

The following terms are used when describing L2TP:

Attribute Value Pair (AVP)
A uniform method of encoding message types and bodies. This method maximizes the extensibility while permitting interoperability of L2TP.

L2TP Access Concentrator (LAC)
A device attached to one or more public service telephone network (PSTN) or ISDN lines capable of handling both PPP operation and the L2TP protocol. The LAC implements the media over which L2TP operates. L2TP passes the traffic to one or more L2TP Network Servers (LNS). L2TP can tunnel any protocol carried by the PPP network.

L2TP Network Server (LNS)
An LNS operates on any platform that can be a PPP end station. The LNS handles the server side of the L2TP protocol. Because L2TP relies only on the single media over which L2TP tunnels arrive, the LNS can have only a single LAN or WAN interface, yet is still able to terminate calls arriving from any PPP interfaces supported by an LAC.

Network Access Server (NAS)
A device providing temporary, on-demand network access to users. This access is point-to-point using PSTN or ISDN lines.

Session (Call)
L2TP creates a session when an end-to-end PPP connection is attempted between a dial user and the LNS. The datagrams for the session are sent over the tunnel between the LAC and LNS. The LNS and LAC maintain the state information for each user attached to an LAC.

Tunnel
A tunnel is defined by an LNS-LAC pair. The tunnel carries PPP datagrams between the LAC and the LNS. A single tunnel can multiplex many sessions. A control connection operating over the same tunnel controls the establishment, release, and maintenance of all sessions and of the tunnel itself.

Supported Features

L2TP runs over UDP/IP and supports the following functions:

- Tunneling of single user dial-in clients.
- Tunneling of small routers, for example a router with a single static route to set up based on an authenticated user's profile.
- Calls can be initiated from the LAC to the LNS (inbound), from the LNS to the LAC (outbound), or by either peer (both). The outbound calls can a *fixed* (always up) or a demand-based L2 tunneling session.
- Multiple calls per tunnel.
- Proxy Authentication for PAP, CHAP and MS-CHAP.
- Proxy LCP.
- LCP restart in the event that Proxy LCP is not used at the LAC.
- Tunnel end-point authentication.
- Hidden AVP for transmitting a proxy PAP password.
- Tunneling using a local rhelm (that is, user@rhelm) lookup table.
- Tunneling using the PPP username lookup in the AAA subsystem.
- Management of L2TP tunnels using SNMP. See "SNMP Management" in the *Protocol Configuration and Monitoring Reference Volume 1*.

Note: Rhelm tunneling requires usernames in *name@rhelm* format. Tunneling this way requires the software to look through two tables to resolve the destination to which the dial-in user is tunneled. The advantage of using this method of tunneling is that you need only define the rhelm and any usernames that match the rhelm will be tunneled to the same destination.

User-based tunneling is resolved in a single table. It allows you the granularity of tunneling each user to a unique destination.

- BRS for an LNS (as a PPP end point).
- The ability to use the **delete interface** command to delete L2TP devices.
- The ability to dynamically reconfigure L2TP devices.
- Establishment of a sequencing, queueing, retransmission and flow control channel. L2TP also performs sequencing, queuing and flow control on data channels.
- The ability to fix the L2TP UDP port (1701) so you can establish IP Security filters based on the UDP port.
- An L2TP router client. L2TP router client is a “client initiated” (also known as voluntary tunneling) model. This function provides secure, tunneled, multi-protocol Virtual Private Network (VPN) services regardless of service provider topology. This function brings the client and LAC into one physical piece of hardware.
- Connection of an inbound call to the appropriate interface based on a remote hostname match. If the remote hostname does not match any of the interfaces configured for hostname matching, the call is completed on an inbound interface that does not use remote hostname matching.

  Note: If you have configured multiple net mappings between the same LAC and LNS pair, make sure only one tunnel exists for each mapping.
- Automatic IP, IPX, and bridging configuration of inbound nets that do not use remote hostname matching. You must manually configure outbound nets and inbound nets that use remote hostname matching.

Other supported Layer 2 Tunneling protocols include:

- L2F-Both NAS and gateway functions are supported.
- PPTP-Router client, PAC (PPTP Access Concentrator), and PNS (PPTP Network Server) are supported.

L2F provides interoperable Layer 2 tunneling when connecting to network devices not supporting L2TP.

PPTP provides interoperable Layer 2 tunneling when connecting to network devices not supporting L2TP. Specifically PPTP can be used for VPN services from Microsoft Windows 95 (DUN 1.2 and higher), Windows 98, and Windows NT to IBM routers.

Note: Both L2F and PPTP are configured in the Layer 2 Tunneling feature.

Timing Considerations

The nature of tunneling PPP packets over routed networks creates some timing issues that you should consider. L2TP assumes that the connection between the LAC and LNS does not have a delay that is long enough to time out the tunneled peers. If the inter-peer latency repeatedly reaches or exceeds that of the PPP state

machine's timeout (usually 3 seconds), then connectivity could be hindered. Note that if the latency between the LAC and LNS is this poor, then connectivity in general is so poor that the connection will be unreasonable even if the PPP state machines were kept alive artificially. If both sides possess the capability, then the PPP timeout may be extended to achieving connectivity over a very poor connection.

Besides latency, a bandwidth mismatch between the LAC/LNS pair and LAC/Client pair may cause problems. For instance, if the actual bandwidth between the LAC and LNS is significantly less than the bandwidth of the PPP client, then the LAC may spend significant time trying to send packets to the LNS. On the other hand, if the connection between the LNS and a host on the LNS home network is exceptionally fast compared with the dial-in client, then the LNS may be overburdened trying to send data to the LAC. L2TP implements a series of internal and external flow control techniques in an attempt to combat these situations.

LCP Considerations

When using Proxy LCP, the LAC negotiates LCP and PPP continues processing at the LNS. The LAC forwards LCP options to the LNS so that the LNS is aware of what was negotiated. The LNS must remain flexible to the parameters negotiated by the client and LAC. If there are any parameters that are unacceptable to the LNS, then L2TP attempts to renegotiate LCP by sending an *LCP Configure Request* to the client across the tunnel.

The requirement for the LNS to remain flexible is of particular concern regarding the MRU. On the IBM LNS, the configured MRU is the maximum allowed for Proxy LCP. If the value in the Proxy LCP message from a LAC is greater than the MRU configured on the LNS, then L2TP will attempt to renegotiate LCP with an MRU equal to the configured MRU without changing other LCP options from the LAC.

Configuring Layer 2 Tunneling

To configure L2T:

1. Access the Layer 2 tunneling feature using the **feature** command.

```
Config> feature layer-2-tunneling
Layer-2-Tunneling config>
```

2. Enable L2TP, L2F, and PPTP as required.

```
Layer-2-Tunneling config> enable L2TP

Layer-2-Tunneling config> enable L2F

Layer-2-Tunneling config> enable pptp
```

3. Add any L2T networks needed. If this is to be strictly an LAC, L2F NAS, or PPTP PAC, you do not have to add any L2T nets. You should define one L2T net for each simultaneous tunneled PPP connection.

```
Layer-2-Tunneling Config>ADD L2-NETS
Additional L2 nets:  [0]? 10
Add unnumbered IP addresses for each L2 net? [Yes]: yes
Adding device as interface 31
Defaulting Data-link protocol to PPP
Adding device as interface 32
Defaulting Data-link protocol to PPP
Adding device as interface 33
Defaulting Data-link protocol to PPP
Adding device as interface 34
Defaulting Data-link protocol to PPP
Adding device as interface 35
Defaulting Data-link protocol to PPP
Adding device as interface 36
```

```
Defaulting Data-link protocol to PPP
Adding device as interface 37
Defaulting Data-link protocol to PPP
Adding device as interface 38
Defaulting Data-link protocol to PPP
Adding device as interface 39
Defaulting Data-link protocol to PPP
Adding device as interface 40
Defaulting Data-link protocol to PPP
```

a. Configure any L2TP, L2F, or PPTP tunnels.

   To configure an L2TP tunnel using an AAA local list:

```
Config>add tunnel-profile
Enter name:  []? lns.org
Tunneling Protocol? (PPTP, L2F, L2TP): L2TP
Enter local hostname: []? lac.org
set shared secret? (Yes, No): [No] Y
Shared secret for tunnel authentication:
Enter again to verify:
Tunnel-Server endpoint address: [0.0.0.0]? 11.0.0.1

      PPP user name: lns.org
      Tunnel Server: 11.0.0.1
      Hostname: lac.org

 User 'lns.org' has been added
 Config>
```

   You can use the previous example to configure tunnel authorization on the LAC as well as “rhelm” tunneling in the form of “user@lns.org.”

   You can set tunnel authentication and authorization to be done at a particular RADIUS server. See “Using Authentication, Authorization, and Accounting (AAA) Security” in *Using and Configuring Features*.

   If you are configuring an LNS and tunnel authentication is disabled on both LAC and LNS, then it is not necessary to configure any tunnel profiles.

   To tunnel by PPP username on a LAC using either an AAA local list or RADIUS:

```
Config>add ppp-user
Enter name:  []? peter
Password:
Enter again to verify:
Allow inbound access for user? (Yes, No):[Yes]
Will 'peter' be tunneled? (Yes, No): [No] Y
Tunneling Protocol (PPTP, L2F, L2TP): [L2TP] L2TP
Enter local hostname: []? lac.org
Tunnel-Server endpoint address: [0.0.0.0]? 11.0.0.1

     PPP user name: peter
     Tunnel Server: 11.0.0.1
     Hostname: lac.org

Is information correct? (Yes, No, Quit): [Yes]

User 'peter' has been added
Config>
```

b. Configure remote hostname matching for the inbound tunnels, if required.

   Note that for client dial-in scenarios, this step is typically not necessary. Use this option when a connection should use a specific net.

   Assuming that the previous configuration was for net 10:

```
Config> net 10
     L2TP 10> set remote-hostname
     Remote Tunnel Hostname: [] ibm.com
```

   Note: To turn off remote hostname matching, use the following commands:

```
Config> net 10
     L2TP 10> set any-remote-hostname
```

4. Configure any L2TP outgoing calls. The following example shows a LAC with IP address 1.1.1.1 and an LNS with IP address 1.1.1.2. The LNS is configured to place a dial-on-demand ISDN call to 5552160 from the LAC.

   LNS Configuration:

```
Config> add tunnel-profile
     Enter name:  []? lac.org
     Tunneling Protocol?  (PPTP, L2F, L2TP): [L2TP]
     Enter local hostname: []? lns.org
     set shared secret? (Yes, No): [No] Y
     Shared secret for tunnel authentication:
     Enter again to verify:
     Tunnel-Server endpoint address: [0.0.0.0]? 1.1.1.1

            Tunnel name: lac.org
               TunnType: L2TP
               Endpoint: 1.1.1.1
               Hostname: lns.org

     User 'lac.org' has been added
     Config>
     Config> add dev layer-2-tunneling
     Config> net 10
     L2TP 10> set connection-direction outbound
     L2TP 10> set idle 30
     L2TP 10> set remote-hostname lac.org
     L2TP 10> enable outbound-call-from-lac
        Outbound Call Type (ISDN, V34)? [ISDN]
        Outbound calling address: 5552160
        Outbound calling subaddress:
     L2TP 10>
     L2TP 10> encapsulator
     PPP 10> set name vickie  a
     L2TP 10>
     L2TP 10> exit
     Config> add ppp-user larry  b
```

   Notes:

   a. Set authentication name in case the LNS device is authenticated. There are additional prompts that are not shown in this example. For details see, “Configuring PPP Authentication” in the chapter “Using Point-to-Point Protocol Interfaces” in the *Software User’s Guide*.

   b. Add users to be authenticated at the LNS. There are additional prompts that are not shown in this example. See Add in the chapter “The CONFIG Process (CONFIG - Talk 6) and Commands” in *Software User’s Guide* for a description of the command syntax and options.

   LAC Configuration:

```
Config> add tunnel-profile
     Enter name:  []? lns.org
     Tunneling Protocol?  (PPTP, L2F, L2TP): [L2TP]
     Enter local hostname: []? lac.org
     set shared secret? (Yes, No): [No] Y
     Shared secret for tunnel authentication:
     Enter again to verify:
     Tunnel-Server endpoint address: [0.0.0.0]? 1.1.1.2

            Tunnel name: lns.org
               TunnType: L2TP
               Endpoint: 1.1.1.1
               Hostname: lac.org
```

```
     User 'lns.org' has been added
     Config>
     Config> add dev dial-in  a
```

Notes:

a. Used to place the physical call.

5. Configure any L2T router clients. The following example shows an L2TP box-to-box connection using the L2TP router client function. This connection is set in one direction and is demand-based.

Client Configuration:

```
Config> add tunnel-profile
     Enter name:  []? lns.org
     Tunnel Protocol? (PPTP, L2T, L2TP): [L2TP]
     Enter local hostname: []? client.org
     set shared secret? (Yes, No): [No] Y
     Shared secret for tunnel authentication:
     Enter again to verify:
     Tunnel-Server endpoint address: [0.0.0.0]? 1.1.1.1

            Tunnel name: lns.org
               TunnType: L2TP
               Endpoint: 1.1.1.1
               Hostname: client.org

     User 'lns.org' has been added
     Config>
     Config> add dev layer-2-tunneling
     Config> net 10
     L2TP 10> set connection-direction outbound
     L2TP 10> set idle 30
     L2TP 10> set remote-hostname lns.org
     L2TP 10> encapsulator
     PPP 10> set name donald   a
     PPP 10> exit
     L2TP 10> exit
     Config>
```

Note: a — Set authentication name in case the client device is authenticated. There are additional prompts that are not shown in this example. For details see, “Configuring PPP Authentication” in the *Software User’s Guide*

.

LNS Configuration:

```
Config> add tunnel-profile
     Enter name:  []? client.org
     Tunneling Protocol? (PPTP, L2F, L2TP): [L2TP]
     Enter local hostname: []? lns.org
     set shared secret? (Yes, No): [No] Y
     Shared secret for tunnel authentication:
     Enter again to verify:
     Tunnel-Server endpoint address: [0.0.0.0]? 1.1.1.2

            Tunnel name: client.org
               TunnType: L2TP
               Endpoint: 1.1.1.2
               Hostname: lns.org

     User 'client.org' has been added
     Config>
     Config> add dev layer-2-tunneling
```

```
Config> net 10
L2TP 10> set connection-direction inbound
L2TP 10> set remote-hostname client.org
L2TP 10> encapsulator
Config>
Config> add ppp-user donald b
Config>
```

Note: **b**— Add users to be authenticated at the LNS. There are additional prompts that are not shown in this example. For details see, "**add** Config command" in the *Software User's Guide* .

6. Configure the various feature L2T parameters using the **set** and **enable** commands, if desired.

```
Layer-2-Tunneling Config>set ?
Layer-2-Tunneling Config>enable ?
```

7. Configure the PPP parameters for all of the L2 nets which are set for inbound and *any* inbound tunnel hostname using the encapsulator command, if desired.

```
Layer-2-Tunneling Config>encapsulator
PPP-L2TP Config>
```

When you have completed the PPP configuration, enter **exit** to return to the L2T feature configuration environment.

Chapter 23. Configuring and Monitoring Layer 2 Tunneling Protocols

This chapter describes the Layer 2 tunneling (L2T) configuration and operational commands. L2T includes Layer 2 Tunneling Protocol (L2TP), Layer 2 Forwarding Protocol (L2F), and Point-to-Point Tunneling Protocol (PPTP). Sections in this chapter include:

Accessing the L2T Interface Configuration Prompt

To access the L2T interface configuration prompt:

1. Enter **talk 6** at the OPCON (*) prompt.
2. Enter **add dev layer-2-tunneling** at the `Config>` prompt (or use the **add l2–nets** command. See "Add" on page 332).
3. Enter **n** *interface#* at the `Config>` prompt.

```
Config> add device layer-2-tunneling
Enter the number of Layer-2-Tunneling interfaces [1]
Adding device as interface 8
Defaulting Data-link protocol to PPP
Config> n 8
Session configuration
L2T config:   8>
```

L2 Tunneling Interface Configuration Commands

Table 50 summarizes the L2T interface configuration commands. Enter these commands at the `L2T Config n>` prompt (where *n* is the net number).

Table 50. L2 Tunneling Interface Configuration Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page xxix.
Disable	Disables outgoing calls.
Enable	Enables outgoing calls.
Encapsulator	Allows you to configure PPP parameters the L2T interface. **Note:** The encapsulator option is only available if an interface has a remote-hostname configured.
List	Displays information about the L2T interface.
Set	Allows you to set various L2T interface parameters.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page xxix.

Disable

Use the **disable** command to disable outbound calls from the L2TP access concentrator (LAC).

Syntax: disable outbound-calls-from-lac

outbound-calls-from-lac
Prevents the LNS from initiating a dial signal from the LAC through an L2TP tunnel.

Enable

Use the **enable** command to enable outbound calls from the L2TP access concentrator (LAC). This command should only be used with L2TP.

Syntax:

enable outbound-calls-from-lac

outbound-calls-from-lac
Allows the LNS to initiate a dial signal from the LAC through an L2TP tunnel.

Example:

```
L2T 10> enable outbound-call-from-lac
        Outbound Call Type (ISDN, V34)? [ISDN]
        Outbound calling address: 1234
        Outbound calling subaddress:
L2T 10>
```

Encapsulator

Use the **encapsulator** command to configure the PPP parameters for the L2T interface.

Syntax: encapsulator

This command is available only when a remote-hostname has been configured. For a list of commands available at the `ppp-L2tp config>`prompt, see “Encapsulator” on page 334.

List

Use the **list** command to display the state of the various L2T interface configuration parameters.

Syntax: list

```
Layer-2-Tunneling Config>list
CONNECTION TYPE
---------- -----
  Connection Direction            INBOUND
  Remote Tunnel Hostname           *ANY*
```

Set

Use the set command to configure the L2T interface operational parameters.

Syntax: set any-remote-hostname

connection-direction

idle

remote-hostname

any-remote-hostname
Clears the outbound remote hostname and disables inbound remote host name matching on this net.

connection-direction [inbound] or [outbound] or [both]
Specifies whether the connection can be initiated by the peer (inbound), the local device (outbound) or either the peer or the local device (both) on this net. If you specify both, you cannot specify zero for the idle time.

Default value: inbound

idle-time *seconds*
Specifies the number of seconds of inactivity after which L2 tunneling will disconnect the tunnel session on this net. A value of zero indicates that the tunnel is fixed and should not be disconnected.

Valid values: 0 to 1024

Default value: 0

remote-hostname *hostname*
Specifies the tunnel hostname of the peer.

For an outbound tunnel, the hostname specifies a tunnel profile configured in the AAA subsystem. This should be the tunnel hostname that the peer uses to identify itself.

For an inbound tunnel, only tunnel peers that identify themselves by this hostname can connect to this interface.

Valid values: Any name from 1 to 64 ASCII characters

Default value: *Name*

Accessing the L2 Tunneling Feature Configuration Prompt

To access the L2 tunneling feature configuration prompt:

1. Enter **talk 6** at the OPCON (*) prompt.
2. Enter **feature layer-2-tunneling** at the `Config>` prompt.

L2 Tunneling Feature Configuration Commands

Table 51 summarizes the L2 tunneling feature configuration commands and the rest of this section explains the commands. Enter these commands at the `Layer-2-Tunneling Config>` prompt.

Table 51. L2 Tunneling Feature Configuration Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Add	Adds L2 tunneling nets and peers.
Disable	Disables L2 tunneling functions.
Enable	Enables L2 tunneling functions.
Encapsulator	Allows you to configure PPP parameters for all of the L2 tunneling nets that are not configured with a remote-hostname (ANY).

Table 51. L2 Tunneling Feature Configuration Commands (continued)

Command	Function
List	Displays information about the L2 tunneling configuration.
Set	Allows you to set buffers, the call receive window, and other L2 tunneling parameters.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Add

Use the **add** command to add L2-Nets. One L2-Net is required for each concurrent PPP session that ends on this router. The end of a tunneled PPP session is the LNS end point of the tunnel.

Syntax: **add**
L2-nets

L2-nets

Note: This command can be entered entirely in lower case. The initial character is shown in upper case for clarity.

Adds L2-Nets to the L2 tunneling configuration. One L2-Net is required for each concurrent PPP session that is to be terminated at this router. If this router is to be used strictly as an LAC, no virtual L2-Nets are necessary. When you enter this command, you are prompted for the number of additional nets and whether to add unnumbered IP addresses for each L2 net.

The number of additional nets refers to how many nets are automatically added at this time. These nets are in addition to any L2-Nets that already exist.

Adding unnumbered IP addresses for each L2-Net automatically adds unnumbered IP entries into the IP routing table for each of the L2-Nets. Unnumbered IP addresses are the preferred mode of operation. If you need numbered addresses for the L2-Nets, you can alter them in the IP protocol configuration environment (refer to the chapter entitled “Configuring IP” in the *Protocol Configuration and Monitoring Reference Volume 1*).

Disable

Use the **disable** command to disable L2 tunneling functions.

Syntax: disable
call-rcv-window
fixed-udp-source-port
force-chap-challenge
hiding-for-pap-attributes
L2f
L2tp
pptp
proxy-auth

proxy-lcp

tunnel-auth

call-rcv-window
L2 tunneling can queue packets for each call in order to perform sequencing and congestion control. Each call has its own window, which is the number of packets that can be sent before an ACK is received. Disabling the *call-rcv-window* turns off flow control and sequencing for all session. This might be desirable when the connection between the LAC and LNS is known to be of high quality, sufficient bandwidth, and not prone to packet reordering.

fixed-udp-source-port
Clears using a fixed UDP port. Disabling this parameter forces you to configure IP Security filters between the LAC and the LNS by IP address.

force-chap-challenge
Disables the LNS CHAP rechallenge of a client. You might need to disable the CHAP rechallenge if the PPP client has difficulty with CHAP rechallenges.

hiding-for-pap-attributes
Disables the encryption of Proxy PAP information between the LAC and LNS.

L2f Disables L2F protocol on this router.

L2tp Disables L2TP protocol on this router.

pptp Disables PPTP protocol on this router.

proxy-auth
Disables sending PPP proxy-authentication from LAC to LNS.

proxy-lcp
Disables sending LCP information from LAC to LNS.

tunnel-auth
Disables tunnel peer authentication based on a shared secret for this router.

Enable

Use the **enable** command to enable L2 tunneling functions.

Syntax:

enable
- fixed-udp-source-port
- force-chap-challenge
- hiding-for-pap-attributes
- L2f
- L2tp
- pptp
- proxy-auth
- proxy-lcp
- tunnel-auth

fixed-udp-source-port
Enabling this parameter allows you to configure IP Security filters by UDP port for L2 tunneling so you can encrypt or authenticate L2 tunneling traffic easily. Sets the UDP port at 1701 for L2TP.

force-chap-challenge
Enables the LNS CHAP rechallenge of a client even if the LNS receives a proxy CHAP. This is preferable from a security standpoint, if it is known that the client can handle such a rechallenge without problems.

hiding-for-pap-attributes
Enables the encryption of Proxy PAP information between the LAC and LNS.

L2f Enables L2F on this router.

L2tp Enables L2TP on this router.

pptp Enables PPTP on this router.

proxy-auth
Enables sending PPP proxy-authentication from LAC to LNS.

proxy-lcp
Enables sending LCP information from LAC to LNS.

tunnel-auth
Enables tunnel peer authentication based on a shared secret for this router.

Encapsulator

Use the **encapsulator** command to access the `ppp-L2tp config>` prompt in order to configure the PPP parameters for all Layer 2 Tunneling interfaces that are configured as inbound and *any* remote-hostname.

Syntax: encapsulator

List

Use the **list** command to display the state of the various L2 tunneling configuration parameters.

Syntax: list

```
Layer-2-Tunneling Config>list
GENERAL ADMINISTRATION
------- --------------
  L2TP                                = Enabled
  L2F                                 = Disabled
  PPTP                                = Disabled
  Maximum number of tunnels           = 20
  Maximum number of calls (total)     = 50
  Buffers Requested                   = 300

CONTROL CHANNEL SETTINGS
------- ------- --------
  Tunnel Auth                         = Enabled
  Tunnel Rcv Window                   = 4
  Retransmit Retries                  = 6
  Local Hostname                      = Host6

DATA CHANNEL SETTINGS
---- ------- --------
  Force CHAP Challenge (extra security)= Disabled
  Hiding for PAP Attributes           = Disabled
  Hardware Error Polling Period (Sec)  = 120
  Call Rcv Window                     = 6
```

```
MISCELLANEOUS
-------------
  SEND PROXY-LCP FROM LAC              = Enabled
  SEND PROXY-AUTH FROM LAC             = Enabled
  Fixed UDP Source Port (1701)         = Disabled
```

Set

Use the set command to configure the L2 tunneling operational parameters.

Syntax: set buffers

call-rcv-window

error-check-direction

host-lookup-password

local-hostname

max-calls

max-tunnels

transmit-retries

tunnel-rcv-window

buffers

Specifies the number of requested internal L2 tunneling buffers. If there is not enough memory to satisfy the request, only a portion of the buffers will be available upon reboot. To confirm the amount of memory while L2T is active, use the **memory** command (see “Memory” on page 340).

Valid values: 1 to 1000

Default value: Depends on model:

Model	Value
12x	100
14x or 24x	150
1Sx or 1Ux	80

call-rcv-window

Specifies the number of packets to be used as a receive window and enables the call-rcv-window. If flow control is enabled on the data channel, a receive window size must be designated, both for use by the protocol on this router and for communication to the peer using start-up messages. The value configured is for all calls initiated by this router. The value of zero means sequence-only (no flow control).

Valid values: 0 to 100

Default value: 0

error-check-period [seconds]

Specifies the LAC’s hardware error polling period. Each polling period will result in a WAN Error Notify message transmitted from LAC to LNS. The range is from 60 to 65000 seconds.

Default value: 120 seconds.

L2 Tunneling Feature Configuration Commands (Talk 6)

host-lookup-password
Specifies the shared secret for RADIUS tunnel authorization. This must match the secret configured on the server.

Default value: None.

local-hostname
Specifies the hostname string identifying the local router that is sent in tunnel setup messages.

Default value: IBM

max-calls
Specifies the maximum number of calls across all tunnels that can be active at a given time either as LAC or LNS.

Valid values: 1 to 500

Default value: Depends on model:

Model	x4x	12x	1Sx/1Ux	
Default	50	40	30	

max-tunnels
Specifies the maximum number of tunnels that can be active at a given time either as LAC or LNS.

Valid values: 1 to 100

Default value: Depends on model:

Model	x4x	12x	1Sx/1Ux	
Default	20	15	10	

transmit-retries
Specifies the number of times an L2TP packet is retransmitted on the control channel before the session or tunnel is declared inactive and is shut down.

Valid values: 2 to 100

Default value: 6

tunnel-rcv-window
Specifies the L2TP receive window size for the reliable control connections transport. This transport transmits and receives the messages necessary for tunnel or session setup, tear down, and maintenance.

Valid values: 1 to 100

Default value: 4

Accessing the L2 Tunneling Monitoring Prompt

To access the L2 tunneling monitoring prompt:

1. Enter **talk 5** at the OPCON (*) prompt.
2. Enter **feature layer-2-tunneling** at the GWCON (+) prompt.

L2 Tunneling Monitoring Commands

This section summarizes and then describes the L2 tunneling monitoring commands. Enter the commands at the `Layer-2-Tunneling Console>` prompt.

Table 52 summarizes the L2 tunneling monitoring commands.

Table 52. L2 Tunneling Monitoring Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Call	Displays statistics and information about each call in progress.
Kill	Ends a tunnel immediately.
Memory	Displays the current L2 tunneling buffer allocation and use.
Start	Starts a tunnel with another peer.
Stop	Stops a tunnel and allows each peer to perform any needed administration.
Tunnel	Displays statistics and information on each existing tunnel.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Call

Use the **call** command to display call statistics and information.

Syntax: call errors
physical-errors
queue
state
statistics

errors Displays the general transmission errors that occurred on the calls.

Example:

```
Layer-2-Tunneling Console> call errors
CallID | Serial # | ACK-timeout | Dropped pkts
 56744 |        1 |           0 |          0
```

CallID The local identifier associated with this call.

Serial #
The number used for logging this call.

ACK-timeout
The number of times a timeout notification has been received from the peer.

Dropped pkts
The number of packets that have been declared lost for this call. These are packets which should have been received, but were signalled as lost by the peer.

physical-errors
Displays the data errors that occurred on the calls.

Example:

```
Layer-2-Tunneling Console> call physical-errors
CallID | Serial# | CRC    |framing|  HW    | buffer|timeout| align-| time since
       |         | Errors| Errors|overrun|overrun| Errors| ment  | updated
 56744 |       1 |     0 |     0 |     0 |     0 |     0 |     0 |
```

CallID The local identifier associated with this call.

Serial #
The number used for logging this call.

CRC Errors
The number of packets on which the CRC did not match.

framing errors
The number of packets with a framing error.

HW overrun
The number of times a hardware overrun occurred.

buffer overrun
The number of times a buffer overrun occurred.

timeout errors
The number of times an interface timed out.

alignment
The number of times an alignment error occurred.

time since updated
The elapsed time since last poll for errors.

queue Displays information about the queue for each call.

Example:

```
Layer-2-Tunneling Console> call queue
CallID | Serial # |Tx Win|Rx Win|   Ns   |   Nr   |Rx Q|Tx Q|priority| out Q
 56744 |       1 |    4 |    4 |    100 |   200  |  0 |  0 |      0 |     0
```

CallID The local identifier associated with this call.

Serial #
The number used for logging this call.

Tx Win
The peer's maximum receive window for data.

Rx Win
The local maximum transmit window.

Ns The next packet sequence number to send for this call.

Nr The next packet sequence number expected to be received for this call.

Rx Q The current number of packets on the receive queue.

Tx Q The current number of packets on the transmit queue.

priority
The number of priority PPP packets waiting to be transmitted by L2TP.

out Q The number of regular PPP packets waiting to be transmitted by L2TP.

state Displays the current state of each call.

Example:

```
Layer-2-Tunneling Console> call state
CallID | Serial # | Net # |    State    | Time Since Chg | PeerID | TunnelID
 56744 |        1 |     2 | Established |    00:00:00    |   345  | 45678
```

CallID The local identifier associated with this call.

Serial #
The number used for logging this call.

Net # The device number associated with this call. For an LNS call, this is the L2-Net. For an LAC call, this is the PPP device that received the initial call.

State The current call state. Valid call states are:

Established
Ready for tunneled network traffic.

Idle The call is idle.

Wait Cs Answer
Waiting for the communication link to open.

Wait Reply
Waiting for a reply from the peer.

Wait Tunnel
Waiting for tunnel establishment.

Time since chg
The elapsed time since the last state change.

PeerID
The Peer's call ID.

TunnelID
The local tunnel associated with this call.

statistics
Displays statistics about the data transmission for each call.

Example:

```
Layer-2-Tunneling Console> call statistics
CallID | Serial # | Tx Pkts | Tx Bytes | Rx Pkts | Rx Bytes |  RTT  | ATO
 56744 |        1 |      34 |     1056 |      45 |     1567 |   10  | 34
```

CallID The local identifier associated with this call.

Serial #
The number used for logging this call.

Tx Pkts
The number of packets transmitted for this call.

Tx Bytes
The number of bytes transmitted for this call.

Rx Pkts
The number of packets received for this call.

Rx Bytes
The number of bytes received for this call.

RTT The currently calculated round trip time for this call.

ATO The currently calculated adaptive time out for this call.

Kill

Use the **kill** to immediately end a tunnel. This command releases all of the local resources for a tunnel thereby forcing the end of the connection. No notification of the end of the tunnel is sent to the peer.

Note: Use this command only if the **stop** command is unable to end a tunnel.

Syntax: kill tunnel *tunnelid*

tunnel *tunnelid*
Specifies the tunnel to end.

Memory

Use the **memory** command to display L2TP's current memory utilization.

Syntax: memory

Example:

```
Layer-2-Tunneling Console> mem
Number of layer-2-tunneling buffers: Requested = 2000, Total = 1200, Free
= 1000
```

In this example, you configured 2000 buffers but were able to allocate only 1200. Currently, 200 buffers are in use leaving 1000 free.

Start

Use the **start** command to start a tunnel with another peer.

Syntax: start (no parameters will prompt for hostname)

tunnel *hostname*

hostname
The name of the host with which L2T establishes the tunnel.

Stop

Use the **stop** command to stop a tunnel. Any required cleanup is completed before the tunnel ends.

Syntax: stop tunnel *tunnelid*

tunnel *tunnelid*
Specifies the tunnel to end.

Tunnel

Use the **tunnel** command to display statistics and information about all tunnels.

Syntax: tunnel

call

errors

peer

queue

state

statistics

transport

calls Displays all tunnels and the call state for each call within each tunnel.

errors Displays the errors that have occurred on a tunnel.

Example:

```
Layer-2-Tunneling Console> tunnel errors
Tunnel ID | Type |ACK-timeouts
 96785    | L2TP |    0
 43690    | PPTP |    2
 96785    | L2F  |    0
```

Tunnel ID
The local identifier associated with a tunnel.

Type The type of tunneling protocol being used.

ACK-timeouts
The number of times a timeout notification has been received from the peer.

peer Displays the tunnels and the peers associated with the tunnels.

Example:

```
Layer-2-Tunneling Console> tunnel peer
Tunnel ID | Type | Peer ID | Peer Hostname
 96785    | L2TP |   89777 | peer1
 11264    | L2F  |   46538 | peer2
 34653    | L2F  |   11209 | peer3
 87511    | PPTP |   55377 | peer4
```

Tunnel ID
The local identifier associated with a tunnel.

Type The type of tunneling protocol being used.

Peer ID
The peer's tunnel identifier assigned to this tunnel.

Peer Hostname
The hostname of the peer as it appears in the local database.

queue Displays information about the queue for each tunnel.

Example:

```
Layer-2-Tunneling Console> tunnel queue
Tunnel ID | Type | Rx Win | Tx Win |   Ns   |   Nr   |  Rx Q  |  Tx Q
 96785    | L2TP |      4 |      4 |      5 |      6 |     0  |     0
 76488    | L2F  |      4 |      4 |      5 |      6 |     0  |     0
 22209    | PPTP |      4 |      4 |      5 |      6 |     0  |     0
```

Tunnel ID
The local identifier associated with a tunnel.

Type The type of tunneling protocol being used.

Rx Win
The local maximum number of packets that constitute the receive window.

Tx Win
The peer's maximum number of packets that constitute the receive window.

Ns The sequence number of the next packet to send.

Nr The sequence number of the next packet to receive.

Rx Q The number of packets currently on the receive queue.

Tx Q The number of packets currently on the transmit queue.

state Displays the current state of all the tunnels.

Example:

```
Layer-2-Tunneling Console> tunnel state
Tunnel ID | Type | Peer ID |    State    | Time Since Chg | # Calls | Flags
 17404    | PPTP |  0      | Established |   00:00:00     |      1  |  0
 96785    | L2TP |  0      | Established |   00:02:05     |      2  |  0
 38237    | L2F  |  0      | Established |   00:00:00     |      1  |  0
```

Tunnel ID
The local identifier associated with a tunnel.

Type The type of tunneling protocol being used.

Peer ID
The peer's tunnel identifier assigned to this tunnel.

State The current tunnel state. Valid tunnel states are:

Established
The tunnel is established.

Idle The tunnel is idle.

Wait Ctrl Reply
The host is waiting for a reply from the peer.

Wait Ctrl Conn
The host is waiting for a connection indication.

Time since chg
The elapsed time since the last state change.

 # Calls
The number of active calls on this tunnel.

Flags The flags used to control the connection messages on this tunnel.

statistics
Displays the statistics associated with the tunnels.

Example:

```
Layer-2-Tunneling Console> tunnel statistics
Tunnel ID | Type | Tx Pkts | Tx Bytes | Rx Pkts | Rx Bytes |  RTT  | ATO
 96785    | L2TP |      4  |      78  |      5  |     89   |   10  | 31
 96366    | L2F  |   9344  |   34578  |    305  |   4300   |   10  | 31
 12344    | PPTP |     24  |     478  |    115  |    2745  |   10  | 31
```

Tunnel ID
The local identifier associated with a tunnel.

Type The type of tunneling protocol being used.

Tx Pkts
The number of packets transmitted.

Tx Bytes
The number of bytes transmitted.

Rx Pkts
The number of packets received.

Rx Bytes
The number of bytes received.

RTT The currently calculated round trip time for tunnel control connection messages.

ATO The currently calculated adaptive timeout for tunnel control connection messages.

transport

Displays UDP information about the tunnels.

Example:

```
Layer-2-Tunneling Console> tunnel transport
Tunnel ID | Type | Peer IP Address | UDP Src | UDP Dest
 96785    | L2TP |  11.0.0.102     |  1056   |   1089
 30000    | L2F  |  11.0.0.104     |  1058   |   1090
 45772    | PPTP |  11.4.4.027     |  1345   |   1020
```

Tunnel ID

The local identifier associated with a tunnel.

Type The type of tunneling protocol being used.

Peer IP address

The peer's IP address for this tunnel.

UDP Src

The UDP source port for this tunnel.

UDP Dest

The UDP destination port for this tunnel.

L2 Tunneling Monitoring Commands (Talk 5)

Chapter 24. Using Network Address Translation

Network Address Translation (NAT) and its extension Network Address and Port Translation (NAPT) can expand the number of IP addresses available to an organization and can prevent users in the public network from becoming aware of some of the addresses in the private network. NAT works by using public IP addresses to represent private IP addresses.

Public IP addresses are the valid addresses of hosts in the IP public network and they must be unique within the public network. If the public network is the Internet, the public IP addresses must be unique Internet addresses provided by the Network Information Center (NIC).

The private addresses are known to the router, but not to the public network. The addresses within each private network must be unique; however, the same address can be duplicated in two different private networks. The private addresses are assigned to hosts within stub networks. Stub networks are networks that have access to the public network through one router only.

NAT expands the number of available IP addresses in several ways:

- It allows each public address to represent multiple private addresses by rotating the use of the public addresses.
- It allows the duplication of addresses as long as each duplicate address is used in a different private network.
- It allows the network administrator to use any IP addresses in the private networks, instead of the NIC addresses that are becoming limited resources.

Using private addresses also hides these addresses from the outside world. This feature of NAT makes it useful as a type of firewall to protect the private addresses from being known.

Important: As stated in section 5.4 of the Internet Draft which defines NAT, “any application that carries (and uses) the IP address (and TCP/UDP port, in the case of NAPT) inside the application will not work through NAT...”. It should be noted that DLSw and XTP make decisions based on the end-point IP addresses — specifically which partner has the higher address. Since the application (such as DLSw or XTP) that is running through NAT thinks that its address is the private address, but the partner application in the other router thinks that the application’s address is the public address, incorrect decisions can be made.

See Figure 31 on page 346 for a drawing of a workstation in a stub network. In this example, the stub network consists of an IP subnet that has the IP address 10.33.96.0 with the subnet mask 255.255.255.0.

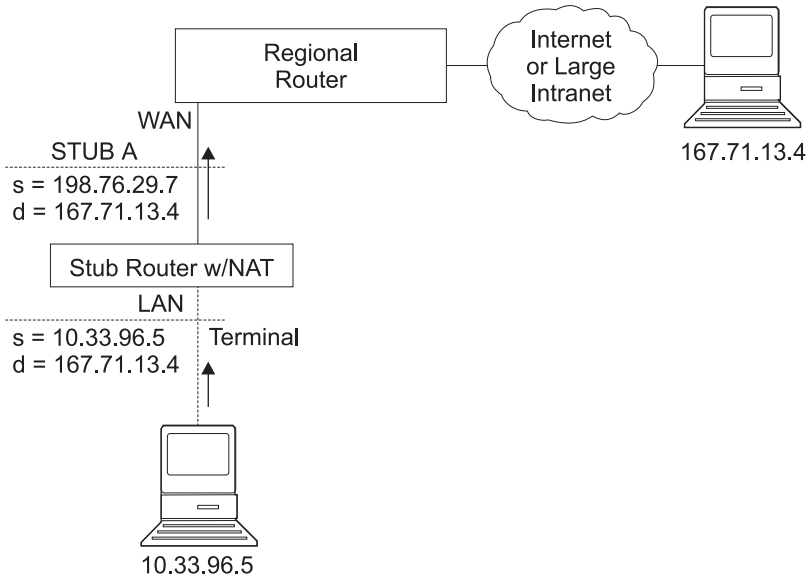


Figure 31. Network Running NAT

To use NAT, the network administrator assigns one or more public IP addresses to a public address pool in the 2210 and assigns a private IP address to each workstation in the stub network. The public IP addresses are assigned to a *reserve pool* and the private IP addresses are assigned to the *translate range*.

The NAT function first binds the private address of a station in the private network to one of the public addresses. Binding means that every packet with that private address will be translated to that public IP address when the packet is outbound. Inbound packets have the public IP address as their destination. NAT recognizes the public address, translates it to the private IP address, and forwards the packet. After traffic stops, the binding is maintained until a timer that you can set times out. At this time, NAT ends the binding and makes the public address available for reuse.

In this example, a packet is transmitted from sending private source address 10.33.96.5 to a destination address in the Internet, 167.71.13.4. NAT in the 2210 translates private address 10.33.96.5 to public address 198.76.29.7. This translation hides the private address 10.33.96.5 from the public network, so that no incoming packet is addressed directly to private address 10.33.96.5. Instead, incoming packets from 167.71.13.4 are addressed to public address 198.76.29.7. When the NAT router receives packets addressed to 198.76.29.7, NAT translates the destination public address to the private address 10.33.96.5 and forwards the packets.

Network Address Port Translation

NAPT can be used only for TCP and UDP traffic. In NAPT, multiple private addresses can use a single public address simultaneously. While NAT maps one public address to one private address, NAPT maps the NAPT public address ***and*** the public port number to a private address and private port number. Only one NAPT address can be configured for each public address pool.

NAPT is configured simply by specifying one public address or a Dynamic-Address interface (which is using PPP/IPCP to retrieve a public address) that will be used for NAPT traffic. The advantage of NAPT is that it can enable one address from the pool of public IP addresses to support many private IP addresses simultaneously.

Static Address Mappings

Sometimes you may want to configure a station or server in the private network that can be directly accessed from the public network. In this case, you should make a static mapping of the private address of the station to a particular public address. All messages outbound from the private address are translated to the designated public address and all messages inbound for the designated public address are automatically forwarded to the associated private address. There are two kinds of static address mappings: NAT and NAPT.

NAT Static Address Mapping

In a NAT mapping, all IP protocols can access the host. This is an example of the configuration of a NAT mapping:

Private address	10.1.1.2
Private port	0
Public NAT address	9.67.1.1
Public port	0

NAPT Static Address Mapping

To specify a TCP or UDP application, you have the option to specify a NAPT mapping that includes a private well-known port. For NAPT static address mapping, a NAPT public address must be configured. For example, to configure a Telnet host at private address 10.1.1.1 to use the NAPT public address 9.67.1.2, the static mapping would be configured as follows:

Private address	10.1.1.1
Private port	23
Public NAPT address	9.67.1.2
Public port	23

The private and public ports are mapped to port 23, which is the well-known port for Telnet. Now, if the administrator also has an FTP server (well-known address 21) at the same private address 10.1.1.1 to map to the NAPT public address 9.67.1.2, that mapping can look like this:

Private address	10.1.1.1
Private port	21
Public NAPT address	9.67.1.2
Public port	21

The server at address 10.1.1.1 has the same NAPT public address (9.67.1.2) for both applications, but NAPT can distinguish between the two by using the different port numbers (23 and 21). However, NAPT cannot distinguish between two servers that use the same NAPT public address and have the same application and port

number. For example, if the NAPT public address and well-known port are the same for 10.1.1.3 port 21 as for 10.1.1.1 port 21, NAPT cannot tell whether to send incoming FTP traffic to server 10.1.1.3 or 10.1.1.1. To configure more than one server with the same NAPT address and application, you must use a port other than the well-known port at the server (for example, start the FTP daemon on port 200).

Setting Packet Filters and Access Control Rules for NAT

In addition to identifying the range of private addresses to be translated by NAT or NAPT, the administrator must set up packet filters and access control rules for IP in the 2210. NAT configuration requires you to configure one inbound and one outbound packet filter on the interface that is connected to the public network. You need to configure one or more access control rules on the inbound packet filter and one or more access control rules on the outbound packet filter. The inbound filter access control rules pass inbound packets with the appropriate defined public addresses to NAT. The outbound filter access control rules pass outbound packets with the appropriate defined private addresses to NAT.

The access control rules that are applied for NAT have the access control rule types ***I*** and ***N*** for inclusive and NAT. Refer to the *Protocol Configuration and Monitoring Reference, Vol. 1* for information about configuring IP access controls.

Note: NAT can also be configured in conjunction with an IPsec tunnel. A sample of this configuration is found in “Configuring Packet Filter Access Control Rules for Router A” on page 291.

Example: Configuration of NAT With IP Filters and Access Control Rules

This example shows how to configure NAT for the stub router in the network pictured in Figure 32 on page 349. See “Chapter 25. Configuring and Monitoring Network Address Translation” on page 353 for descriptions of the commands.

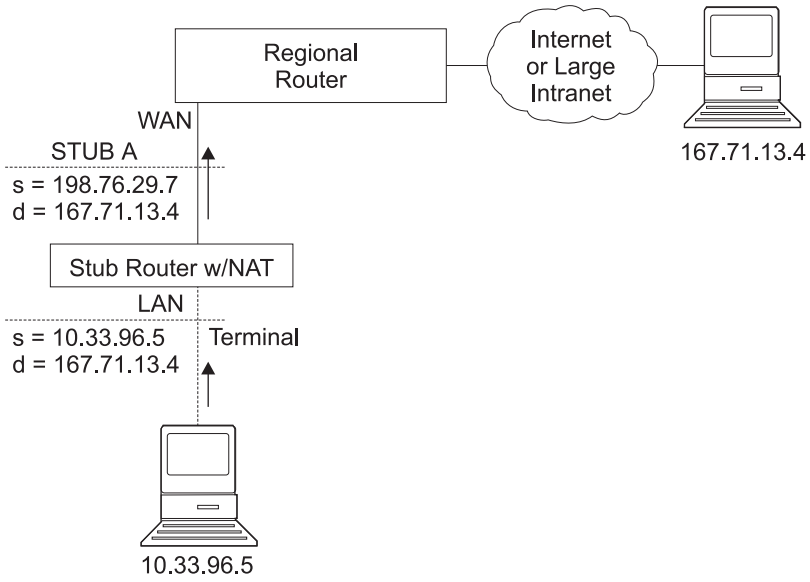


Figure 32. Network Running NAT

Follow this procedure:

1. Set up pools of public addresses for use by NAT and NAPT. To do this, use the **reserve** command.

   ```
   NAT config> reserve No 198.76.29.7 255.255.255.0 6 pool1 198.76.29.7
   NAT config> reserve No 198.76.29.15 255.255.255.0 3 pool1 0.0.0.0
   ```

   In this example, a pool called *pool1* is established. The NAPT address in the pool is 198.76.29.7. The addresses 198.76.29.13 and 198.76.29.14 are not available, so the pool is set up to exclude them. The parameters entered are: *public-address, mask, number-in-group, name, and napt-address*. The value 0.0.0.0 for the NAPT address means that none of the addresses in this group is the NAPT address. Use 0.0.0.0 for the NAPT address in all groups if you do not configure NAPT for the pool.

2. Use the **translate** command to establish the ranges of private addresses to be translated by the public addresses in pool1. The parameters entered are: *private-address, mask, and name*.

   ```
   NAT config> translate 10.33.96.0 255.255.255.0 pool1
   ```

3. Set up static mappings for stations inside the private network that are to be permanently mapped to one of the public addresses. The following commands identify one machine (10.33.96.5) that will receive any type of traffic from the public network. A second machine (10.33.96.4) is both a Telnet and an HTTP server. The parameters are *private-address, private-port-number, public-address, and public-port-number*. Note that the NAPT address for pool1 is used as the public address for the host that is configured with two port numbers.

   ```
   NAT config> map 10.33.96.5 0 198.76.29.8 0
   NAT config> map 10.33.96.4 23 198.76.29.7 23
   NAT config> map 10.33.96.4 80 198.76.29.7 80
   ```

4. Enable NAT.

   ```
   NAT config> enable NAT
   ```

5. Create two IP packet filters so that IP will pass packets to NAT. These are inbound and outbound packet filters for interface 0, which is the interface connected to the public network.

```
IP Config> add packet-filter outbound out-0 0
IP Config> add packet-filter inbound in-0 0
```

6. Use the **update** command to bring up the `packet-filter '`*`filter-name`*`' Config>` prompt. Add an access control rule for NAT to the inbound filter. Packets received over the public interface (net 0) that are destined for an address in NAT's reserved public address pool should be passed to NAT. NAT will replace the public address (and the public port if the packet is destined for the NAPT address) with the correct private address (and the private port if the packet is destined for the NAPT address). The 0.0.0.0 address and mask for the Internet source indicate that any source addresses from the public network will be passed to NAT.

```
IP Config>update packet-filter
Packet-filter name [ ]? in-0
Packet-filter 'in-0' Config> add access
Enter type [E]? IN
Internet source [0.0.0.0]?
Source mask [255.255.255.255]? 0.0.0.0
Internet destination [0.0.0.0]? 198.76.29.0
Destination mask [255.255.255.255]?255.255.255.0
Enter starting protocol number ([0] for all protocols) [0]?
Enable logging? (Yes or [No]):
Packet-filter 'in-0' Config>
```

   The range of addresses in the access control rule is greater than the range of addresses defined in pool1. If the address of the packet passed to NAT is in the range defined in the access control rule but is not one of the ones in the public address pool, NAT passes the packet back to IP unchanged.

7. If you wish the router to pass the packets that do not match the access control rule, rather than drop them, you can create a wildcard access control rule. The following example shows such an access control rule:

```
Packet-filter 'in-0' Config> add access
Enter type [E]? I
Internet source [0.0.0.0]? 0.0.0.0
Source mask [255.255.255.255]? 0.0.0.0
Internet destination [0.0.0.0]? 0.0.0.0
Destination mask [255.255.255.255]?0.0.0.0
Enter starting protocol number ([0] for all protocols) [0]?
Enable logging? (Yes or [No]):
Packet-filter 'in-0' Config>
```

8. Add an access control rule for NAT to the outbound packet filter. Packets to be forwarded from the net 0 interface that have a source address on the private network are identified so that IP can pass them to NAT. NAT replaces the private address with one of the public addresses in pool1.

```
Packet-filter 'out-0' Config> add access
Enter type [E]? IN
Internet source [0.0.0.0]? 10.33.96.0
Source mask [255.255.255.255]? 255.255.255.0
Internet destination [0.0.0.0]?
Destination mask [255.255.255.255]?0.0.0.0
Enter starting protocol number ([0] for all protocols) [0]?
Enable logging? (Yes or [No]):
Packet-filter 'out-0' Config>
```

   With this packet filter as with filter *in-0*, you can add a wildcard inclusive access control rule as the last access control rule if you plan to forward packets that do not match the access control rule.

9. You can use the **list packet-filter** *filter-name* command from the `IP Config>` prompt to check the accuracy and sequence of the access control rules in each packet filter.

10. Enable the access controls for IP.

```
IP Config> set access-control on
```

11. Reset IP and NAT using talk 5. Until now, you have created changes in the router configuration, but these changes have not affected the router. The reset commands for IP and NAT cause the router to read in the new configuration and run with the rules defined in the configuration.

```
NAT> reset NAT
IP> reset IP
```

Chapter 25. Configuring and Monitoring Network Address Translation

This chapter describes the Network Address Translation (NAT) configuring and monitoring commands and includes the following sections:

- "Accessing the Network Address Translation Configuration Environment"
- "Network Address Translation Configuration Commands"
- "Accessing the Network Address Translation Monitoring Environment" on page 360
- "Network Address Translation Monitoring Commands" on page 360

Accessing the Network Address Translation Configuration Environment

To access the NAT configuration environment, enter the following command at the `Config>` prompt:

```
Config> feature nat
Network Address Protocol user configuration
NAT config>
```

Network Address Translation Configuration Commands

This section explains the Network Address Translation (NAT) configuration commands. To configure NAT, enter these commands at the `NAT config>` prompt.

Table 53. NAT Configuration Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page xxix.
Change	Changes public IP address reserve pools, private address translate ranges, and static mappings.
Delete	Deletes public IP address reserve pools, private address translate ranges, and static mappings.
Disable	Disables NAT.
Enable	Enables NAT.
List	Lists information about the NAT configuration.
Map	Creates a static NAT or NAPT binding for a station or server.
Reserve	Creates a public IP address pool and appends addresses to that pool.
Reset	Causes the router to read in the NAT configuration and run according to the NAT rules that have been configured.
Set	Sets timeouts.
Translate	Identifies the private IP addresses to be translated by the NAT public address pool.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page xxix.

Change

Use the **change** command to change public IP address reserve pools, private IP address translate ranges, and static mappings.

Syntax:

change reserve
translate
mappings

reserve *pools*
Provides prompts that enable you to change characteristics of any of the public IP address reserve pools (such as IP addresses and masks) .

Valid Values: An index number to identify the configured pool. This number is displayed when you enter the **list reserve pools** command.

Default Value: none

translate *ranges*
Provides prompts that enable you to change characteristics of any of the private IP address translate ranges (such as IP addresses and masks).

Valid Values: An index number to identify the configured translate range. This number is displayed when you enter the **list translate** command.

Default Value: none

mappings
Provides prompts that enable you to change characteristics of any of the static address mappings (such as IP addresses and ports).

Valid Values: An index number to identify the configured mapping. This number is displayed when you enter the **list mappings** command.

Default Value: none

Delete

Use the **delete** command to delete public IP address reserve pools, private IP address translate ranges, and mappings.

Syntax:

delete reserve
translate
mappings

reserve *pools*
Provides prompts that enable you to delete any of the public IP address reserve pools.

Valid Values: An index number to identify the configured pool. This number is displayed when you enter the **list reserve pools** command.

Default Value: none

translate *ranges*
Provides prompts that enable you to delete any of the private IP address translate ranges.

Valid Values: An index number to identify the configured translate range. This number is displayed when you enter the **list translate** command.

Default Value: none

mappings
Provides prompts that enable you to delete any of the static address mappings.

Valid Values: An index number to identify the configured mapping. This number is displayed when you enter the **list mappings** command.

Default Value: none

Disable

Use the **disable** command to disable NAT. You can disable NAT so that it will drop packets requiring translation or you can disable NAT so that it will pass packets requiring translation.

Syntax:

disable nat

drop

pass

drop Disables NAT so that it drops packets requiring translation.

pass Disables NAT so that it passes packets requiring translation.

Enable

Use the **enable** command to enable NAT. Enabling NAT makes it ready to run, but it will not run until you use the **reset** command or restart the router.

Syntax:

enable nat

List

Use the **list** command to list the public IP address reserve pools, the private IP address translate ranges, the mappings, the global settings, or all the NAT information.

Syntax:

list

reserve

addresses

pools

translate

mappings

global

all

In the following example, times are displayed as hours, minutes, and seconds. Entry age is the time elapsed since the entry was last used. A binding means that traffic is flowing between these two addresses. The timeouts determine how much time will elapse after the last communication before a binding is dropped. See the **set** command for more information about timeouts.

Example:

```
NAT config>list all
NAT Globals:
NAT is ENABLED
Tcp Timeout....:   24:00:00
Non-Tcp Timeout:    0:01:00
NAT Reserved Address Pool(s):
Index First Address    Mask             Count NAPT Address     Pool Name
1     9.8.7.1          255.255.255.0    3     0.0.0.0          pool1
2     9.8.7.6          255.255.255.0    12    9.8.7.9          pool1
NAT Translate Range(s):
Index IP Address       IP Mask          Associated Pool Name
1     7.1.1.0          255.255.255.0    pool1
2     10.0.0.0         255.0.0.0        pool1
NAT Static Mapping(s):
Index Private Address:Port    Public Address.:Port
1     10.1.2.3        0       9.8.7.1        0
2     7.1.1.1         21      9.8.7.9        21
```

Map

Use the **map** command to statically bind a host or server in the private network to a public address. This command, which can be used to set up servers in the private network, establishes an association at NAT startup that never changes.

Static mappings with the public and private port number 0 are NAT mappings; those with other values for the port numbers are NAPT mappings.

Syntax:

map *private-address private-port-number public-address public-port-number*

private-address

The private address of the workstation.

Valid Values: an Internet host address in valid IP format. This should be the address assigned to a station in the stub network that requires permanent access from the public network, such as a server.

Default Value: none

private-port-number

The TCP/UDP port number of the application running in the device with the private address. Entering **0** creates a NAT binding and entering another value creates a NAPT binding. Common port values for NAPT are 23 for Telnet, 21 for FTP, and 80 for HTTP.

Valid Values: 0 - 65535

Default Value: 0

public-address

The public IP address to which this private address is to be mapped. This must be a NAPT address for a NAPT mapping and a NAT address for a NAT mapping.

Valid Values: a valid IP address unique to the public network. The public network can be the Internet or an intranet, depending upon the design of the network.

Default Value: none

public-port-number
The port number of the packets to be translated at the public address. The value 0 represents all ports. Common values are 23 for Telnet, 21 for FTP, and 80 for HTTP.

Valid Values: 0 - 65535

Default Value: 0

In this example, the server with private IP address 10.11.12.200 accepts all traffic from the Internet; the server with private address 10.11.12.199 is a Telnet server and an FTP server.

Example:

```
map 10.11.12.200 0 9.8.7.2 0
map 10.11.12.199 23 9.8.7.9 23
map 10.11.12.199 21 9.8.7.9 21
```

Reserve

Use the **reserve** command to create and append a range of IP addresses to a public address pool. Additionally, it can be used to append a Dynamic IP interface to the public address pool.

Syntax:

reserve *dynamic [interface][public-address][mask][number-in-group] name [napt-address]*

Note: The values shown in brackets are now optionally displayed.

- Dynamic - Specifies if this entry is for a group of public addresses or for a Dynamic-Address interface that will retrieve its IP address from a PPP connection that is using IPCP. Valid values are *yes* or *no*. The default is *no*. If Dynamic=*yes*, then you only need to specify the interface and name. If Dynamic=*no*, you do not specify interface, but you must specify all the other values.
- Interface - Species the Dynamic-Address interface as configured within IP. Any valid interface number can be specified. The default is zero.

public-address
The first public IP address in the sequence of addresses that make up this range or group in the pool. For example, if this group in the pool includes the 12 addresses in sequence from 9.8.7.6 through 9.8.7.17, this value is 9.8.7.6.

Note: To add another range of addresses to the public address pool, use the **reserve** command separately for each group, relating one group to another by using the same pool name. For example, addresses 9.8.7.6 through 9.8.7.17 can be configured in one group within pool1 and addresses 9.8.7.1 through 9.8.7.3 can be configured in another group within the same pool. Then, addresses 9.8.7.4 and 9.8.7.5 are not configured or used by that pool.

Valid Values: a valid IP address that is unique to the public network

Default Value: none

mask A mask to select bits from the IP address. The mask, like an Internet address, is 32 bits long. The 1s in the mask select the network or subnet part of the address. The 0s select the host portion. For example, the address 9.8.7.6 and the mask 255.255.0.0 includes the range of all addresses of which the first two bytes are 9.8 (that is, 9.8.0.0 through 9.8.255.255).

Valid Values: any valid IP mask

Default Value: none

number-in-group
Specifies how many sequential addresses, beginning with the *public-address*, are included in the group. For the addresses 9.8.7.6 through 9.8.7.17, this value is 12.

Valid Values: 1 - the value that can be defined by the IP mask

Default Value: none

name The name of the public address reserve pool. This string has to match the pool name on the corresponding **translate** command.

Valid Values: any name, using up to 16 printable characters; leading and trailing blanks are ignored.

Default Value: none

napt-address
The one IP address from the public address pool that will be used by Network Address Port Translation (NAPT). This address is used for TCP and UDP traffic to map multiple private addresses to the one NAPT address according to the protocol port number. Using NAPT is optional. If it is used, there can be only one NAPT address per public address pool. If there is no NAPT address for a pool or group, enter the value **0.0.0.0**. You need only enter the NAPT address once for the pool.

Valid Values: one of the public IP addresses. It does not necessarily have to be included in the range of values defined in the public address pool, but it must be in the same subnet.

Default Value: 0.0.0.0 (meaning no NAPT)

Example:

```
reserve no 9.8.7.1 255.255.255.0 3  pool1 0.0.0.0
reserve no 9.8.7.6 255.255.255.0 12 pool1 9.8.7.9
reserve yes 2 dynamic_ip_pool
```

Reset

Use the **reset** command to reset NAT. This command deletes all bindings, frees all memory used by NAT, and restarts NAT based on the current Talk 6 configuration. Resetting NAT does not disrupt any other components of the 2210.

Syntax:

reset nat

Note that if NAT encounters an invalid configuration, you will see a message to that effect. Review the NAT ELS messages to see why NAT initialization failed.

Set

Use the **set** command to set TCP and non-TCP timeouts.

Syntax:

set **tcp**

nontcp

tcp *timeout*

The time that NAT maintains a TCP binding after the last message passes between the two bound workstations. A binding is the maintenance of the relationship between a private address and one of the public IP addresses.

Valid Values: 0 - 65535 minutes (0 minutes to about 45 days)

Default Value: 1440 minutes (24 hours)

nontcp *timeout*

The time that NAT maintains a binding that is not TCP after the last message passes between the two bound stations. A binding is the maintenance of the relationship between a private address and one of the public IP addresses.

Valid Values: 0 - 65535 minutes (0 minutes to about 45 days)

Default Value: 1 minute

Translate

Use the **translate** command to add a subnet to the list of addresses that NAT will translate. Each subnet is a translate range. This command must be entered once for each translate range that NAT must know. Any number of translate ranges can use a single public address reserve pool.

Syntax:

translate *private-address mask name*

private-address

Any IP host or subnet address that should be translated.

Valid Values: an address in valid dotted decimal IP format. When ANDed with its subnet mask, this address identifies all addresses in a stub subnet. A stub subnet is a network that accesses the public network only through the router.

Default Value: none

mask **Valid Values**: The network or subnet mask associated with the stub network to be translated.

Default Value: class mask of the private address

name The name of the public address pool NAT should use for this range of private addresses.

Valid Values: any name, using up to 16 printable characters. It must match a public address pool name created by the **reserve** command.

Default Value: none

Accessing the Network Address Translation Monitoring Environment

To access the NAT monitoring environment, type

```
* t 5
```

Then, enter the following command at the **+** prompt:

```
+ feature NAT
NAT>
```

The NAT> prompt appears.

Network Address Translation Monitoring Commands

This section describes the IP Security monitoring commands. Enter these commands at the NAT> prompt.

Table 54. NAT Monitoring Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
List	Lists information about NAT.
Reset	Causes the router to read in the NAT configuration and run according to the NAT access rules that have been configured. NAT does not affect the running of the router until you enter the **reset NAT** command.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

List

Use the **list** command to display information about the NAT configuration.

Syntax:

list all
binding
fragment
global
reserve
pools

addresses

statistics

translate

In the following example, times are displayed as hours, minutes, and seconds. Entry age is the time elapsed since the entry was last used. A binding means that a session is established between these two addresses. The timeouts determine how much time will elapse after the last communication before a binding is dropped. See the **set** command in Talk 6 for more information about timeouts.

Example:

```
NAT>list all
NAT Globals:
Current State    Tcp Timeout    Non-Tcp Timeout    Memory Usage (in bytes)
ENABLED             24:00:00            0:01:00           408

NAT Statistics:
Requests :     Passes      Drops      Holds
       0 :          0          0          0

NAT Address Binding(s):
Private Address//Port  Public Address//Port  Bind Type  Entry Age
        7.1.1.1    21         9.1.1.1    21  STATIC       0:00:13
       10.1.2.3     0         9.1.1.2     0  STATIC       0:00:13

NAT TCP Session Information:
Private Address//Port  Public Address//Port  Tcp State  Data Delta  Entry Age
        7.1.1.1    21         9.1.1.1    21  ESTAB'ED      0        0:00:56

NAT Translate Range(s):
Base Ip Address        Range Mask      Associated Reserve Pool
       7.1.1.0         255.255.255.0   carol
      10.0.0.0         255.0.0.0       carol

NAT Reserve Pool(s):
Reserve Pool     Pool Size    NAPT Address   1st Available Address
carol                21           9.1.1.1          9.1.1.12
-------------------------------------------------------------
Number of Reserve Pools using NAPT.....:     1
Number of configured Reserved Addresses:    21

NAT Fragment Information:
Number of Entries    Number of Saved Fragments
           0                          0
```

Reset

Use the **reset** command to reset NAT. This command deletes all bindings, frees all memory used by NAT, and restarts NAT based on the current Talk 6 configuration. Resetting NAT does not disrupt any other components of the 2210.

Syntax:

reset nat

Chapter 26. Using a Dial-In Access to LANs (DIALs) Server

A DIALs Server allows remote users to dial in to a LAN and access the resources of the LAN in the same manner as if they were locally attached with a LAN adapter. Similarly, the DIALs Server also allows LAN-attached users to dial out to WAN resources (such as bulletin boards, FAX machines, Internet Service Providers (ISP) and other on-line services) eliminating the need for an analog phone line and modem on their workstation.

The DIALs Server can be configured for both dial-in and dial-out users simultaneously. The IBM DIALs Dial-In Client runs on the remote workstation and provides the dial-in function. Figure 33 shows an example of a device used as a DIALs Server supporting the dial-in function.

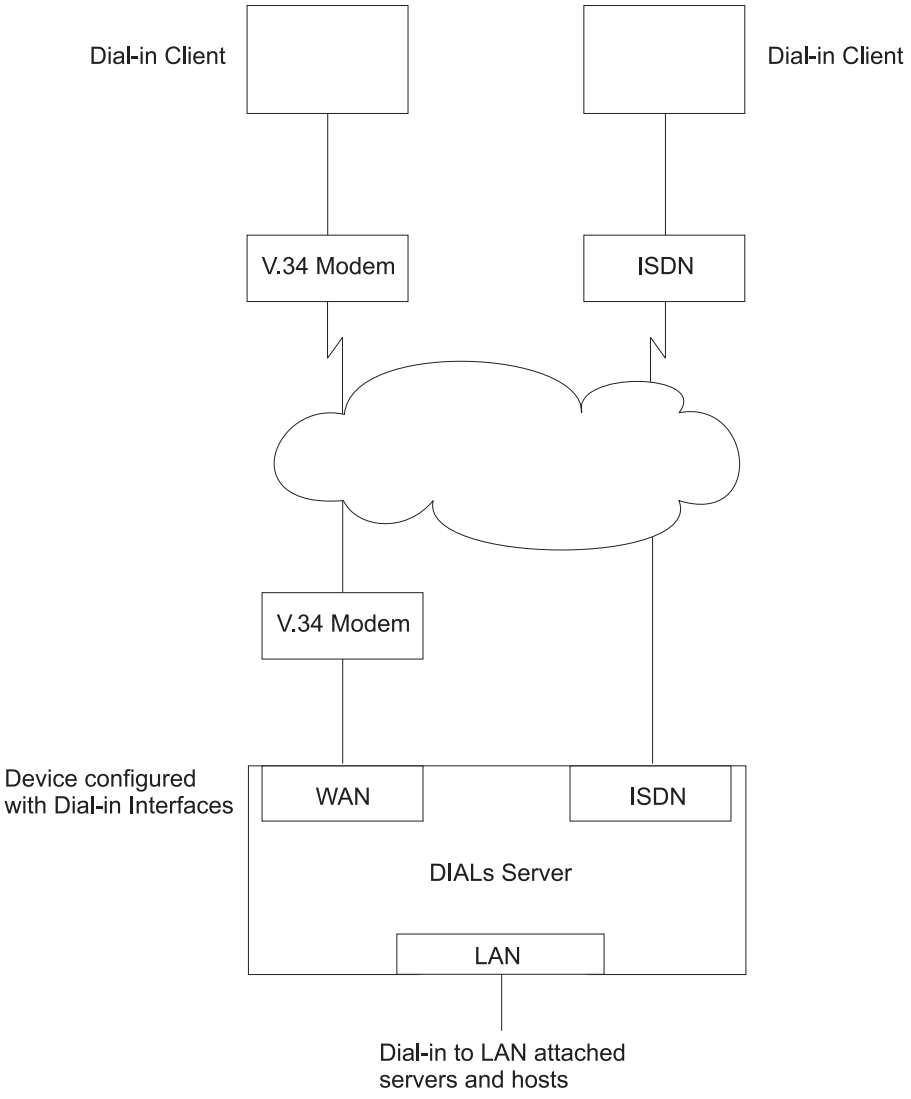


Figure 33. An Example of a DIALs Server Supporting Dial-In

The IBM DIALs Dial-Out Client runs on the network-attached workstation and provides the dial-out function. Figure 34 on page 364 shows an example of a 2210 used as a DIALs Server supporting the dial-out function.

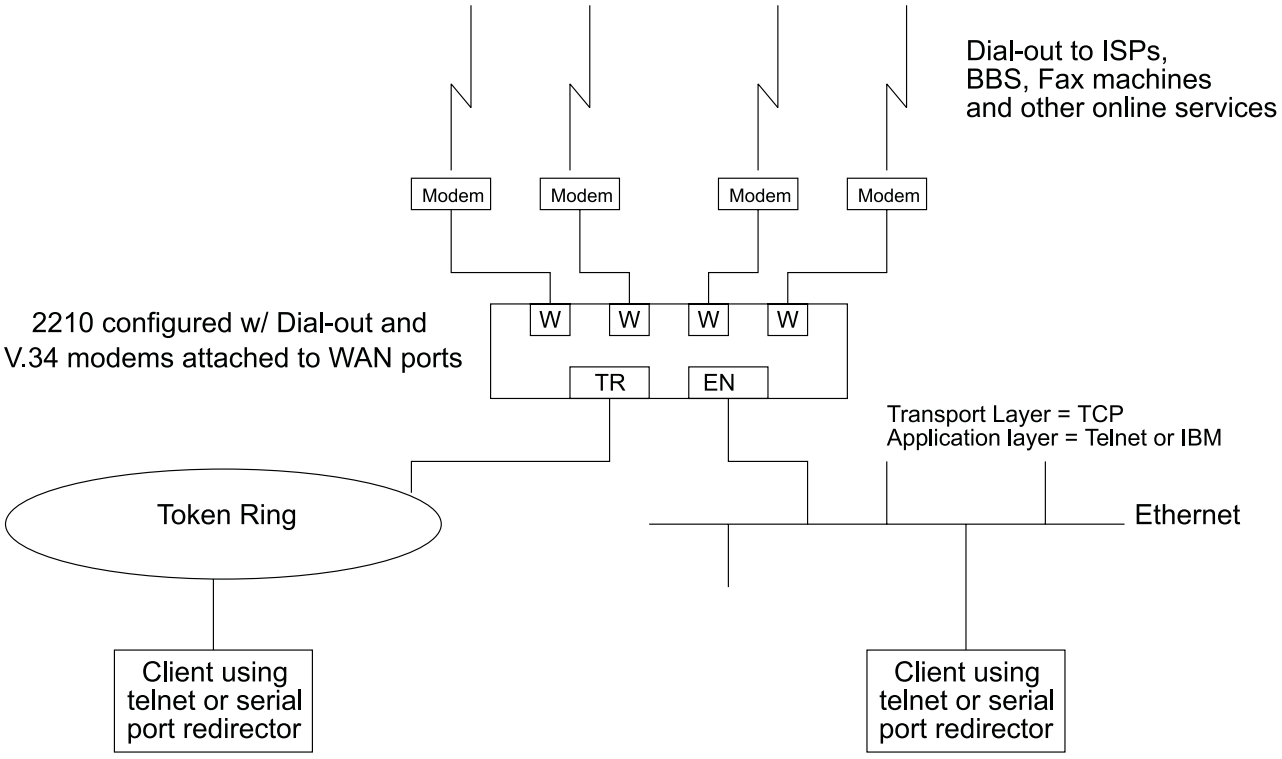


Figure 34. An Example of a DIALs Server Supporting Dial-Out

Before Using Dial-In-Access

Before using Dial-In Access, you need:

- A workstation running the IBM DIALs Dial-In Client or another PPP dial-in client (referred to as the ***dial-in client*** or ***PPP dial-in client*** throughout the following sections).
- Completed protocol configurations on the client machine.
- ISDN interfaces, integrated modem interfaces, a null modem interface, or external V.34 modems connected to the WAN ports of the 2210 that you want to use for single user dial-in.
- A fully configured DIALs Server in your LAN.

Configuring Dial-In Access

This section describes how to configure both dial-in and dial-out functions on the DIALs Server. Configuring a client to use dial-in access is described in the documentation associated with the client the workstation uses.

Configuring Dial-In Interfaces

Dial-in interfaces on the 2210 are a special type of dial-circuit. Because most of the settings for a typical dial-circuit are not relevant for single-user dial-in applications, a new device type called ***dial-in*** can be added that sets appropriate defaults for the dial-circuit. Adding a dial-in device also sets up the PPP encapsulator configuration defaults that work with the majority of PPP dial-in clients, including the IBM DIALs

Dial-In client. These defaults are described in “Dial Circuit Parameter Defaults for Dial-In Interfaces” and “Dial Circuit PPP Encapsulator Parameters for Dial-In Circuits”.

Note: DIALs function can only be enabled on dial-in circuits. Dial-in circuits are only supported when the base net is a V.34 or a ISDN net.

Dial Circuit Parameter Defaults for Dial-In Interfaces

Notes:

1. Do not override the parameters described in this section. Doing so will prevent the dial-in function from operating correctly.
2. Some parameters may not be displayed or configurable. For a complete description of the parameters, see “Configuring and Monitoring Dial Circuits” in the *Software User’s Guide*.

The following defaults are set when you add a dial-in interface:

- ***Idle time*** is set to 0. Note that a standard circuit is defined as a circuit where the idle timer has no meaning. It will not be a fixed circuit to automatically dial-out. The only time the circuit will dial-out is if a PPP callback has been negotiated or if Multilink PPP has been enabled on this circuit. See “Shiva Password Authentication Protocol (SPAP)” and “Using the Multilink PPP Protocol” in the *Software User’s Guide* .
- ***Inbound calls*** are allowed. Any inbound is setup because PPP dial-in clients do not use the LID exchange implemented by Nways dial-circuits.
- ***Outbound calls*** are allowed.

  Note: “Outbound” for a dial-in circuit is not the same as a dial-out circuit. See “Before Configuring Dial-Out Interfaces” on page 366.

- A default destination address is set up for “default_address” This address is added to either the list of V.34 addresses or ISDN addresses. Because these calls are inbound and the only outbound calls will be the result of either a callback or a multilink PPP exchange, the destination address is meaningless. However the address is required for the circuit parameters. Do not delete this address or your circuits will come up disabled.

Dial Circuit PPP Encapsulator Parameters for Dial-In Circuits

Note: For a complete description of the following parameters see “Using Point-to-Point Protocol Interfaces” in *Software User’s Guide*.

The following defaults are set when you add a dial-in interface:

- Authentication is enabled for SPAP, CHAP, and PAP.
- The PPP MRU is set to 1522. This MRU size is needed for the Windows 3.1, OS/2, and DOS versions of the IBM DIALs Dial-In clients. Do not change this setting unless you know you are not using these clients.
- Automatically enables DIALs on the PPP encapsulator. This turns on some of the features important for Dial-In Access to LANs users such as the NetBIOS Control protocol, NetBIOS Frame Control protocol, time remaining, SPAP authentication, callback, LCP identification, and automatic addition and deletion of IP static routes to the client. See “Using Point-to-Point Protocol Interfaces” in *Software User’s Guide* for more information on the DIALs features.

Adding a Dial-In Interface

To add a dial-in interface:

1. Configure a V.34 or ISDN base net on one of the available WAN interfaces of the 2210. See “Using the V.34 Network Interface” and “Using the ISDN Interface” in the *Software User’s Guide* for configuration details.
2. Enter **talk 6** to access the `Config >` prompt.
3. Enter **add device dial-in** at the `Config >` prompt to add the dial-in interface. You will be asked how many dial-in circuits to add. This command will create the new nets, report their net numbers, prompt for the base net number and prompt to enable for Multilink PPP.

   Example: Assume the current maximum net is 3 and you want to add 1 dial-in net to the base 2 net.

Figure 35 is an example of defining a dial-in interface.

Figure 35. Adding a Dial-In Interface

```
Config>add dev dial-in
Adding device as interface 4
Defaulting Data-link protocol to PPP
Use "net 4" command to configure circuit parameters
Base net for this circuit [0]? 2

Enable as a Multilink PPP link? [no]

Disabled as a Multilink PPP link.

Use "set data-link" command to change the data-link  protocol
Use "net " command to configure dial circuit parameters.
Config>li dev
Ifc 0     Ethernet               CSR  81600, CSR2  80C00, vector 94
Ifc 1     V.34 Base Net          CSR  81620, CSR2  80D00, vector 93
Ifc 2     V.34 Base Net          CSR  81640, CSR2  80E00, vector 92
Ifc 3     PPP Dial-in Circuit
Ifc 4     PPP Dial-in Circuit
```

Before Configuring Dial-Out Interfaces

Before configuring and using dial-out interfaces on the 2210, you need:

- IBM Nways software with DIALs support loaded on a 2210.
- An external V.34 modem, an integrated modem, or a null modem, , or an ISDN interface if connecting to an available WAN port on the 2210. See “Using the V.34 Network Interface” in the *Software User’s Guide* for configuration information.
- A workstation connected to the LAN that has access to the 2210 DIALs Server.
- Software on the client such as telnet, a telnet redirector or the IBM DIALs Dial-Out clients. IP must be correctly configured on the client in order for the dial-out client to work.

Null Modem Usage

When using a null modem, use D25NM-3 full handshake:

Pin mapping:

1 to 1	**1 to 1**
2 to 3	**3 to 2**

4 to 5	**5 to 4**
6 to 8, 20	**8, 20 to 6**
7 to 7	**7 to 7**

Configuring Dial-Out Interfaces

The following steps describe how to configure a dial-out interface on your device.

1. Connect a V.34 modem to the WAN port that you will use as a dial-out interface.
2. Connect to the console of the 2210 DIALs Server.
3. Enter **talk 6** at the * prompt.
4. Set up a V.34 interface. See "Using the V.34 Network Interface" in the *Software User's Guide* for details.
5. Add a dial-out interface using the **add device dial-out** command. When prompted for the interface, use an available V.34 interface number.

   Notes:

   a. Multiple circuits can be configured on top of a V.34 base net. However, only one circuit can be active at any given time.

   b. The software defines a V.34 address called ***default_address***. Do not delete this address as it is required by dial-out and dial-out will not work without it.
6. Configure the PPP authentication server, if you are using the IBM DIALs Dial-Out client, and add PPP users as described in "PPP Authentication Protocols" in the *Software User's Guide*. The added PPP users should have dial-out enabled. Dialing out using telnet does not require authentication, therefore do not configure authentication for telnet sessions.
7. Configure the global dial-out parameters using the **feature dials** command. See the **feature** command in the *Software User's Guide*.

   In this environment you can configure the dial-out inactivity timer, the dial-out server name, modem pools, and other parameters.
8. For the IBM DIALs Dial-Out client to work correctly, a SNMP community must be defined with read access granted to all dial-out clients that should be able to use the dial-out server. This is required for the dial-out chooser application to be able to discover dial-out servers on the network. Refer to "SNMP Management" in the *Protocol Configuration and Monitoring Reference Volume 1* for information about how to configure a SNMP community.
9. Restart the device.

Configuring Modem Pools

Modem pools are defined as a group of modems which appear to the user as one modem. When the user needs to dial-out, the first available modem in this pool is used. Modem pools are created in the 2210 DIALs Server by defining groups of dial-out interfaces with the same portname. By default, all dial-out interfaces are named "ALL_PORTS" which creates a modem pool. Naming the dial-out interfaces individually enables a user to select a particular modem to dial-out.

To configure a modem pool:

1. Enter **talk 6** at the * prompt.
2. Enter **net *n***, where ***n*** is the number of the dial-out interface as defined in "Using the V.34 Network Interface" in the *Software User's Guide*. This action places you in the configuration environment for the interface.

3. Enter **encapsulator** (see “Configuring and Monitoring Dial Circuits” in the *Software User’s Guide*) at the `Circuit Config>` prompt. This action places you in the dial-out configuration environment.
4. Enter **set portname** at the `Dial-out Config>` prompt. This action will prompt you for the name of the port (up to 30 characters). If you specify an existing port name, the modem is added to the pool with that name.
5. Restart the 2210.

Before Configuring Global DIALs Parameters

This section describes the global DIALs Server parameters.

Server Provided IP Addresses

The router can be configured to provide an IP address for a dial-in client to use for the duration of its connection. The address the router will assign to the client can be retrieved by 4 different methods. These methods, in order of priority are listed below:

1. User ID

   An IP address can be stored in the PPP user profile for each client. When a client connects and requests an IP address, the router retrieves the address configured in that user’s PPP user profile. This allows the user to get the same IP address each time, but requires a unique IP address for every user.

   Use the `Config>` **add ppp-user** command to configure an IP address in the PPP user profile.

2. Interface

   An IP address can be stored in the dial-in interface configuration. When a client connects and request an IP address, the router retrieves the address from the interface through which the connection was made. This method requires a unique IP address for each dial-in interface.

   To set the interface IP address:

   - Use the `Config>` **list devices** command to display the interface number assigned to the hardware interface.
   - Use the `Config>` **net ’x’** command, where ’x’ is the configured interface number, to access the command prompt for the interface.
   - Use the `PPP Config>` **set ipcp** command to set the interface IP address.

3. Pool

   Blocks of IP addresses can be stored in a IP address pool. When a client connects and requests an address, the router retrieves an address from the pool. When the client disconnects, the address is returned to the pool. This method provides a single location for configuring dial-in client’s IP address without the need for an address server.

   Use the `DIALs config>` **add ip-pool** command to add a pool of IP addresses.

4. DHCP Proxy

   An IP address can be leased from a DHCP server. When a client connects and requests an address, the router requests an address from the DHCP server on behalf of the client. This method requires a DHCP server be present on the LAN or configured in the router. One DHCP server can provide addresses for clients on multiple routers. See “Dynamic Host Configuration Protocol (DHCP)” on page 369 for more information.

   Use the `DIALs config>` **add dhcp-server** command to add a DHCP server.

IP Address Assignment Methods

The IP address used by a dial-in client for the duration of the connection may come from 5 different sources. These sources are listed in order of precedence:

1. client provided
2. user id assigned
3. interface assigned
4. address pool
5. DHCP server

When a dial-in client connects, the router steps through these sources until it finds an address or exhausts all sources. If no IP address can be found, IPCP negotiation fails. Any combination of methods may be used.

The default configuration is:

```
Client     :  Enabled
UserID     :  Enabled
Interface  :  Enabled
Pool       :  Enabled
DHCP Proxy :  Disabled
```

Note: There are no addresses configured by default in the PPP user profile, the interface or the IP address pool.

Dynamic Host Configuration Protocol (DHCP)

The Dynamic Host Configuration Protocol (DHCP) was developed to provide configuration parameters to hosts on a network. Among other configuration parameters, DHCP has a mechanism for allocation of network addresses to hosts.

The Proxy DHCP feature acts as a client *on behalf* of a dial-in PPP user. This allows the device to obtain an IP address lease for the duration of the dial-in session, or until the lease expires. The IP address that is allocated from the DHCP server is communicated to the dial-in client through PPP IPCP (see “IP Control Protocol” in the *Software User’s Guide* for a description of IPCP). The dial-in client software has no knowledge that DHCP was used to allocate an IP address, and thus requires no DHCP activation of any kind.

Proxy DHCP requires that at least one DHCP server be configured and accessible from the router.

Proxy DHCP requires that the addresses being allocated to dial-in users be within the same subnet of a directly connected LAN. In a typical configuration, this requires enabling proxy ARP subnet routing to allow the router to answer ARP requests to hosts on the local network on behalf of the dial-in clients.

Basic DHCP Setup

The most basic configuration calls for a single DHCP server on the same network as the router, with dial-in addresses to be leased within the same subnet as this LAN.

When the client dials in, a lease for an IP address is obtained from the DHCP server and used in IPCP negotiation with the client.

1. Connect 2210 and DHCP to the same LAN.

2. Configure and start the DHCP server (see your DHCP server's documentation for how to setup your server to lease IP addresses. Remember, the IP addresses to be leased MUST be within a subnet of a directly connected LAN and proxy ARP must be enabled on the 2210).
3. The typical setup for Proxy DHCP disables Client-Specified, Userid, and Interface and Pool IP Address Negotiation options:

```
Dials Config>list ip
DIALs client IP address specification:
Client : disabled
UserID : disabled
Interface : disabled
DHCP Proxy : enabled
```

4. Add DHCP server (`Dials Config>` **add dhcp 10.0.0.111**)
5. Set dial-in client software to *Server assigned*.

   Notes:

   a. *Server assigned* configuration varies among different dial-in client implementations.

   b. The client software should not be configured to obtain its address from DHCP. The client should obtain its address by sending an address of 0.0.0.0 to IPCP on the initial configure request.
6. For this setup, let the DHCP GATEWAY ADDRESS default to 0.0.0.0.

Multiple Hops to DHCP Server

The configured DHCP server(s) should be IP addresses which are reachable from the connected router. You should always be able to ping the server from the remote access box.

When the DHCP server is located multiple hops away, the server needs to know an address to reply to, and to indicate which pool to allocate an IP address from. The pool to allocate an IP from is important because the DHCP server could be utilized to serve addresses to a number of subnets and there must be some indication as to which pool of addresses to select from. The DHCP Gateway Address (*giaddr*) is used for this (the terminology is based on the definition given in RFC 2131). The *giaddr* must be an address that is local to the 2210, such as the token ring or Ethernet LAN port. Also, since the *giaddr* is the address which the DHCP server will use to reply, make sure you can ping this address from the DHCP server itself.

Multiple DHCP Servers Network

You can configure multiple DHCP servers for redundancy. When you configure multiple servers, the Proxy DHCP client asks all servers for an address and accepts the first response received. If any of the DHCP servers are more than one hop away, or are connected to a subnet which is not associated with the addresses in its pool, then *giaddr* must be configured. See "Multiple Hops to DHCP Server".

While there can be more than one DHCP server offering addresses, it is important to not allow the pool of addresses configured at each server to overlap. Further, because there is only one *giaddr* for the DHCP server to respond to and perform a lookup with, each pool of address must be in the same subnet as each other.

Dynamic Domain Name Server (DDNS)

A Domain Name Server (DNS) maps IP addresses to hostnames and is typically static in nature. Dynamic DNS is a feature that, when used with a DDNS DHCP

server and a DNS server, enables DHCP to dynamically update the DNS server with an IP address and hostname mapping. This feature may only be used in conjunction with Proxy DHCP.

When you enable Dynamic DNS on the 2210 and you configure a hostname in the user profile (see “PPP Authentication Protocols” in the *Software User’s Guide*), this hostname is passed as option 81 (DDNS) to the DHCP SERVER. If you configured the DHCP server correctly for DDNS, the DHCP server updates the DDNS server with the IP address that it leased to the router and the hostname that the router sent to it. This allows other users to access the dial-in client through the hostname rather than requiring the client to know the dynamically chosen IP address.

Chapter 27. Configuring DIALs

This chapter describes DIALs configuration and operational commands. The chapter includes:

Accessing the DIALs Global Configuration Environment

Use the following procedure to access the global configuration process.

1. At the OPCON prompt, enter **talk 6**. (For more detail on this command, refer to *The OPCON Process and Commands* in the Software User's Guide.) For example:

   ```
   * talk 6
   Config>
   ```

   After you enter the **talk 6** command, the CONFIG prompt (`Config>`) displays on the terminal. If the prompt does not appear when you first enter configuration, press **Return** again.
2. At the CONFIG prompt, enter the **feature dials** command to get to the `DIALs Config>` prompt and access the DIALs global parameter configuration environment.

DIALs Global Configuration Commands

Table 55. DIALs Global Configuration Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page xxix.
Add	Adds a (Dynamic Host Configuration Protocol) DHCP server to the list of DHCP servers or adds an IP address pool.
Delete	Deletes a DHCP server from the list or removes a block of addresses from an IP address pool
Disable	Disables IP address assignment methods, dial-out protocols, multi-chassis MP, SPAP Banner, and Dynamic DNS.
Enable	Enables various methods of IP address assignments, dial-out protocols, multi-chassis MP, SPAP Banner, and Dynamic DNS.
List	Lists the Global DIALs parameters and their values.
Set	Sets time-allowed, dhcp gateway address, NetBIOS Name Server addresses, locally assigned MAC addresses, Virtual Connections (VC) Dynamic Name Server addresses, dial-out inactivity timer, and dial-out server-name.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page xxix.

Add

Use the **add** command to add a new Proxy DHCP server to a list of servers or to add an IP pool of addresses.

The proxy DHCP server list contains the IP addresses of the DHCP servers that will, in turn, lease IP addresses to the dial-in clients. Multiple servers may be added for redundancy. The maximum number of servers is 20.

The IP address pool feature provides a method by which the router may retrieve an IP address from a locally defined pool of addresses to a dial-in client. The client may use this address for the duration of the connection to the router. A pool consists of one or more blocks of IP addresses. The maximum number of blocks is 20. Each of these blocks is defined by a base IP address and the number of addresses in the block. The addresses in each block are ascending and contiguous, starting with the base address.

Syntax:

add dhcp-server *ipaddress*

ip-pool *baseaddress #addresses*

dhcp-server *ipaddress*

Adds a dhcp-server with the specified IP address.

Example :

```
DIALs Config> add dhcp-server
DIALs Proxy DHCP server address [0.0.0.0]? 10.0.0.1
```

ip-pool *baseaddress #addresses*

Add a block of addresses to the IP pool.

Example:

```
DIALs Config> add ip-pool
Base address []? 192.1.100.18
Number of addresses [1]? 57
DIALs config>add ip-pool
Base address []? 192.2.200.1
Number of addresses [1]? 250
DIALs config>list ip-pools
     Configured IP address pools:
          Base Address      Last Address        Number
          ------------      ------------        ------
          192.1.100.18      192.1.100.74            57
           192.2.200.1     192.2.200.250           250
```

Delete

Use the **delete** command to delete an existing Proxy DHCP server from the list of servers or to remove a block of addresses from the IP address pool.

Syntax:

delete dhcp-server *ip address*

ip-pool *baseaddress #addresses*

dhcp-server *ipaddress*

Removes a dhcp-server with the specified IP address.

Example:

```
DIALs Config> delete dhcp-server
Enter the address to be deleted [0.0.0.0]? 10.0.0.1
```

ip-pool *baseaddress #addresses*
Removes a block of addresses from the IP pool.

Example:

```
DIALs Config> delete ip-pool
Base IP address of the block to be removed []? 192.2.200.1
```

Disable

Use the **disable** command to disable an IP address assignment method, dial-out protocols, SPAP Banner, and Dynamic DNS.

Syntax:

disable dynamic-dns
dial-out
ip-address-assignment *type*
spap-banner

dial-out ***type***
Disables the use of dial-out with either telnet or IBM DIALs Dial-Out clients. You can specify:

dials Disables all IBM DIALs Dial-Out clients

telnet Disables all telnet clients.

To disable both types of clients you must enter the disable dial-out command for each type. Disabling both types of clients disables dial-out on the 2210.

dynamic-dns
Disables the sending of DHCP option 81 for the user's hostname. See "Dynamic Domain Name Server (DDNS)" on page 370 for more information.

IP-address-assignment ***type***
Disables various IPCP address assignment techniques. You may specify any of the following:

- Client – Prevents client-assigned IP address assignment.
- Userid – Prevents using the authenticated user profile for an IP address.
- Interface – Prevents the router from using the IPCP settings for the interface.
- Pool – Prevents the router from using the IP address pool to assign addresses to clients.
- DHCP-proxy – Prevents the router from leasing an address from the DHCP server.

See "Server Provided IP Addresses" on page 368 for additional information about assignment techniques.

spap-banner
Disables the sending of a SPAP banner to a remote user authenticated with SPAP.

Note: Entering a \n will force a new line character in the banner displayed at the client.

Enable

Use the **enable** command to enable IP address assignment, dial-out protocols, SPAP Banner, and Dynamic DNS.

Syntax:

enable dynamic-dns
ip-address-assignment . . .
spap-banner

dial-out *type*
Enables the use of dial-out with either telnet or IBM DIALs Dial-Out clients. By default, both types of clients are enabled. You can specify:

dials Enables all IBM DIALs Dial-Out clients

telnet Enables all telnet clients.

dynamic-dns
Disables sending of DHCP option 81 for the user's hostname. See "Dynamic Domain Name Server (DDNS)" on page 370 for more information.

IP-address-assignment *type*
Enables various IPCP address assignment techniques. The router will attempt each method enabled in the order listed. You may specify any of the following:

- Client – Allows the client to specify the address it wants to use.
- Userid – The router will look in the authenticated PPP user profile for an IP address. If the address is nonzero, it will be offered to the client.
- Interface – The router will look at the IP address configured on the interface. If the address is nonzero, it will be offered to the client.
- Pool – The router will request an address from the IP address pool. If an address is available, it will be offered to the client.
- DHCP-proxy – The router will attempt to lease an address from DHCP. If successful, the address will be offered to the client.

See "Server Provided IP Addresses" on page 368 for additional information about assignment techniques.

spap-banner
Enables the sending of a SPAP banner to a remote user authenticated with SPAP. Use the **set spap-banner** command described on "Set" on page 378 to enter the text of the SPAP banner. Refer to "Shiva Password Authentication Protocol (SPAP)" in the *Software User's Guide* for more information.

List

Use the **list** command to display the current configuration. The DHCP state and lease times can be monitored for each net from the Point-to-Point console. See the **listipcp** command in the *Software User's Guide* for an example.

Syntax:

list all
dhcp-servers

dial out

dynamic-dns

ip-address-assignment

ip-pools

name-servers

spap-banner

time-allowed

vc-parameters

Example:

```
DIALs config>li all
DIALs client IP address assignment:
Client     :  Enabled
UserID     :  Enabled
Interface  :  Enabled
Pool       :  Enabled
DHCP Proxy :  Disabled


Configured IP address pools:
     Base Address      Last Address        Number
     ------------      ------------        ------
       11.0.0.100        11.0.0.129            30
       11.0.0.210        11.0.0.229            20

Configured DHCP servers:           11.0.0.2         11.0.0.50
Proxy DHCP is currently disabled
DHCP gateway address (giaddr): 11.0.0.10

Dynamic DNS: Enabled

Primary Domain Name Server     (DNS): 11.0.0.2
Secondary Domain Name Server   (DNS): None
Primary NetBIOS Name Server   (NBNS): 11.0.0.2
Secondary NetBIOS Name Server (NBNS): None

Time allowed for connections: Unlimited

SPAP banner  :Enabled
Welcome to the network...

Box-level dial-out settings
Inactive timer:                                        15
LAN Protocols enabled for dial-out:         TELNET DIALs
Server name:                                      DIALOUT_SERVER

Number of Mac Addresses defined = 0
Base MAC Address: 000000000000

VC: Maximum Virtual Connections = 50
VC: Maximum suspend time (hours) (0 is unlimited) = 12
VC: Idle timeout period (seconds) = 30

Multi-chassis MP: Endpoint discriminator (0 means use box s/n) = 0

DIALs config>
```

The example shows the following:

DIALs client IP address specification

Displays the IP address assignment techniques and whether they are enabled. You would receive this section of the display and the section containing the box-level dial-out settings in response to the **list ip-address-assignment** command.

IP address pools
Displays the configured IP address pools. You would receive this section of the display in response to the **list ip-pool** command.

Configured DHCP servers
Displays the list of IP addresses currently configured as DHCP servers. This section also lists the interface being used for the DHCP gateway. You would receive this section of the display in response to the **list dhcp-servers** command.

Dynamic Name Servers
Displays whether Dynamic DNS is enabled. You would receive this section of the display in response to the **list dynamic-dns** command.

primary domain server (dns)
This line and the following lines display the configured primary and secondary name servers. You would receive this section of the display in response to the **list name-servers** command.

time allowed
Displays the maximum amount of time (in minutes) for dials users. You would receive this section of the display in response to the **list time-allowed** command.

spap banner
Displays the contents of the spap banner. You would receive this section of the display in response to the **list spap-banner** command.

vc connections
Displays information about configured virtual connections.

multi-chassis mp
Displays the configured endpoint discriminator.

Set

Use the **set** command to set the time-allowed, dhcp gateway address, NetBIOS Name Server addresses, Dynamic Name Server addresses and dial-out inactivity timer , and dial-out server-name.

Syntax:

set
- dhcp-gateway-address
- dial-out . . .
- dns . . .
- laa
- multi-chassis-mp
- nbns . . .
- spap-banner . . .
- time-allowed
- vc-parameters

dhcp-gateway-address *interface# ipaddress*
Sets the IP address associated with the DHCP gateway. DHCP uses the address as:

1. An address to which DHCP replies

2. An indication of the pool of addresses from which DHCP allocates an IP address

If the DHCP server is not on a directly attached LAN interface, then you must configure this address to the address of one of the LAN interfaces that has IP connectivity to the DHCP server. See “Dynamic Host Configuration Protocol (DHCP)” on page 369 and the definition of “giaddr” in RFC 1541 for more information.

dial-out ***parameter***

Sets the inactivity timer or server name for dial-out nets. ***Parameter*** can be:

inactivity-timer

Sets the dial-out inactivity timer for dial-out nets. This is defined as the amount of time, in minutes, that a user can be connected without data traffic over the connection. For example, if the inactivity-timer is set to 5 minutes and during any 5 minute interval, no data is received or transmitted, the connection will be dropped and the modem will become available. The default is 0, which means that the inactivity timer is disabled and the connection will be maintained indefinitely.

servername

Sets the name of the dial-out server. This can be any string up to 30 characters in length. The default is “2210_DIALS_SERVER”. This is the name that the IBM DIALs Dial-Out clients see when they use the “Chooser” application to discover dial-out servers. This parameter has no meaning for telnet dial-out clients.

dns ***type ipaddress***

Configures the primary and secondary domain name servers (DNS). ***Type*** can be:

primary

Sets the IP address of the primary DNS server for the dial-in client to use. This value is negotiated during IPCP for some dial-up clients (particularly Windows 95).

secondary

Sets the IP address of the secondary DNS server for the dial-in client to use. This value is negotiated during IPCP for some dial-up clients (particularly Windows 95).

laa *#MAC_addresses MAC_address_base*

Sets the number of MAC addresses and the base address for the Locally Administered Address (LAA) table. Only Layer-2-Tunneling nets will use LAA addresses.

#MAC_addresses

Specifies the number of Mac addresses to add to the LAA table, beginning with the *MAC_Address_Base*.

Valid values: 0 to 256

Default value: 0

MAC_address_base

Specifies the base MAC address of the LAA table.

Valid values: Any valid MAC address

Default value: 000000000000

Example:

```
DIALs config>set laa
    Number of Mac Addresses:  [0]? 20
    Locally Administered Mac Address Base (hex) [000000000000]? 002210aaaaaa
    DIALs Config>
```

multi-chassis-mp

Sets the endpoint discriminator to be used. All links that are to join the same bundle must have the same endpoint discriminator.

Example:

```
DIALs Config> set multi-chassis-mp
   Enter Endpoint Discriminator to use from stacked group (0 for box S/N): 2345
```

nbns *type ipaddress*

Configures the primary and secondary NetBIOS name servers. ***Type*** can be:

primary

Sets the IP address of the primary NetBIOS name server.

secondary

Sets the IP address of the secondary NetBIOS name server.

spap-banner

Allows configuration of a message that is sent out to all clients that successfully complete SPAP authentication.

Example:

```
DIALs config>set spap-banner
SPAP banner  :Disabled

Enter Banner: Welcome to the network...
```

time-allowed

Sets the time allowed for PPP dial-in users and dial-out users. This parameter defines the maximum amount of time (in minutes) that a user can be connected. The default value is 0, which means the user can be connected for an unlimited amount of time.

vc-parameters

Use this parameter to set the global default virtual connection attributes. The system prompts you for the maximum number of connections, the maximum suspend time, and the inactivity timeout value.

Example:

```
Config> feature DIALs
DIALs Config> set vc-parameters
Maximum Virtual Connections [50]? 40
Maximum suspended time (hours) (0 is unlimited) [10]? 18
Inactivity Timeout (seconds) [30]? 60
DIALs Config>
```

Maximum Virtual Connections

The maximum number of virtual connections that can be active or suspended. When using VCs with MP, configure this value to be 1 greater than the number of physical connections.

Valid values: 0 to 255

Default value: 50

Maximum suspended time

The maximum amount of time, in hours, a virtual connection can be

suspended before the system ends the connection. Specifying 0 for this parameter allows a virtual connection to be suspended indefinitely.

Valid values: 0 to 48

Default value: 12

Inactivity Timeout
The number of seconds that a virtual connection can be inactive before it is suspended.

Valid values: 10 to 1024

Default value: 30

Accessing the DIALs Global Monitoring Environment

Use the following procedure to access the DIALs monitoring commands.

1. At the OPCON prompt, enter **talk 5**. (For details on this command, see the chapter "The OPCON Process and Commands" in *Software User's Guide*.) For example:

   ```
   * talk 5
   +
   ```

   After you enter the **talk 5** command, the GWCON prompt (+) displays on the terminal. If the prompt does not appear when you first enter configuration, press **Return** again.
2. At the + prompt, enter the **feature dials** command to get you to the `DIALS Console>` prompt and access the global monitoring environment.

   Example:

   ```
   + feature dials
   DIALS Console>
   ```

DIALs Global Monitoring Commands

Table 56. DIALs Global Monitoring Commands

Command	Function
Clear	Clears a specific suspended virtual connection.
List	Displays the state of various virtual connections, or all virtual connections.
Reset	Dynamically activates DIALS parameters.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page xxix.

Clear

Use the **clear** command to clear specific suspended virtual connections.

Syntax:

clear vc *connection_id*

vc *connection_id*
Specifies the suspended virtual connection that you are ending. To obtain the *connection_id*, enter either the **list all-vc** or **list suspended-vcs** command.

List

Use the **list** command to display all virtual connections, active virtual connections, suspended virtual connections, or the values of the vc-parameters.

Syntax:

list all
active-vcs
all-vcs
dhcp-servers
ip-address-assignment
ip-pool
suspended-vcs

active-vcs
Displays the attributes of all active virtual connections. See description of the **all-vcs** parameter for an explanation of the attributes.

all-vcs
Displays the attributes of all active and suspended virtual connections. This display is a combination of the displays for the **list active-vcs** and **list suspended-vcs** commands.

Example:

```
+ feature dials
DIALS console> list all
 DIALs client IP address assignment:
Client     :  Enabled
UserID     :  Enabled
Interface  :  Enabled
Pool       :  Enabled
DHCP Proxy :  Disabled


Current IP address pools:
     Base Address      Last Address         Total          Free
     ------------      ------------         -----          ----
*      11.0.0.100        11.0.0.129           30            30
       11.0.0.210        11.0.0.229           20            19

Current DHCP servers:           11.0.0.2         11.0.0.50
Proxy DHCP is currently disabled
DHCP gateway address (giaddr): 11.0.0.10


Active VCs:
Conn ID    Interface Idle-Timeout Connected Username
========== ========= ============ HHH:MM:SS ================
1656494850         8           30   0:26:15 don
7293521502         9           30   1:41:57 jane


Suspended VCs:
           Hrs.Max
Conn ID    Suspend Suspended Username
========== =======  HH:MM:SS ======================
9256166098      12   0: 4:13 joe
```

The attributes for active and suspended VCs are:

Conn ID
The connection id of the virtual connection. The system assigns the id when it establishes the connection.

Username
The AAA. RADIUS, or local-list user that establishes the virtual connection.

For active VCs:

Interface
The network interface that is managing the virtual connection.

Note: Do not assign IP addresses to dial-up clients using interface assignment to avoid problems caused by other users using this interface which the VC suspended.

Idle Timeout
The amount of inactive time, in seconds, after which the system will suspend the VC. This corresponds to the value of inactivity timer in the **set** command.

Connected HHH:MM:SS
The total amount of time in hours, minutes, and seconds, that the VC has been connected to an interface.

For suspended VCs:

Hrs. Max Suspended
The maximum number of hours a VC can be in suspend state before the system ends the connection. This corresponds to the value of maximum suspended time in the **set** command.

Suspended HH:MM:SS
The total amount of time in hours, minutes, and seconds, that the VC has been suspended.

dhcp-servers
Displays configured information about DHCP servers and their IP addresses.

ip-address-assignment
Display the methods by which IP addresses can be assigned to clients

ip-pool
Display the current usage of the pool.

Example:

```
DIALs Console> list ip-pool
Current IP address pools:
     Base Address      Last Address         Total          Free
     ------------      ------------         -----          ----
*    192.1.100.18      192.1.100.74            57            57
      192.2.200.1     192.2.200.250           250           250

Note: The * indicates from which block the next address will be retrieved.
```

suspended-vcs
Displays the attributes of all suspended virtual connections. See description of the **all-vcs** parameter for an explanation of the attributes.

vc-parameters
Displays the values of the vc-parameters that were set using the **set vc-parameters** command.

Reset

Use the **reset** command to dynamically activate the configuration changes made to the DIALs interface in `talk 6`.

Syntax:

reset all

dhcp-parameters

ip-address-assignment

ip-pool

vc-parameters

all Dynamically activate the DHCP, IP address assignment, and IP-pool configuration changes.

dhcp-parameters
Dynamically activate the DHCP configuration.

ip-address-assignment
Dynamically activate the IP address assignment method configuration.

ip-pool
Dynamically activate the IP address pool configuration.

vc-parameters
Dynamically updates VC config changes.

Dial-Out Interface Configuration Commands

To access the dial-out interface parameter environment:

1. Enter **talk 6** at the `*` prompt.
2. Enter **net *n*** at the `Config >` prompt.
3. Enter **encapsulator** at the `Circuit config: n>` prompt.

Table 57 lists the commands available from the `dial-out config>` prompt.

Table 57. Dial-Out Interface Configuration Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Set	Defines the port name associated with a modem.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Set

Use the **set** command to define the port name for a modem.

Syntax:

set portname *name*

portname
Defines the name of the port associated with a modem. Use this name to define ***modem pools.*** The name can be up to 30 characters in length.

Default value: ALL_PORTS

Example: `dial-out config>`**set portname localcalls**

Monitoring Dial-In Interfaces

Monitoring dial-in interfaces is the same as monitoring other PPP dial circuits. For details, see “Configuring and Monitoring Point-to-Point Protocol Interfaces” in the *Software User’s Guide* .

Monitoring Dial-Out Interfaces

Table 58 lists the commands available when monitoring dial-out interfaces.

Table 58. Dial-Out Interface Monitoring Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Clear	Resets the statistics for this dial-out interface.
List	Lists the current state of the dial-out interface, the number of bytes transmitted and received on this interface, and the client’s current parameters.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Clear

Use the **clear** command to reset the statistics for the number of octets received and transmitted by this interface.

Syntax:

clear

Example:

```
clear
Statistics reset.
```

List

Use the **list** command to display current state of the dial-out interface. The **list** command always displays the current state of the dial-out net, the time since the state change, and the number of bytes received and transmitted.

Syntax:

list

Example for inactive interface:

```
list
Dial-out Settings for current session:

Dial-out state is DOWN
Time since change           = 52 minutes and 34 seconds

Dial-out Octets transmitted = 0
Dial-out Octets received    = 0

Session down, no valid settings
```

Note: When a client connects to a dial-out port using telnet, no user name is present because the server did not perform any authentication.

Example for active interface:

```
list
Dial-out Settings for current session:

Dial-out state is UP
Time since change           = 3 seconds

Dial-out Octets transmitted = 14
Dial-out Octets received    = 765

Current user                = not available
Time allowed for user       = unlimited
Inactivity timer for port   = 10 minutes
Line speed                  = 57600
Current DTR state           = DTR ON
Current dial-out protocol   = TELNET
Options negotiated:
        Will Suppress Go Ahead
        Wont' Echo characters
```

Example for an active IBM DIALs Dial-Out client:

```
list
Dial-out Settings for current session:

Dial-out state is UP
Time since change           = 12 seconds

Dial-out Octets transmitted = 11
Dial-out Octets received    = 756

Current user                = ebooth
Time allowed for user       = unlimited
Inactivity timer for port   = 10 minutes
Line speed                  = 57600
Current DTR state           = DTR ON
Current dial-out protocol   = DIALs
```

Chapter 28. Using DHCP Server

This chapter describes how to use the DHCP Server. It includes the following sections:

Introduction to DHCP

Dynamic Host Configuration Protocol (DHCP) is a client/server protocol that is based upon the Bootstrap Protocol (BOOTP). The DHCP server provides centrally controlled reusable IP addresses and other TCP/IP configuration information for DHCP clients. Its functionality can alleviate the burden that Network Managers have of distributing configuration information to new and existing users. This feature is compliant to RFC 2131 but supports many additional features not included in that document. There is also support for BOOTP clients as defined in RFC 951.

With DHCP, supporting clients can send broadcast DISCOVER messages to find DHCP servers in their network and subsequently be OFFERED their configuration data dynamically across the network. DHCP uses the well know BOOTP UDP ports (68 for the server and 67 for the client) to communicate requests and responses. DHCP clients and servers can use existing BOOTP relay agents to extend their service range. DHCP offers many advantages over statically configured networks, including the ability to support changing networks. Clients are only leased their IP addresses so when they no longer have a need for it or are moving to another subnet, the address can be RELEASED and made available for other clients to use.

DHCP Operation

DHCP allows clients to obtain IP network configuration information, including an IP address, from a central DHCP server. DHCP servers control whether the addresses they provide to clients are allocated permanently or are leased for a specific time period. When a client receives a leased address, it must periodically request that the server revalidate the address and renew the lease.

The processes of address allocation , leasing, and lease renewal are all handled by the DHCP client and server programs and are transparent to end-users. The clients use RFC architected messages to accept and use the options served them by the DHCP server. For example:

1. The client broadcasts a message (containing its client ID) announcing its presence and requesting an IP address (DHCPDISCOVER message) and desired options such as subnet mask, domain name server, domain name and static route.
2. Optionally, if routers on the network are configured to forward DHCP and BOOTP messages (using BOOTP Relay), the broadcast message is forwarded to DHCP servers on the attached networks.

3. Each DHCP server that receives the client's DHCPDISCOVER message sends a DHCPOFFER message to the client offering an IP address. The DHCP server checks for duplicate IP addresses on the network before issuing an offer. The server checks the configuration file to see if it should assign a static or dynamic address to this client. In the case of a dynamic address, the server selects an address from the address pool, choosing the least recently used address. An address pool is a range of IP addresses to be leased to clients. In the case of a static address, the server uses a Client statement from the DHCP server configuration to assign options to the clients. Upon making the offer, the DHCP server reserves the offered address.
4. The client receives the offer message(s) and selects the server it wants to use. When a DHCP client receives an offer, it makes note of how many of the requested options are included in the offer. The DHCP client continues to receive offers from DHCP servers for a period of 4 seconds after the first offer is received, making note of how many of the requested options are included in each offer. At the end of that time, the DHCP client compares all offers and selects the one that meets its criteria.
5. The client broadcasts a message to indicate the server it selected and requests use of the IP address offered by that server (DHCPREQUEST message).
6. If a server receives a DHCPREQUEST message indicating that the client has accepted the server's offer, the server marks that address as leased. If the server receives a DHCPREQUEST message indicating that the client has accepted an offer from a different server, the server returns the address to the available pool. If no message is received within a specified time, the server returns the address to the available pool. The selected server sends an acknowledgment which contains additional configurations information to the client (DHCPACK message).
7. The client determines whether the configuration information is valid. Upon receipt of a DHCPACK message, the DHCP clients sends an Address Resolution Protocol (ARP) request to the supplied IP address to see if it is already in use. If it receives a response to the ARP request, the client declines (DHCPDECLINE message) the offer and initiates the process again. Otherwise, the client accepts the configuration information.
8. Accepting a valid lease, the client enters a BINDING state with the DHCP server, and proceeds to use the IP address and options. If the DHCP client is a Dynamic-Address client, the DHCP client notifies the Dynamic Domain Name Server of its host name-to-IP address mapping.

To DHCP clients that request options, the DHCP server typically provides options that include subnet mask, domain name server, domain name, static route, class-identifier (which indicates a particular vendor), and user class.

However a DHCP client can request its own, unique set of options. For example, Windows NT 3.5.1 DHCP clients are required to request options. The default set of client requested DHCP options provided by IBM includes subnet mask, domain name server, domain name, and static route. For option descriptions, see "DHCP Options" on page 394.

Lease Renewals

The DHCP client keeps track of how much time is remaining on the lease. At a specified time prior to the expiration of the lease, usually when half of the lease time has passed, the client sends a renewal request, containing its current IP

address and configuration information, to the leasing server. If the server responds with a lease offer, the DHCP client's lease is renewed.

If the DHCP server explicitly refuses the request, the DHCP client may continue to use the IP address until the lease time expires and then initiate the address request process, including broadcasting the address request. If the server is unreachable, the client may continue to use the assigned address until the lease expires.

Client Movement

One benefit of DHCP is the freedom it provides a client host to move from one subnet to another without having to know ahead of time what IP configuration information it needs on the new subnet. As long as the subnets to which a host relocates have access to a DHCP server, a DHCP client will automatically configure itself correctly to access those subnets.

In order for DHCP clients to reconfigure to access a new subnet, the client host must be rebooted. When a host restarts on a new subnet, the DHCP clients tries to renew its old lease with the DHCP server which originally allocated the address. The server refuses to renew the request since the address is not valid on the new subnet. Receiving no server response or instructions from the DHCP server, the client initiates the IP address request process to obtain a new IP address and access the network.

Changing Server Options

With DHCP, you can make changes at the server, reinitialize the server, and distribute the changes to all the appropriate clients. A DHCP client retains DHCP option values assigned by the DHCP server for the duration of the lease. If you implement configuration changes at the server while a client is already up and running, those changes are not processed by the DHCP client until the clients attempts to renew its lease or until it is restarted.

Note: If the server is reinitialized (using the `t 5` **reset dhcp** command), the lease time information displayed by the router will be lost until the DHCP clients renew their lease.

Number of DHCP servers

The number of servers that you need will depend largely on the number of subnets you have, the number of DHCP clients you plan to support, whether you use BOOTP Relay, and the lease time you choose. Keep in mind that the DHCP protocol does not currently define server-to-server communication. Thus, they cannot share information, nor can one DHCP server perform as a "hot backup" in the event the other one fails. DHCP clients send broadcast messages. By design, broadcast messages do not cross subnets. To allow the client's messages to be forwarded outside its subnet, additional routers must be configured to forward DHCP requests using the BOOTP Relay agent. Otherwise, you will need to configure a DHCP server on each subnet.

A Single DHCP server

If you choose to use a single DHCP server to serve hosts on a subnet, consider the effects if the single server fails. Generally, the failure of a server will affect only DHCP clients that are attempting to join the network. Typically, DHCP clients

already on the network will continue operating unaffected until their lease expires. However, clients with a short lease time may lose their network access before the server can be restarted. To minimize the impact of server downtime if you have only one DHCP server for a subnet, you should choose a sufficiently long lease time to allow time to restart or respond to the failed DHCP server.

Multiple DHCP servers

To avoid a single point of failure, you can configure two or more DHCP servers to serve the same subnet. If one server fails, the other can continue to serve the subnet. Each of the DHCP servers must be accessible either by direct attachment to the subnet or by using a BOOTP Relay agent.

Because two DHCP servers cannot serve the same addresses, address pools defined for a subnet must be unique across DHCP servers. Therefore, when using two or more DHCP servers to serve a particular subnet, the complete list of addresses for that subnet must be divided among the servers. For example, you could configure one server with an address pool consisting of 70% of the available addresses for the subnet and the other server with an address pool consisting of the remaining 30% of the available addresses.

Using multiple DHCP servers decreases the probability of having a DHCP related network access failure, but does not guarantee against it. If a DHCP server for a particular subnet fails, the other DHCP server may not be able to service all the requests from new clients which may, for example , exhaust the server's limited pool of available addresses.

However, you can bias which DHCP server exhausts it pool of addresses first. DHCP clients tend to select the DHCP server offering more options. To bias service toward the DHCP server with 70% of the available addresses, offer fewer DHCP options from the server holding 30% of the available addresses for the subnet.

BOOTP Servers

If you already have BOOTP clients and servers in your network, you may want to consider replacing your BOOTP servers with DHCP servers. DHCP servers can optionally serve BOOTP clients the same IP configuration information as current BOOTP servers. If you cannot replace your BOOTP servers with DHCP servers and want to have both serve your network, the following precautions are recommended:

- Turn off BOOTP support in the DHCP server.
- Make sure your BOOTP servers and DHCP servers do not give out the same addresses.
- Configure the BOOTP Relay support in your routers to forward BOOTP broadcasts to both the appropriate BOOTP and DHCP servers.

A DHCP server allocates a permanent IP address to a BOOTP client. In the event that subnets are renumbered in such a way that a BOOTP assigned address is unusable, the BOOTP client must restart and obtain a new IP address.

Special DHCP Clients

You may have DHCP clients or Network Servers which have individual or special administrative needs, such as:

- A Permanent Lease:

You can assign permanent leases to designated hosts by specifying an infinite lease time. Also the DHCP server will allocate a permanent lease to BOOTP clients that explicitly request it as long as support for BOOTP clients is enabled. The DHCP server will also allocate a permanent lease to DHCP hosts that explicitly request it.

- A Specific IP Address:

  You can reserve a specific address and configuration parameters for a specific DHCP or BOOTP client host on a particular subnet.

- Specific Configuration Parameters:

  You can allocate specific configuration information to a client regardless of its subnet.

- Manually Defined Workstations:

  You should explicitly exclude addresses from DHCP subnets for existing hosts that do not use DHCP or BOOTP for configuring their IP network access. Although DHCP servers and clients automatically check to see if an IP address is in use before allocating or using it, they will not be able to detect addresses of manually defined hosts that are turned off or temporarily off the network. In that case, duplicate address problems may occur when a manually defined host reaccesses the network, unless its IP address is explicitly excluded.

Lease Times

The default lease time is 24 hours. Keep in mind that the DHCP lease time can affect your network operation and performance:

- Short lease times will increase the amount of network traffic due to DHCP lease renewal requests. For example, if you set a lease time of 5 minutes, each client sends a renewal request about every 2.5 minutes.
- Lease times that are too long can limit the ability to reuse IP addresses. Very long lease times also delay configuration changes that occur when a client restarts or renews a lease.

The lease time you choose depends largely on your needs, including:

- The number of hosts to support compared to the number of available addresses. If you have more hosts than addresses, you may want to choose a short lease time of one to two hours. This will help ensure that unused addresses are returned to the pool as soon as possible.
- The time available to make network changes. Hosts receive changes to configuration information when they are restarted or renew their lease. Be sure to allow a timely and adequate window to make these changes. For example, if you usually make changes overnight, you might assign a lease time of 12 hours.
- The number of DHCP servers that are available. If you have only a few DHCP servers for a large network, you may want to choose a longer lease time to minimize the impact of server down time.

For complex networks that need to support a combination of host leasing requirements, you can define DHCP classes.

Concepts and Terminology

The following concepts are used to describe the DHCP server function:

Scope The term scope, when discussing the DHCP server Configuration, will be used to identify what a certain parameter value pertains to. Figure 36 on page 392 illustrates the following scopes:

- Global option 1
- Global option 3
- Global class ClassA

  ClassA has redefined option 1, but will inherit the value of option 3 from the global scope.
- Global client ClientA

  ClientA has redefined option 3, but will inherit the value of option 1 from the global scope.
- subnet SubA
  - Redefines Option 1.
  - Inherits the value of Option 3 from the global scope.
  - Defines ClassA within the scope of SubA.

    It redefines the value of option 1, but will inherit the value of option 3 from SubA (which also happens to be inherited from the global scope).
  - Defines ClientB within the scope of SubA.

    ClientB has redefined option 3, but will inherit the value of option 1 from SubA.
- vendor-option vendorA

  Vendor-options are an exception. Vendor-options are independent and are not inherited outside of the vendor-option scope.

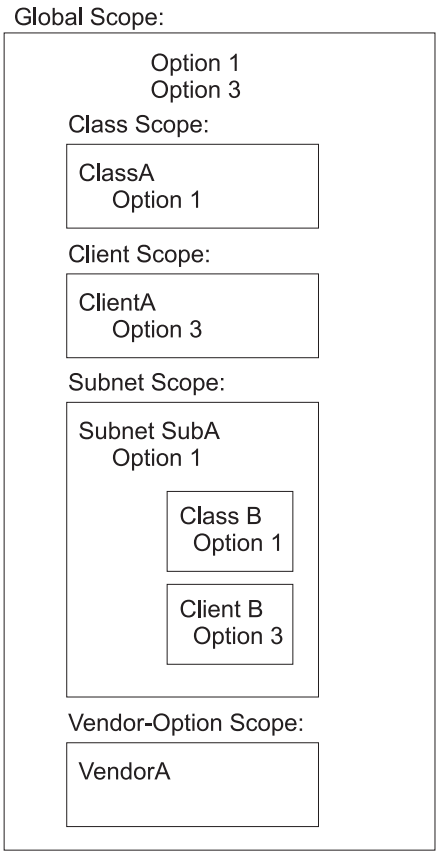


Figure 36. Scope Concepts

Subnet

A subnet defines the parameters for an address pool administered by a DHCP server. An address pool is a range of IP addresses to be leased to clients. Parameters that can be specified include the lease time and other options for clients using the address pool. The lease time and other options can be inherited from the global scope.

Subnet Groups

A subnet group is a way to identify multiple subnets that are to be grouped together on the same interface. All the subnets in a given group are given the same subnet group name and a unique priority. The priority is used to determine the order addresses are given out according to the address policy the group is associated with. A subnet can belong to one of two address policies:

- Inorder

  This policy is the default. The inorder policy administers addresses starting with the subnet with the lowest priority and ending with the subnet with the highest priority.

- Balance

  The balance policy administers addresses from the group of defined subnets in a round-robin order. The first address is administered from the subnet with the lowest priority. The second address is administered from the subnet with the next lowest priority, and so on. When an address from the highest priority subnet has been administered, the policy returns to the subnet with the lowest priority until all addresses are exhausted from all the subnets in the group.

Classes

A class defines the parameters for a user defined group of clients, administered by the DHCP server. Classes can be defined under the global or a subnet scope. When a class is defined within a subnet scope, the DHCP server will only serve clients in the class that are both located in the specified subnet and request the class. Only classes that are defined within a subnet's scope can specify a range of addresses. The range can be either a subset of the subnet range or can be equal to the subnet range. A client that requests an IP address from a class which has exhausted its range is offered an IP address from the subnet range, if available. The client is offered the options associated with the exhausted class.

Clients

A client can be used to:

- Define a static IP address and DHCP options for a specific end station
- Exclude a specific endstation from service
- Exclude an IP address from a range of available IP addresses

Each client has a specified hardware type, client id and IP address. The hardware types are defined in RFC 1340 and are shown below. For all hardware types besides 0, the client ID is the hardware address of the endstation (or MAC address). For hardware type of 0, the client id is a character string. Typically, this would be a domain name.

When defining a client, you are prompted for either an IP address, *any* or *none*. If you define an IP address, that IP address is reserved for that client. If you choose *any*, then that client will be given any available IP address within that subnet. If you have several subnets records defined within the same subnet, each having a unique range, then a client that is configured

with *any* will get the first available address within the subnet, not necessarily from the range of the specific subnet record that the client is defined under. If you choose *none*, then that end station will not be served any IP address at all. To exclude an IP address from being administered, you would define a client record with a hardware type and client id of 0.

Hardware types that are defined by RFC1340 and that pertain to the IBM 2210 are:

```
Hardware Type                              Value
-------------                              -----
Reserved                                   0
Ethernet                                   1
IEEE 802 Networks (including Token Ring)   6
```

For the complete list, refer to RFC 1340.

DHCP Server and Lease Parameters

The following DHCP server parameters can be defined at the global level:

- bootstrapserver
- canonical
- lease expire interval
- lease time default
- ping time
- support unlisted clients
- support bootp
- used ip address expire interval

See “Set” on page 434 for a description of these parameters.

DHCP Options

DHCP allows you to specify options to provide additional configuration information to a client. The options are defined in RFC 2132 and various other RFCs.

Option Formats

All options expect the configuration data to be in one of the following formats:

Format	Definition
IP address	A single IP address in dotted-decimal notation.
IP addresses	One or more IP addresses in dotted-decimal notation, separated by blanks.
IP address pair	Two IP addresses in dotted-decimal notation, separated by blanks.
IP address pairs	One or more IP address pairs, each pair separated from another by a blank.
Boolean	0 or 1 (True or False).
Byte	A decimal number between -128 and 127 (inclusive).

Unsigned byte	A decimal number between 0 and 255 (inclusive). You cannot specify a negative value for an unsigned byte.
List of unsigned bytes	One or more decimal numbers between 0 and 255 (inclusive) separated by blanks. You cannot specify a negative number for an unsigned byte.
Short	A decimal number between -32768 and 32767 (inclusive).
Unsigned short	A decimal number between 0 and 65535 (inclusive). You cannot specify a negative number for an unsigned short.
List of unsigned shorts	One or more decimal numbers between 0 and 65535 (inclusive) separated by blanks. You cannot specify a negative number for an unsigned short.
Long	A decimal value between -2147483648 and 2147483647 (inclusive).
Unsigned long	A decimal number between 0 and 4294967295 (inclusive). You cannot specify a negative number for an unsigned long.
String	A string of characters.
N/A	Indicates no specification is needed because the client generates this information.

Each DHCP option is identified by a numeric code.

Architected options 0 through 127 and option 255 are reserved for definitions by RFCs. The DHCP server, the DHCP client, or both server and client use options in this set. Some architected options can be modified by the administrator. Other options are for exclusive use by the client and server.

Note: Hexadecimal values are not allowed for architected options with known formats.

Options that the administrator cannot or should not configure at the DHCP server include:

52 Option Overload

53 DHCP message type

54 Server identifier

55 Parameter request list

56 Message

57 Maximum DHCP message size

60 Class identifier

Options 128 through 254 represent user-defined options that can be defined by administrators to pass information to the DHCP client to implement site-specific configuration parameters.

Additionally, IBM provides a set of IBM-specific options such as option 192: TXT RR

The format of a user-defined option is:

Syntax:

option *code value*

where,

code Any option code from 1 through 254, except codes that are already defined in a RFC.

value Must always be a string. At the server, it can be an ASCII string or a hexadecimal string. At the client, however, it always appears as a hexadecimal string as passed to the processing program.

The server passes the specified value to the client. However, a program or command file must be created to process the value.

Base Options Provided to the Client

The following base options are provided to the client. See “Option Formats” on page 394 for a description of the configuration format.

1 **Subnet Mask** This option is specified only at the DHCP server. The client’s subnet mask, specified in a 32-bit dotted-decimal notation. Although not required, in most configurations the DHCP server should send option 1, subnet mask, to the DHCP clients. Client operation may be unpredictable if the client receives no subnet mask from the DHCP server and assumes a subnet mask that is not appropriate of the subnet. If not specified, the client used the default subnet masks:

- Class A network 255.0.0.0
- Class B network 255.255.0.0
- Class C network 255.255.255.0

Option format: IP addresses

2 **Time Offset** This options is specified only at the DHCP server. The offset (in seconds) of the client’s subnet from Coordinated Universal Time (CUT). The offset is a signed 32-bit integer.

Option format: Long

3 **Router** This option is specified only at the DHCP server. IP addresses (in order of preference) of the routers on the client’s subnet.

Option format: IP addresses

4 **Time Server** This option is specified only at the DHCP server. IP addresses (in order of preference) of the time servers available to the client.

Option format: IP addresses

5 **Name Server** This option is specified only at the DHCP server. IP addresses (in order of preference) of the IEN 116 name servers available to the client.

Note: This is not the Domain Name Server option. Use Option 6 to specify a Domain name server.

Option format: IP addresses

6 **Domain Name Server** This option is specified only at the DHCP server. IP addresses (in order of preference) of the Domain Name System servers available to the client.

Option format: IP addresses

Note: If Dynamic-Address is configured on a PPP interface, you may be able to retrieve a Primary and a Secondary DNS address using IPCP from an Internet Service Provider (ISP). To pass these DNS addresses along to the DHCP clients in option 6, you must configure the un-numbered IP interface address (such as 0.0.0.n) that corresponds to the Dynamic-Address interface. The DHCP server will convert this value to the retrieved addresses when the client sends a request. If the DHCP client has requested its configuration information from the Server prior to the PPP interface activating, the client will have to restart or renew its lease to receive the learned DNS addresses.

7 **Log Server** This option is specified only at the DHCP server. IP addresses (in order of preference) of the MIT-LCS UDP Log servers available to the client.

Option format: IP addresses

8 **Cookie Server** This option is specified only at the DHCP server. IP addresses (in order of preference) of the Cookie, or quote-of -the-day servers available to the client.

Option format: IP addresses

9 **LPR Server** This option can be specified at both the DHCP client and DHCP server. However, if specified only at the DHCP client, the configuration will be incomplete. IP addresses (in order of preference) of the line printer servers available to the client. Option 9 eliminates the need for clients to specify the LPR_SERVER environment variable.

Option format: IP addresses

10 **Impress Server** This option is specified only at the DHCP server. IP addresses (in order of preference) of the Imagen Impress servers available to the client.

Option format: IP addresses

11 **Resource Location Server** This option is specified only at the DHCP server. IP addresses (in order of preference) of the Resource Location (RLP) servers available to the client. RLP servers allow clients to locate resources that provide a specified service, such as a domain name server.

Option format: IP addresses

12 **Host Name** This option can be specified at both the DHCP client and the DHCP server. If the DHCP client does not provide a host name, the DHCP server ignores option 12. Host name of the client (which may include the local domain name). The minimum length for the host name option is 1 octet and the maximum is 32 characters. See RFC 1035 for character set restrictions.

Option format: String

13 **Boot File Size** This option is specified only at the DHCP server. The length (in 512-octet blocks) of the default boot configuration file for the client.

Option format: Unsigned short

14 **Merit Dump File** This option is specified only at the DHCP server. The path name of the merit dump file in which the client's core image is stored if the client crashes. The path is formatted as a character string consisting of characters from the Network Virtual Terminal (NVT) ASCII character set. The minimum length is 1 octet.

Option format: String

15 **Domain Name** This option is specified at both the DHCP client and the DHCP server. If no value is specified at the DHCP server in option 15, the client is required to provide a value for option 12, host name, and option 15, domain name. This statement may appear within the global scope, or with a Subnet, Class or Client scope.

Option format: String

16 **Swap Server** This option is specified only at the DHCP server. The IP address of the client's swap server.

Option format: IP address

17 **Root Path** This option is specified only at the DHCP server. The path that contains the client's root disk. The path is formatted as a character string consisting of characters from the NVT ASCII character set. The minimum length is 1 octet.

Option format: String

18 **Extension Path** This option is specified only at the DHCP server. The extension path option specifies a string that can be used to identify a file that is retrievable using the Trivial File Transfer Protocol (TFTP). The minimum length is 1 octet.

Option format: String

IP Layer Parameters per Host Options

19 **IP Forwarding** This option is specified only at the DHCP server. Enable (1) or disable (0) forwarding by the client of its IP layer packets.

Option format: Boolean

20 **Non-Local Source Routing** This option is specified only at the DHCP server. Enable (1) or disable (0) forwarding by the client of its IP layer data grams with non-local source routes.

Option format: Boolean

21 **Policy Filter** This option is specified only at the DHCP server. IP address-net mask pair used to filter data grams with non-local source routes. Any data gram whose next hop address does not match one of the filter pairs is discarded by the client. The minimum length for the policy filter option is 8 octets.

Option format: IP address pairs

22 **Maximum Data gram Reassembly Size** This option is specified only at the DHCP server. Maximum size data gram the client will reassemble. The minimum value is 576.

Option format: Unsigned short

23 **Default IP Time-to-Live** This option is specified only at the DHCP server.

Default time-to-live (TTL) the client uses on outgoing data grams. TTL is an octet with a value between 1 and 255.

Option format: Unsigned byte

24 **Path MTU Aging Timeout** This option is specified only at the DHCP server. Timeout in seconds used to age Path Maximum Transmission Unit (MTU) values discovered by the mechanism that is described in RFC 1191.

Option format: Unsigned long

25 **Path MTU Plateau Table** This option is specified only at the DHCP server. Table of MTU sizes to sue in Path MTU discover as defined in RFC 1191. The minimum MTU value is 68. The minimum length for the path MTU plateau table option is 2 octets. The length must be a multiple of 2.

Option format: Unsigned short

IP Layer Parameters per Interface Options

26 **Interface MTU** This option is specified only at the DHCP server. Maximum Transmission Unit (MTU) to sue on this interface. The minimum MTU value is 68.

Option format: Unsigned short

27 **All Subnets are Loca**l This option is specified only at the DHCP server. Client assumes (1) or does not assume (0) all subnets use the same Maximum Transmission Unit (MTU). A value of 0 means the client assumes some subnets have smaller MTUs.

Option format: Boolean

28 **Broadcast Address** This option is specified only at the DHCP server. Broadcast address used on the client's subnet.

Option format: IP address

29 **Perform Mask Discovery** This option is specified only at the DHCP server. Clients performs (1) or does not perform (0) subnet mask discovery using Internet Control Message Protocol (ICMP).

Option format: Boolean

30 **Mask Supplier** This option is specified only at the DHCP server. Client responds (1) or does not respond (0) to subnet mask requests using Internet Control Message Protocol (ICMP).

Option format: Boolean

31 **Perform Router Discovery** This option is specified only at the DHCP server. Client solicits (1) or does not solicit (0) routers using router discovery as defined in RFC 1256.

Option format: Boolean

32 **Router Solicitation Address** This option is specified only at the DHCP server. Address to which a client transmits router solicitation requests.

Option format: IP address

33 **Static Route** This option is specified only at the DHCP server. Static routes (designation address-router pairs in order of preference) the client installs in its routing cache. The first address is the destination address and the second address is the router for the destination. Do not specify 0.0.0.0 as a default route destination.

Option format: IP address pairs

Link Layer Parameters per Interface Options

34 **Trailer Encapsulation** This option is specified only at the DHCP server. Client negotiates (1) or does not negotiate (0) the use of trailers when using Address Resolution Protocol (ARP). For more information see RFC 893.

Option format: Boolean

35 **ARP Cache Timeout** This option is specified only at the DHCP server. Timeout in seconds for Address Resolution Protocol (APR) cache entries.

Option format: Unsigned long

36 **Ethernet Encapsulation** This option is specified only at the DHCP server. For an Ethernet interface, client uses IEEE 802.3 (1) Ethernet encapsulation described in RFC 1042 or Ethernet V2 (0) encapsulation described in RFC 894.

Option format: Boolean

TCP Parameter Options

37 **TCP Default TTL** This option is specified only at the DHCP server. Default time-to-live (TTL) the client uses for sending TCP segments.

Option format: Unsigned byte

38 **TCP Keep-alive Interval** This option is specified only at the DHCP server. Interval in seconds the client waits before sending a keep-alive message on a TCP connection. A value of 0 indicates the client does not send keep-alive messages unless requested by the application.

Option format: Unsigned long

39 **TCP Keep-alive Garbage** This option is specified only at the DHCP server. Client sends (1) or does not send (0) TCP keep-alive messages that contain an octet of garbage for compatibility with previous implementations.

Option format: Boolean

Application and Service Parameter Options

40 Network Information Service Domain This option is specified only at the DHCP server. The client's Network Information Service (NIS) domain. The domain is formatted as a character string consisting of characters from the NVT ASCII character set. The minimum length is 1 octet.

Option format: String

41 **Network Information Service Domain** This option is specified only at the DHCP server. IP addresses (in order of preference) of Network Information Service (NIS) servers available to the client.

Option format: IP addresses

42 **Network Time Protocol Servers** This option is specified only at the DHCP server. IP addresses (in order of preference) of Network Time Protocol (NTP) servers available to the client.

Option format: IP addresses

43 **Vendor-Specific Information** Option 43 is specified only at the DHCP

server, which returns this option to a client that sends option 60, Class Identifier. This information option is used by clients and servers to exchange vendor-specific information, which is specified in the vendor-option definition. Considerations in using Option 43 to encapsulate vendor information are:

- To permit interoperability between clients and servers from different vendors, each vendor must clearly document its option 43 content using the standard format from RFC 2132.
- Each vendor should specify the specific options that can be encapsulated within option 43 in a form that DHCP servers from another vendor can easily implement. For example, the vendor should:
  - Represent those options either in a data types already defined for DHCP options or in other well-defined data types.
  - Choose options that can be readily encoded in configuration files for exchange with servers provided by other vendors.
  - Be readily supportable by all servers.

Servers that cannot interpret the vendor-specific information sent by a client must ignore it. Clients that do not receive desired vendor-specific information should attempt to operate without it. Refer to RFC 2131 and RFC 2132 for additional information about this option.

Note: Because of these considerations, IBM instead uses options 192 and 200 for IBM-specific options.

Option format: String

44 **NetBIOS over TCP/IP Name Server** This option is specified only at the DHCP server. IP addresses (in order of preference) of NetBIOS name servers (NBNS) available to the client.

Option format: IP addresses

45 **NetBIOS over TCP/IP Datagram Distribution Server** This option is specified only at the DHCP server. IP addresses (in order of preference) of NetBIOS data gram distribution (NBDD) name servers available to the client.

Option format: IP addresses

46 **NetBIOS over TCP/IP Node Type** This option is specified only at the DHCP server. Node type used for NetBIOS over TCP/IP configurable clients as described in RFC 1001 and RFC 1002. Values to specify client types include:

- 0x1 B-node
- 0x2 P-node
- 0x4 M-note
- 0x8 H-node

Option format: Unsigned byte

47 **NetBIOS over TCP/IP Scope** This option is specified only at the DHCP server. NetBIOS over TCP/IP scope parameter for the client, as specified in RFC 1001/1002. The minimum length is 1 octet.

Option format: Unsigned byte

48 **X Window System Font Server** This option is specified only at the DHCP server. IP addresses (in order of preference) of X Window System font servers available to the client.

Option format: IP addresses

49 **Window System Display Manager** This option is specified only at the DHCP server. IP addresses (in order of preference) of systems running X Window System Display Manager available to the client.

Option format: IP addresses

DHCP Extensions Options

50 **Requested IP Address** This option is specified only at the DHCP client. The DHCP server can refuse a DHCP client request for a specific IP address. Allows the client to request (DHCPDISCOVER) a particular IP address.

Option format: N/A

51 **IP Address Lease Time** This option can be specified at both the DHCP client and the DHCP server. The DHCP client can use option 51 to override the defaultLeaseInterval value the DHCP server offers. Allows the client to request (DHCPDISCOVER or DHCPREQUEST) a lease time for an IP address. In a reply (DHCPOFFER), a DHCP server uses the option to offer a lease time. This option may be specified within the global, subnet, class or client scope. Use X'ffffffff' to indicate an infinite (permanent) lease.

Option format: Unsigned long

58 **Renewal (T1) Time Value** This option is specified only at the DHCP server. Interval in seconds between the time the server assigns an address and the time the client transitions to the renewing state.

Option format: Unsigned long

59 **Rebinding (T2) Time Value** This option is specified only at the DHCP server. Interval in seconds between the time the server assigns an address and the time the clients enters the rebinding state.

Option format: Unsigned long

60 **Class-Identifier** This option is specified only at the DHCP client. This information is generated by the client and does not have to be specified. Type and configuration of the client, supplied by the client to the server. For example, the identifier may encode the client's vendor-specific hardware configuration. The information is a string of *n* octets, interpreted by servers. For example: hex: X'01' X'02' X'03'. Servers not equipped to interpret the class-specific information sent by a client must ignore it. The minimum length is 1 octet.

Option format: N/A

61 **Client Identifier** This option can be specified at both the DHCP client and the DHCP server. The DHCP client can use option 61 to specify the unique client identifier. The DHCP server can use option 61 to index the database of address bindings. This value is expected to be unique for all clients in an administrative domain.

Option format: String

62 **NetWare/IP Domain Name** This option is specified only at the DHCP server. Netware/IP Domain Name. The minimum length is 1 octet and the maximum length is 255

Option format: String

63 **NetWare/IP** This option is specified only at the DHCP server. A general purpose option code used to convey all the NetWare/IP related information except for the NetWare/IP domain name. A number of NetWare/IP sub-options will be conveyed using the option code. The minimum length is 1 and the maximum length is 255.

Option format: String

64 **NIS domain Name** This option is specified only at the DHCP server. Network Information Service (NIS)+ V3 client domain name. The domain is formatted as a character string consisting of characters from the NVT ASCII character set. Its minimum length is 1.

Option format: String

65 **NIS Servers** This option is specified only at the DHCP server. IP addresses (in order of preference) of Network Information Service (NIS+ V3 servers available to the client.

Option format: IP addresses

66 **Server Name** This option is specified only at the DHCP server. Trivial File Transfer Protocol (TFTP) server name used when the “sname” field in the DHCP header has been used for DHCP options.

Option format: String

67 **Boot File Name** This option is specified only at the DHCP server. Name of the boot file when the file field in the DHCP header has been used for the DHCP options. The minimum length is 1.

Note: Use this option to pass a boot file name to a DHCP client. The boot file name is required to contain the fully-qualified path name and be less than 128 characters in length. For example: option 67 c:\path\boot_file_name. This file contains information that can be interpreted in the same way as the 64-octet vendor-extension field within the BOOTP response, with the exception that the file length is limited to 128 characters by the BootP header.

Option format: String

68 **Home Address** This option is specified only at the DHCP server. IP addresses (in order of preference) of the mobile IP home agents available to the client. The option enables a mobile host to derive a Mobil home address, and determine the subnet mask for the home network. The usual length will be four octets, containing a single home agent’s home address, but the length can be zero. A zero length indicates that no home agents are available.

Option format: IP addresses

69 **SMTP Servers** This option is specified only at the DHCP server. IP addresses (in order of preference) of the Simple Mail Transfer Protocol (SMTP) servers available to the client.

Option format: IP addresses

70 **POP3 Server** This option is specified only at the DHCP server. IP addresses (in order of preference) of the Post Office Protocol (POP) servers available to the client.

Option format: IP addresses

71 **NNTP Server** This option is specified only at the DHCP server. IP addresses (in order of preference) of the Network News Transfer Protocol (NNTP) servers available to the client.

Option format: IP addresses

72 **WWW Server** This option is specified only at the DHCP server. IP addresses (in order of preference) of the World Wide Web (WWW) servers available to the client.

Option format: IP addresses

73 **Finger Server** This option is specified only at the DHCP server. IP addresses (in order of preference) of the Finger servers available to the client.

Option format: IP addresses

74 **IRC Server** This option is specified only at the DHCP server. IP addresses (in order of preference) of the Internet Relay Chat (IRC) servers available to the client.

Option format: IP addresses

75 **StreetTalk Server** This option is specified only at the DHCP server. IP addresses (in order of preference) of the StreetTalk servers available to the client.

Option format: IP addresses

76 **STDA Server** This option is specified only at the DHCP server. IP addresses (in order of preference) of the StreetTalk Directory Assistance (STDA) servers available to the client.

Option format: IP addresses

77 **User Class** This option is specified only at the DHCP client. DHCP clients use option 77 to indicate to DHCP servers what class the host is a member of. The user class must be manually entered in the \DHCPCD.CFG file as the value for option 77 in order to receive parameters defined for the class at a DHCP server. The DHCPCD.CFG file is located in the ONDEMAND\SERVER\ETC directory.

Option format: String

78 **Directory Agent** This option is specified only at the DHCP server. The Dynamic Host Configuration Protocol provides a framework for passing configuration information to hosts on a TCP/IP network. Entities using the Service Location Protocol need to find out the address of Directory Agents in order to transact messages. In certain other instances they may need to discover the correct scope and naming authority to be used in conjunction with the service attributes and URLs which are exchanged using the Service Location Protocol. A directory agent has a particular scope, and may have knowledge about schemes defined by a particular name authority.

Option format: IP address

79 **Service Scope** This option is specified only at the DHCP server. This

extension indicates a scope that should be used by a service agent, when responding to Service Request messages as specified by the Service Location Protocol.

Option format: String

80 **Naming Authority** This option is specified only at the DHCP server. This extension indicates a naming authority, which specifies the syntax for schemes that may be used in URLs for use by entities with the Service Location Protocol.

Option format: String

IBM-specific Options

IBM provides a set of IBM-specific options by defining options within the user-defined range (128-254). These options are used instead of defining a vendor option (option 43) for IBM. It is recommended that you do not redefine these options.

192 **TXT RR** If this option is specified at the DHCP server, the DHCP client user is required to complete the system administrator information fields. Note: This option is only supported by TCP/IP Version 4.1 for OS/2 clients. This option provides up to four required text labels or entry fields the system administrator can specify, such as the name of a user, the user's phone number, or other fields that the DDNS Client configuration program prompt the user for. These fields allow the system administrator to identify the actual person who configured the host name or other data. The DDNS configuration program does not display these fields unless the system administrator specifies them. This information is stored in a text record in the DNS. The pairs of field labels and data are required to fit within a single TXT resource record. The space available is divided evenly between the pairs. The value is also updated in file DDNSCLI.CFG on the Dynamic-Address client.

Option format: String

Vendor Options

The DHCP protocol provides a way to supply vendor-specific information to a DHCP client using RFC-architected options 43 and 60.

60 **Option 60** is configured at a DHCP client and sent to the DHCP server to identify the client as one from a specific vendor.

43 **Option 43** is configured at the DHCP server to define the vendor-specific information to be returned to the client in response to the client's option 60 request. For the Common Code DHCP server, option 43 is configured using the add vendor-option command. A vendor-option is only defined within the global scope. The vendor option consists of the name of the vendor and the option data. The option data has two formats:

Hex data
This is entered with the vendor name when the add vendor-option command is issued. The hex data must be entered as a hex string with blanks between the bytes: "01 AA 55"

Options
Any DHCP option can be added to a vendor-option scope by the add option command.

Note: Hex data and options are mutually exclusive in a vendor definition. You can define one or the other, but not both.

Configuring IP for DHCP

In order for the DHCP server to successfully assign IP addresses and configuration information for clients on an added subnet, IP will have to be configured appropriately. This will occur when the DHCP server is directly connected to a subnet that it is configured to support.

If a BOOTP relay agent is being used to forward DHCP request messages to this DHCP server, there may not be any required IP configuration to support a subnet that is not directly connected to the server.

Adding an IP Address

An IP address which falls within the DHCP configured subnet will need to be added to the connecting interface. If there are multiple addresses defined on that interface, the address added for DHCP must be the **last** one added. IP will only present a broadcast DHCP DISCOVER message to the server as if it came in on the first address found for that interface.

Note: The first address found on an interface is the **last** address that was added to the interface.

Example:

- DHCP has added a subnet as follows:

```
DHCP Server config>list subnet all
subnet      subnet          subnet          starting        ending
name        address         mask            IP Addr         IP Addr
-------------------------------------------------------------------------
net-one     192.168.8.0     255.255.255.0   192.168.8.2     192.168.8.50
```

- IP will require the following:

```
IP config>add address
Which net is this address for [0]? 0
New address []? 192.168.8.1
Address mask [255.255.255.0]?

IP config>list add
IP addresses for each interface:
intf     0   192.168.8.1      255.255.255.0    Local wire broadcast, fill 1
intf     1                                     IP disabled on this interface
intf     2   0.0.0.2          255.255.255.255  Local wire broadcast, fill 1
intf     3                                     IP disabled on this interface
```

Using IP Simple-Internet-Access

If Simple-Internet-Access is enabled in IP and DHCP has not previously been configured, the following configuration will be automatically generated in the DHCP server. Simple-Internet-Access will also automatically configure the NAT feature and other IP filters and access controls. If DHCP is already configured there will be no changes/additions to the DHCP configuration. Refer to Using Simple Internet Access in the “Using IP” chapter in *Protocol Configuration and Monitoring Reference Volume 1* for more information and restrictions.

- IP has been configured as follows:

```
IP config>enable simple-internet-access
Interface to Service Provider [0]? 3
SIMPLE-INTERNET-ACCESS enabled on interface 3

IP config>add address
Which net is this address for [0]? 0
New address []? 192.168.8.1
Address mask [255.255.255.0]?

IP config>list add
IP addresses for each interface:
intf     0   192.168.8.1      255.255.255.0    Local wire broadcast, fill 1
intf     1                                     IP disabled on this interface
intf     2                                     IP disabled on this interface
intf     3   0.0.0.3          255.255.255.255  Local wire broadcast, fill 1
                                               SIMPLE-INTERNET-ACCESS Enabled
```

- DHCP server will have the following configuration generated:

```
DHCP Server config> list global
.
.
DHCP Server enabled: Yes
.
.
DHCP Server config>list subnet all
subnet      subnet           subnet           starting         ending
name        address          mask             IP Addr          IP Addr
-------------------------------------------------------------------------
simple-net  192.168.8.0      255.255.255.0    192.168.8.2      192.168.8.50

DHCP Server config>list option subnet
Enter the subnet name []? simple-net
option  option
code    data
-----------------------------------------------------------------
 1      255.255.255.0
 3       192.168.8.1
 6       0.0.0.3
```

Sample DHCP Server Configuration

ASCII Text File

This section provides a typical DHCP server configuration in an ASCII text format. This example is strictly for the purpose of illustration, to show a configuration in a format that may be familiar to you. The IBM 2210 does not support ASCII configurations.

You can use the blocked numbers (**1**) to relate the functions described in this ASCII example to the equivalent `talk 6` configuration shown in “OPCON (Talk 6) Configuration” on page 408.

```
1  Configuration of Server parameters

leaseTimeDefault          120                 # 120 minutes
leaseExpireInterval       20 seconds
supportBOOTP              yes
supportUnlistedClients  yes


2  Global options.  Passed to every client unless overridden at a lower scope.

option 15       "raleigh.ibm.com"         # domain name
```

```
option 6         9.67.1.5                          # dns server

class manager
{
  option 48   6.5.4.3
  option 9    9.37.35.146
  option 210  "manager_authority"   # site specific option given to all managers
}

3 Vendor-options


vendor  XI-clients hex"01 02 03"

vendor XA-clients
{
   option 23 100   # IP TTL
}

4 A typical subnet

subnet  9.2.23.0  255.255.255.0        9.2.23.120-9.2.23.126
{
        option 28       9.2.23.127         # broadcast address
        option 9   5.6.7.8
        option 51  200

5 class manager defined at the subnet scope.  Option 9 here will override
the option 9 specified in the global manager class.
                 class manager

        {
                option 9   9.2.23.98
        }

6 Programmers have their own subnet range

        class developers  9.2.23.125-9.2.23.126
        {
                option 51       -1                # infinite lease.
                option 9        9.37.35.1         # printer used by the developers
        }
}

7 Example of a client that will accept any address but will have
its own set of options.

client 6        0x10005aa4b9ab  ANY
{
        option 51 999
        option 1 255.255.255.0
}

8 Exclude an address from service.

client 0       0               9.2.23.121
```

OPCON (Talk 6) Configuration

The following is an example of the same configuration using `talk 6`.

```
1 Configuration of Server parameters

Config>f dhcp-server
DHCP server user configuration
```

```
DHCP Server config> enable dhcp
DHCP Server config>

DHCP Server config> set lease-time-default hours 2
DHCP Server config>set lease-expire-interval seconds 20
DHCP Server config>set support-bootp yes
DHCP Server config>set support-unlisted-clients global yes

DHCP Server config>li glob
DHCP server Global Parameters
=============================

DHCP server enabled: Yes

Balance:  No subnet groups defined

Inorder:  No subnet groups defined

Canonical: No

Lease Expire Interval:  20 second(s)
Lease Time Default:     2 hour(s)


Support BOOTP Clients: Yes
Bootstrap Server: Not configured

Support Unlisted Clients: Yes

Ping Time:  1 second(s)
Used IP Address Expire Interval:  15 minute(s)
```

2 Global options. Passed to every client unless overridden at a lower scope.

```
DHCP Server config>add option global 15 raleigh.ibm.com
DHCP Server config>add option global 6 9.67.1.5

DHCP Server config>li option global
option  option
code    data
----------------------------------------------------------------
15      raleigh.ibm.com
6       9.67.1.5
DHCP Server config>add class global
Enter the class name []? manager
Class record with name manager has been added

DHCP Server config>add option class-global
Enter the class name []? manager
Enter the option code [1]? 48
Enter the option data []? 6.5.4.3

DHCP Server config>add option class-global 9 9.37.35.146
DHCP Server config>add option class-global manager 210 manager_authority

DHCP Server config>li class global manager
class
name
----------------------------------------------------------------
manager

 Number of Options: 3
 option  option
 code    data
```

```
 ---------------------------------------------------------------
 48       6.5.4.3
 9        9.37.35.146
 210     manager_authority
```

3 Vendor-options

```
DHCP Server config>add vendor-option XI-client
Enter the vendor hex data []? 01 02 03
Vendor-option record with name XI-client has been added

DHCP Server config> add vendor-option XA-client
Enter the vendor hex data []?
Vendor-option record with name XA-client has been added
DHCP Server config> add option vendor-option XA-client 23 100

DHCP Server config>li vendor-option all
vendor      hex
name        data
---------------------------------------------------------------
XI-client   01 02 03
XA-client
DHCP Server config>li vendor-option det XA-client
vendor      hex
name        data
---------------------------------------------------------------
XA-client

 Number of Options: 1
 option  option
 code    data
 ---------------------------------------------------------------
 23      100
```

4 A typical subnet

```
DHCP Server config>add subnet
Enter the subnet name []? sub1
Enter the IP subnet []? 9.2.23.0
Enter the IP subnet mask [255.255.255.0]?
Enter start of IP address range [9.2.23.1]? 9.2.23.120
Enter end of IP address range [9.2.23.150]? 9.2.23.126
Enter the subnet group name []?
Subnet record with name sub1 has been added
DHCP Server config>
DHCP Server config> add option subnet
Enter the subnet name []? sub1
Enter the option code []? 28
Enter the option data []? 9.2.23.127
DHCP Server config> add option subnet 9 5.6.7.8
DHCP Server config>add option subnet sub1 51 200

DHCP Server config>add class subnet
Enter the subnet name []? sub1
Enter the class name []? manager
Enter start of IP address range []?
Class record with name manager has been added

DHCP Server config>add option class-subnet sub1 manager
Enter the option code [1]? 9
Enter the option data []? 9.2.23.98
```

6 Programmers have their own subnet range

```
DHCP Server config>add class subnet
```

```
Enter the subnet name []? sub1
Enter the class name []? developers
Enter start of IP address range []? 9.2.23.125
Enter end of IP address range []? 9.2.23.126
Class record with name developers has been added

DHCP Server config>add option class-subnet sub1 developers 51 -1
DHCP Server config>add option class-subnet sub1 developers 9 9.37.35.1


DHCP Server config>li subnet detailed sub1
subnet      subnet          subnet          starting        ending
name        address         mask            IP Addr         IP Addr
-------------------------------------------------------------------------
sub1        9.2.23.0        255.255.255.0   9.2.23.120      9.2.23.126


 Number of Classes: 2
 class
 name
 ---------------------------------------------------------------
 manager

  Number of Options: 1
  option  option
  code    data
  ---------------------------------------------------------------
  9       9.2.23.98
 developers
  starting IP address: 9.2.23.125
  ending   IP address: 9.2.23.126

  Number of Options: 2
  option  option
  code    data
  ---------------------------------------------------------------
  51      -1
  9       9.37.35.1

 Number of Options: 3
 option  option
 code    data
 ---------------------------------------------------------------
 28      9.2.23.127
 9       5.6.7.8
 51      200
```

7 Example of a client that will accept any address but will have its own set of

```
DHCP Server config>add client global
Enter the client name []? any-addr
Enter the client's hardware type (0 - 21) [1]? 6
Enter the client ID (MAC address or string) []? 10005aa4b9ab
Enter the client's IP address (IP address, any, none) []? any

DHCP Server config>add option client-global any-addr 51 999
DHCP Server config>add option client-global any-addr 1 255.255.255.0
```

8 Exclude an address from service.

```
Enter the client name []? excl-addr
Enter the client's hardware type (0 - 21) [1]? 0
Enter the client ID (MAC address or string) []? 0
Enter the client's IP address (IP address, any, none) []? 9.2.23.121

 DHCP Server config>li cli all
```

```
client      client  client                          attached    IP
name        type    identifier                      to subnet   address
-------------------------------------------------------------------------------
any-addr    6       10005aa4b9ab                                Any
excl-addr   0       0                                           9.2.23.121


DHCP Server config>li client global any-addr
client      client  client                          IP
name        type    identifier                      address
-------------------------------------------------------------------
any-addr    6       10005aa4b9ab                    Any

 Number of Options: 2
 option  option
 code    data
 ----------------------------------------------------------------
 51      999
 1       255.255.255.0
```

Chapter 29. Configuring and Monitoring DHCP Server

This chapter describes how to use the DHCP server configuration and operating commands and includes the following sections:

- "Accessing the DHCP Server Configuration Environment"
- "DHCP Server Configuration Commands"
- "Accessing the DHCP Server Monitoring Environment" on page 441
- "DHCP Server Monitoring Commands" on page 442

Accessing the DHCP Server Configuration Environment

Use the following procedure to access the DHCP server *configuration* process.

1. At the OPCON prompt, enter **talk 6**. For example:

```
* talk 6
Config>
```

   After you enter the **talk 6** command, the Config prompt (`Config>`) displays on the terminal. If the prompt does not appear when you first enter configuration, press **Return** again.

2. At the Config prompt, enter the **feature dhcp-server** command to get to the `DHCP Server config>` prompt.

DHCP Server Configuration Commands

Table 59. DHCP Server Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page xxix.
Add	Adds a class, client, subnet, or vendor-option.
Change	Changes the definition of a class, client, subnet, or vendor-option.
Default	Returns certain global variables to their default values.
Delete	Deletes a class, subnet, or vendor-option.
Disable	Disables DHCP Server globally.
Enable	Enables DHCP Server globally.
List	Lists definitions of a class, client, globals, subnet, or vendor-option.
Set	Sets definitions for global parameters or options under a specified scope.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page xxix.

Add

Use the **add** command to add a class, subnet or vendor-option.

Syntax:

add class
client
option

subnet

vendor-option

class *scope [subnet_name] class_name [range_start] [range_end]*

Defines a class.

scope Specifies the scope in which the class is being added.

Valid Values: global or subnet

Default Value: None

subnet_name

This is valid only if the **scope** is *subnet*. Indicates the name of the subnet to which the class is being added.

Valid Values: Any existing subnet name

Default Value: None

class-name

Indicates the name of the class.

Valid Values: An ASCII string up to 40 characters in length

Default Value: None

range-start

This is valid only if the **scope** is *subnet*. Specifies the starting IP address for the IP address pool to which clients will be assigned.

Valid Values: Any valid IP address within the range of the subnet to which the class is being added.

Default Value: The first IP address of the subnet range belonging to the specified subnet.

range-end

This is valid only if the **scope** is *subnet*. Specifies the ending IP address for the IP address pool to which clients will be assigned.

Valid Values: Any valid IP address within the range of the subnet to which the class is being added. This value must be greater than the value specified for **range-start**.

Default Value: The starting IP address plus 5 of the subnet range belonging to the specified subnet. If the resulting IP address is no longer within the subnet range, then the default is the ending IP address of the subnet range.

Example:

```
DHCP Server config> add class global
Enter class name? ClassA

DHCP Server config> add class subnet
Enter the subnet name[]? subA
Enter class name[]? ClaA
Enter start of IP address range[10.1.1.1]?
Enter end of IP address range[10.1.1.6]?
```

client *scope [subnet_name] client_name id-type id-value address*

Defines a client

scope Specifies the scope in which the client is being added.

Valid Values: global or subnet

Default Value: None

subnet-name
Valid only if the **scope** is *subnet*. Specifies the name of the subnet to which the client is being added.

Valid Values: Any existing subnet name

Default Value: None

client-name
Indicates the name of the client.

Valid Values: Any 10-character ASCII string

Default Value: None

id-type
Indicates the hardware type of the client. Hardware types defined in RFC 1340 that are applicable to the IBM 2210 are shown below as valid values.

Valid Values:

0 Unspecified. Indicates a symbolic name for the client.

1 Ethernet

6 IEEE 802 networks (including 802.5 Token Ring)

Default Value: 1

id-value
Specifies the client identifier. If the **id-type** is *0*, then the **id-value** is a 64-character string. Otherwise, the **id-value** is a MAC address.

Note: An **id-type** of *0* and an **id-value** of *0* indicates that the specified IP address should not be distributed by the server.

Valid Values: 0 or any valid MAC address (12 hexadecimal digits)

Default Value: None

address
Specifies either the IP address to be supplied to the client or a character string indicating that the client will not be serviced or that the client can be supplied with any address from the IP address pool.

Valid Values:

Any valid IP address
In dotted decimal format. If the client is defined within a subnet scope, the IP address must be within the subnet range.

none Indicates that the matching client will not be serviced

any Indicates that any IP address in the subnet pool can be supplied to the client.

Default Value: None

Note: An **id-type** of *0* and an **id-value** of *0* indicates that the specified IP address should not be distributed by the server.

Example:

```
DHCP Server config> add client global
     Enter the client name []? ClientA
     Enter the client's hardware type (0 - 21) [1]? 0
     Enter the client ID (MAC address or string) []? ClientA
     Enter the client's IP address (IP address, any, none) []? 9.1.1.1
     Client record with name ClientA has been added

DHCP Server config> add client subnet
     Enter the subnet name []? subA
     Enter the client name []? CliA
     Enter the client's hardware type (0 - 21) [1]? 1
     Enter the client ID (MAC address or string) []? 400000000010
     Enter the client's IP address (IP address, any, none) []? 10.1.1.10
     Client record with name CliA has been added
```

option *scope [subnet-name] [class-name] [client-name] [vendor-name] code data*
Defines an option. Options can exist globally, or within a subnet, class, client, or vendor-option scope.

scope Specifies the scope in which the option is being added.

Valid Values:

- class-global
- class-subnet
- client-global
- client subnet
- global
- subnet
- vendor-option

Default Value: None

subnet-name
Valid only if the **scope** is *subnet , class-subnet,* or *client-subnet*. Specifies the name of the subnet to which the client is being added.

Valid Values: Any existing subnet name

Default Value: None

class-name
Valid only if the **scope** is *class-global* or *class-subnet*. Indicates the name of the class to which the option is being added.

Valid Values: An existing class name

Default Value: None

client-name
Valid only if the **scope** is *client-global* or *client-subnet*. Indicates the name of the client to which the option is being added.

Valid Values: Any existing client name

Default Value: None

vendor-name
Valid only if the **scope** is *vendor-option*. Indicates the name of the vendor to which the option is being added.

Valid Values: Any existing vendor name

Default Value: None

code Specifies the option code. The DHCP options are defined in RFC 2132. See "DHCP Options" on page 394 for a description of options and their formats.

Valid Values: 1 - 255

Default Value: 1

data Specifies the option data. Option data can be defined in three ways.

- ASCII strings for specific formats defined in RFC 2132.
- Hexadecimal conversion at initialization time. The data should be entered as *hex: 01 aa 04*.
- Character string. The data should be entered as *abcdef*.

Example:

```
DHCP Server config> add option global
     Enter the option code [1]? 3
     Enter the option data []? 9.167.100.1
```

Example:

```
DHCP Server config> add option subnet
     Enter the subnet name []? subA
     Enter the option code [1]? 3
     Enter the option data []? 9.167.100.1
```

Example:

```
DHCP Server config> add option class-global
     Enter the class name []? ClassA
     Enter the option code [1]? 3
     Enter the option data []? 9.167.100.1
```

Example:

```
DHCP Server config> add option client
     Enter the client name []? ClientA
     Enter the option code [1]? 3
     Enter the option data []? 9.167.100.1
```

Example:

```
DHCP Server config> add option class-subnet
     Enter the subnet name []? subA
     Enter the class name []? ClaA
     Enter the option code [1]? 3
     Enter the option data []? 9.167.100.1
```

Example:

```
DHCP Server config> add option client-subnet
     Enter the subnet name []? subA
     Enter the client name []? CliA
     Enter the option code [1]? 3
     Enter the option data []? 9.167.100.1
```

Example:

```
DHCP Server config> add option vendor-option
     Enter the vendor name []? 200
     Enter the option code [1]? 85
     Enter the option data []? hex:01 AA 04
```

Example:

```
DHCP Server config> add option vendor-option
```

```
Enter the vendor name []? 200
Enter the option code [1]? 86
Enter the option data []? 9.67.85.4
```

subnet *subnet_name subnet-address subnet-mask range-start range-end [subnet_group_name] [subnet_group_priority] [policy-list]*

Defines a subnet.

subnet-name

Indicates the name of the subnet.

Valid Values: Any 10-character ASCII string

Default Value: None

subnet-address

Specifies the address of the subnet. The address is specified in dotted decimal format.

Valid Values: Any valid IP subnet address

Default Value: None

subnet-mask

Specifies the subnet address mask. The subnet address must be within the subnet mask and cannot contain a larger number of bits than the mask.

Valid Values: Any valid IP mask in dotted decimal format

Default Value: Calculated based upon the subnet address

range-start

Specifies the starting IP address of the IP pool of addresses that this server will administer for this subnet. If *range-start* is not specified, then all the addresses in the subnet are administered by the server.

Valid Values: Any valid IP host address within the specified subnet in dotted decimal format

Default Value: The first IP address of the subnet

range-end

Specifies the ending IP address of the IP pool of addresses that this server will administer for this subnet.

Valid Values: Any valid IP host address within the specified subnet in dotted decimal format

Default Value: range-start plus 50. If the resulting IP address is no longer within the subnet, then the default is the last IP address in the subnet.

subnet-group-name

Specifies the subnet group name to which this subnet belongs.

Valid Values: Any ASCII string up to 64 characters in length

Default Value: None

subnet-group-priority

Specifies this subnet's priority within the subnet group. This priority is used to determine the order in which the addresses are assigned within a specific subnet group.

Valid Values: 1 - 65535

Default Value: 1

policy-list

Identifies which policy address list, Balance or Inorder, to which the subnet group will be added. If the subnet group already exists on one list and the other is specified, the subnet group will be moved to the new list.

Valid Values: Inorder or Balance

Default Value: If this is a new subnet, the default is Inorder. Otherwise, it is the current policy list to which the subnet group belongs.

Example:

```
DHCP Server config> add subnet
 Enter the subnet name []? subA
 Enter the IP subnet []? 10.1.1.0
 Enter the IP subnet mask [255.255.255.0]?
 Enter start of IP address range [10.1.1.1]?
 Enter end of IP address range [10.1.1.31]?
 Enter the subnet group name []? group1
 Enter the subnet group priority (1 - 65535) [1]?
 Enter the access policy list (Inorder or Balance) [Inorder]?
 Subnet record with name sub1 has been added
 Subnet group group1 is being added to the Inorder List
```

vendor-option *vendor_name [hex_value]*

Adds a vendor-option. There are two ways to provide vendor-option data:

- Enter hex data when prompted
- Add specific options to the vendor using the **add option vendor** command. See page 416 for option information.

vendor_name

Specifies the name of the vendor.

Valid Values: An ASCII string up to 40 characters in length

Default Value: None

hex-value

Specifies the hexadecimal ASCII string which represents the hexadecimal value of the data portion of the option.

Valid Values: Any valid hexadecimal string in the following format: *01 aa 04*

Default Value: None

Example:

```
DHCP Server config> add vendor-option
     Enter the vendor name []? XA-client
     Enter the vendor hex data [] 01 aa 04?
     Vendor-option record with name XA-client has been added
```

Change

Use the **change** command to modify the configuration of a class, client, subnet or vendor-option.

DHCP Server Configuration Commands (Talk 6)

Syntax:

change class
client
subnet
vendor-option

class *scope [subnet_name] class_name new_class_name [new_range_start] [new_range_end]*

Modifies a class.

scope Specifies the scope of the class being modified.

Valid Values: global or subnet

Default Value: None

subnet-name

Valid only if the **scope** is *subnet*. Indicates the name of the subnet to which the class belongs.

Valid Values: Any existing subnet name.

Default Value: None

class-name

Indicates the name of the class.

Valid Values: Name of an existing class

Default Value: None

new-class-name

Indicates the new name of the class.

Valid Values: An ASCII string up to 40 characters in length

Default Value: Existing class name

new-range-start

Valid only if the **scope** is *subnet*. Specifies the new starting IP address for the IP address pool to which clients will be assigned.

Valid Values: Any IP address within the subnet range

Default Value: Existing range-start

new-range-end

Specifies the new ending IP address for the IP address pool to which clients will be assigned.

Valid Values: Any valid IP address within the subnet range, greater than **new-range-end**

Default Value: Existing range-end

Example:

```
DHCP Server config> change class global
     Enter the class name []? ClassA
     Enter the new class name [ClassA]?
```

Example:

```
DHCP Server config> change class subnet
     Enter the subnet name []? subA
     Enter the class name []? ClaA
```

```
Enter the new class name [ClaA]?
Enter start of IP address range [10.1.1.1]?
Enter end of IP address range [10.1.1.6]?
```

client *scope [subnet_name] client_name new-client_name new-id-type new-id-value new-address*

Modifies a client

scope Specifies the scope of the client being modified.

Valid Values: global or subnet

Default Value: None

subnet-name

Valid only if the **scope** is *subnet*. Indicates the name of the subnet to which the client belongs.

Valid Values: Any existing subnet name

Default Value: None

client-name

Indicates the name of the client.

Valid Values: An existing client name

Default Value: None

new-client-name

Indicates the new name of the client.

Valid Values: An ASCII string up to 10 characters in length

Default Value: Existing client name

new-id-type

Indicates the new hardware type of the client.

Valid Values: 0 - 21 See page 415.

Default Value: Existing hardware type of the client

new-id-value

Specifies the new client identifier.

Valid Values: 0 or any valid MAC address (12 hexadecimal digits)

Default Value: Existing client id-type

Note: An **id-type** of *0* and an **id-value** of *0* indicates that the specified IP address should not be distributed by the server.

new-address

Specifies either the new IP address to be supplied to the client or a character string indicating that the client will not be serviced or that the client can be supplied with any address from the IP address pool.

Valid Values:

Any valid IP address

none Indicates that the matching client will not be serviced

any Indicates that any IP address in the subnet pool can be supplied to the client.

Default Value: None

Note: An **id-type** of *0* and an **id-value** of *0* indicates that the specified IP address should not be distributed by the server.

Example:

```
DHCP Server config> change client global
     Enter the client name []? ClientA
     Enter the new client name [ClientA]?
     Enter the new client hardware type (0 - 21) [0]?
     Enter the new client ID  [ClientA]?
     Enter the client's new IP address (IP address, any, none) [9.1.1.1]?
     Client ClientA has been changed
```

Example:

```
DHCP Server config> change client subnet
     Enter the subnet name []? subA
     Enter the client name []? CliA
     Enter the new client name [ClientA]?
     Enter the new client hardware type (0 - 21) [1]?
     Enter the new client ID  [400000000010]?
     Enter the client's new IP address (IP address, any, none) [10.1.1.10]?
     Client CliA has been changed
```

subnet *subnet_name new_subnet_name new_subnet_address new_subnet_mask new-range_start new-range_end*

Modifies a subnet.

subnet_name

Indicates the name of the specific subnet to be modified.

Valid Values: An existing subnet name

Default Value: None

new_subnet_name

Indicates the new name of the specified subnet.

Valid Values: Any 10-character ASCII string

Default Value: Original subnet name

new_subnet_addresss

Specifies the new address of the subnet. The address is specified in dotted decimal notation.

Valid Values: Any valid IP subnet address

Default Value: Existing subnet address

new_subnet_mask

Specifies the new subnet address mask. The subnet address must be within the subnet mask and cannot contain a larger number of bits than the mask.

Valid Values: Any valid IP mask

Default Value: Existing subnet mask

new-range-start

Specifies the new starting IP address of the IP pool of addresses that this server will administer for this subnet. If *range-start* is not specified, then all the addresses in the subnet are administered by the server.

Valid Values: Any valid IP address within the subnet range

Default Value: Existing pool starting address

new-range-end
Specifies the new ending IP address of the IP pool of addresses that this server will administer for this subnet.

Valid Values: Any valid IP address within the subnet range and larger than the starting pool address

Default Value: Existing pool ending address

Example:

```
DHCP Server config> change subnet
     Enter the subnet name []? subA
     Enter the new subnet name [subA]?
     Enter the new IP subnet[10.1.1.0]?
     Enter the new IP subnet mask[255.255.0.0]?
     Enter new start of IP address range [10.1.1.1]?
     Enter new end of IP address range [10.1.1.31]?
     Enter the new subnet group name [group11]?
     Enter the new subnet group priority [1]?
     Enter the new access policy list (Inorder or Balance) [Inorder]?
```

vendor-option *vendor_name new_vendor_name [new_hex_value]*
Modifies a vendor-option.

vendor_name
Specifies the new name of the vendor option.

Valid Values: An existing vendor name

Default Value: None

new_vendor_name
Specifies the new name of the vendor option.

Valid Values: An ASCII string up to 40 characters in length

Default Value: Existing vendor option name

new_hex_value
Specifies the new hexadecimal ASCII string which represents the hexadecimal value of the data portion of the option. A hex value cannot be added if specific options have been added to this vendor option.

Valid Values: Any valid hexadecimal string

Default Value: Existing hexadecimal string

Example:

```
DHCP Server config> change vendor-option
     Enter the vendor name []? XA-clients
     Enter the new vendor name [XA-clients]?
     Enter the new vendor data [01 aa 04]?
```

Delete

Use the **delete** command to delete a class, client, option, subnet, subnet-group, or vendor-option.

Syntax:

delete class
client

option

subnet

subnet-group

vendor-option

class *scope [subnet_name] class_name*

Deletes a class and all options defined under its scope.

scope Specifies the scope in which the class is being deleted.

Valid Values: global or subnet

Default Value: None

subnet-name

Only valid if the **scope** is *subnet*. Specifies the name of the subnet that the class is being deleted from.

Valid Values: Any existing subnet name

Default Value: None

class-name

Indicates the name of the class to be deleted.

Valid Values: An existing class name

Default Value: None

Example:

```
DHCP Server config> delete class global
     Enter the class name []? ClassA
```

Example:

```
DHCP Server config> delete class subnet
     Enter the subnet name []? subA
     Enter the class name []? ClaA
```

client *scope [subnet_name] client_name*

Deletes a client and all options defined under its scope.

scope Specifies the scope in which the client is being deleted.

Valid Values: global or subnet

Default Value: None

subnet_name

Only valid if the **scope** is *subnet*. Specifies the name of the subnet that the client is being deleted from.

Valid Values: An existing subnet name

Default Value: None

client_name

Indicates the name of the client to be deleted.

Valid Values: An existing client name

Default Value: None

Example:

```
DHCP Server config> delete client global
     Enter the client name []? ClientA

Example:

DHCP Server config> delete client subnet
     Enter the subnet name []? subA
     Enter the client name []? CliA
```

option *scope [subnet_name] [class_name] [client_name] [vendor_name] code*
Deletes an option within the specified scope.

scope Specifies the scope in which the option is being deleted.

Valid Values:

- class-global
- class-subnet
- client-global
- client subnet
- global
- subnet
- vendor-option

Default Value: None

subnet-name
Valid only if the **scope** is *subnet , class-subnet,* or *client-subnet*. Specifies the name of the subnet from which the client is being deleted.

Valid Values: Any existing subnet name

Default Value: None

class-name
Valid only if the **scope** is *class-global* or *class-subnet*. Indicates the name of the class from which the option is being deleted.

Valid Values: An existing class name

Default Value: None

client-name
Valid only if the **scope** is *client-global* or *client-subnet*. Indicates the name of the client from which the option is being deleted.

Valid Values: Any existing client name

Default Value: None

vendor-name
Valid only if the **scope** is *vendor-option*. Indicates the name of the vendor from which the option is being deleted.

Valid Values: Any existing vendor name

Default Value: None

code Specifies the option code. The DHCP options are defined in RFC 2132. See “DHCP Options” on page 394 for a description of options and their formats.

Valid Values: 1 - 255

Default Value: 1

Example:

```
DHCP Server config> delete option global
     Enter the option code [1]? 3
```

Example:

```
DHCP Server config> delete option subnet
     Enter the subnet name []? subA
     Enter the option code [1]? 3
```

Example:

```
DHCP Server config> delete option class-global
     Enter the class name []? ClassA
     Enter the option code [1]? 3
```

Example:

```
DHCP Server config> delete option client
     Enter the client name []? ClientA
     Enter the option code [1]? 3
```

Example:

```
DHCP Server config> delete option class-subnet
     Enter the subnet name []? subA
     Enter the class name []? ClaA
     Enter the option code [1]? 3
```

Example:

```
DHCP Server config> delete option client-subnet
     Enter the subnet name []? subA
     Enter the client name []? CliA
     Enter the option code [1]? 3
```

Example:

```
DHCP Server config> delete option vendor-option
     Enter the vendor name []? XI-clients
     Enter the option code [1]? 85
```

Example:

```
DHCP Server config> delete option vendor-option
     Enter the vendor name []? 200
     Enter the option code [1]? 86
```

subnet *subnet_name*

Deletes a subnet and all classes, clients, and options that are defined under its scope.

subnet_name

Specifies the name of the subnet being deleted.

Valid Values: Any existing subnet name

Default Value: None

Example:

```
DHCP Server config> delete subnet
     Enter the subnet name []? subA
     You are about to delete a subnet subA
     and all the associated class, client, and option records associated with it
     Are you sure you want to continue? [No]:
```

subnet-group *subnet_group_name*

Deletes all subnets associated with a particular subnet group and all the classes, clients and options defined under the subnet scopes.

subnet_group_name
Specifies the name that identifies the subnet group.

Valid Values: An existing subnet group name

Default Value: None

Example:

```
DHCP Server config> delete subnet-group
     Enter the subnet group name []? group2
     You are about to delete a all subnets in group group2
     and all the associated class, client, and option records associated with them
     Are you sure you want to continue? [No]:
```

vendor-option *vendor_name*
Deletes a vendor-option and any options defined under its scope.

vendor_name
Specifies the name of the vendor.

Valid Values: An ASCII string up to 40 characters in length

Default Value: None

Example:

```
DHCP Server config> delete vendor-option
     Enter the vendor name []? XA-clients
```

Disable

Use the **disable** command to disable DHCP server globally.

Syntax:

disable dhcp-server

Example:

```
DHCP Server config> disable dhcp-server
```

Enable

Use the **enable** command to enable DHCP server globally.

Syntax:

enable dhcp-server

Example:

```
DHCP Server config> enable dhcp-server
```

List

Use the **list** command to list configuration information about a class, client, global parameters, subnets or vendor-options and any associated options.

DHCP Server Configuration Commands (Talk 6)

Syntax:

list class
client
global
option
subnet
vendor-option

class all
global *class-name*
subnet *class-name*

Lists either a summary of all the configured classes or the details of a specific class.

class-name
Indicates the name of the class to be displayed.

Valid Values: An existing class name

Default Value: None

Example:

```
DHCP Server config> list class all

     class              attached
     name               to subnet
     ------------------------------------------------------------
     ClassA
     ClaA               subA
```

Example:

```
DHCP Server config> list class global
     Enter the class name []? ClassA

     class
     name
     ------------------------------------------------------------
     ClassA
     Bootstrap Server: 100.100.100.100
     Canonical: Yes
     Support Unlisted Clients: Yes
     Number of Options: 1
     option    option
     code      data
     ------------------------------------------------------------
     1         255.255.0.0
```

Example:

```
DHCP Server config> list class subnet
     Enter the subnet name []? subA
     Enter the class name []? ClaA

     class
     name
     ------------------------------------------------------------
     ClaA
```

```
     starting IP address: 10.1.1.3
     ending IP address: 10.1.1.5
     Bootstrap Server: 100.100.100.100
     Canonical: Yes
     Support Unlisted Clients: DHCP

     Number of Options: 1
     option     option
     code       data
     ----------------------------------------------------------------
     6          9.67.100.1
```

client all

global *client-name*

subnet *client-name*

Lists either a summary of all the configured clients or the details of a specific client.

client-name

Indicates the name of the client to be displayed.

Valid Values: An existing client name

Default Value: None

Example:

```
DHCP Server config> list client all
     client   client   client         attached    IP
     name     type     identifier     to subnet   address
     ------------------------------------------------------------------------------
     ClientA   0       ClientA                    9.1.1.1

     CliA      1       400000000010    subA       10.1.1.10
```

Example:

```
DHCP Server config> list client global
     Enter the client name []? ClientA
```

Example:

```
DHCP Server config> list client subnet
     Enter the subnet name []? subA
     Enter the client name []? CliA

     client client    client        IP
     name   type      identifier    address
     ---------------------------------------------------------------
     CliA   1         400000000010  10.1.1.10
      Bootstrap Server: 200.200.200.200
      Canonical: Yes

     Number of Options: 1
     option   option
     code     data
     ----------------------------------------------------------------
     6        9.67.100.1
```

global

Lists global parameters.

Example:

```
DHCP Server config> list global

    DHCP server Global Parameters
    =============================
    DHCP server enabled: Yes

    Balance: group2

    Inorder: group1

    Canonical: No

    Lease Expire Interval: 1 minute(s)
    Lease Time Default: 1 day(s)

    Support BOOTP Clients: No
    Bootstrap Server: Not configured

    Support Unlisted Clients: Yes
    Ping Time: 1 second(s)
    Used IP Address Expire Interval: 15 minute(s)
```

option *scope [subnet-name] [class-name] [client-name] [vendor-name] code*

scope Specifies the scope in which the option is being listed.

Valid Values:

- class-global
- class-subnet
- client-global
- client subnet
- global
- subnet
- vendor-option

Default Value: None

subnet-name
Valid only if the **scope** is *subnet*, *class-subnet*, or *client-subnet*. Specifies the name of the subnet to which the option being listed belongs.

Valid Values: Any existing subnet name

Default Value: None

class-name
Valid only if the **scope** is *class-global* or *class-subnet*. Indicates the name of the class to which the option being listed belongs.

Valid Values: An existing class name

Default Value: None

client-name
Valid only if the **scope** is *client-global* or *client-subnet*. Indicates the name of the client to which the option being listed belongs.

Valid Values: Any existing client name

Default Value: None

vendor-name
Valid only if the **scope** is *vendor-option*. Indicates the name of the vendor to which the option being listed belongs.

Valid Values: Any existing vendor name

Default Value: None

code Specifies the option code. The DHCP options are defined in RFC 2132. See “DHCP Options” on page 394 for a description of options and their formats.

Valid Values: 1 - 255

Default Value: 1

Example:

```
DHCP Server config> list option global

     option   option
     code     data
     ---------------------------------------------------------------
     3        9.67.100.1
```

Example:

```
DHCP Server config> list option class-global
     Enter the class name []? ClassA
     option   option
     code     data
     ---------------------------------------------------------------
     3        9.67.100.1
```

Example:

```
DHCP Server config> list option class-subnet
     Enter the subnet name []? subA
     Enter the class name []? claA

     option   option
     code     data
     ---------------------------------------------------------------
     3        9.67.100.1
```

Example:

```
DHCP Server config> list option client-global
     Enter the client name []? ClientA
     option   option
     code     data
     ---------------------------------------------------------------
     3        9.67.100.1
```

Example:

```
DHCP Server config> list option client-subnet
     Enter the subnet name []? subA
     Enter the client name []? cliA
```

```
     option   option
     code     data
     ----------------------------------------------------------------
     3        9.67.100.1
```

Example:

```
DHCP Server config> list option subnet
      Enter the subnet name []? subA


     option   option
     code     data
     ----------------------------------------------------------------
     6        9.67.100.1
```

Example:

```
DHCP Server config> list option vendor-option
      Enter the vendor name []? XI-clients


     option   option
     code     data
     ----------------------------------------------------------------
     85        hex:01 aa 04
     86        9.67.85.4
```

subnet

all

detailed *subnet-name*

Lists either a summary of all the configured subnets or the details of a specific subnet.

subnet-name

Indicates the name of the subnet to be displayed.

Valid Values: An existing subnet name

Default Value: None

Example:

```
DHCP Server config> list subnet all

     name    address    mask          IP Addr    IP Addr
     ------------------------------------------------------------------------
     subA    10.1.1.0   255.255.0.0   10.1.1.1   10.1.1.31
     subB    11.1.1.0   255.255.0.0   11.1.1.1   11.1.1.31
```

Example:

```
DHCP Server config> list subnet detailed
     Enter the subnet name []? subA

     subnet    subnet     subnet        starting      ending
     name      address    mask          IP Addr       IP Addr
     ------------------------------------------------------------------------
     subA      10.1.1.0   255.255.0.0   10.1.1.1      10.1.1.31
     Subnet Group: group1/1

     Number of Classes: 1
     class
     name
```

```
-------------------------------------------------------------
ClaA
starting IP address: 10.1.1.1
ending   IP address: 10.1.1.6
Bootstrap Server:   100.100.100.100
Canonical: Yes
Support Unlisted Clients: DHCP

Number of Options: 1
option   option
code     data
-------------------------------------------------------------
6        9.67.100.1

Number of Clients: 1
client     client     client            IP
name       type       identifier        address
-------------------------------------------------------------------
CliA       1          400000000010      10.1.1.10
Bootstrap Server: 200.200.200.200
Canonical: Yes

Number of Options: 1
option option
code   data
-------------------------------------------------------------
6      9.67.100.1

Number of Options: 1
option option
code   data
-------------------------------------------------------------
1      255.255.255.0
```

vendor-option all

detailed *vendor-name*

Lists either a summary of all the configured vendors or the details of a specific vendor-option.

vendor-name

Indicates the name of the vendor-option to be displayed.

Valid Values: An existing vendor-name

Default Value: None

Example:

```
DHCP Server config> list vendor-option all

          vendor          hex
          name            data
          ----------------------------------------------------------
          XA-clients      01 AA 04
          XI-clients

DHCP Server config> list vendor-option detailed
          Enter the vendor name []? XI-clients
          vendor          hex
          name            data
          ----------------------------------------------------------
          XI-clients

          Number of Options: 2
          option    option
```

```
code      data
----------------------------------------------------------------
85        hex:01 AA 04
86        9.67.85.4
```

Set

Use the **set** command to specify values for global parameters and to add subnet groups to the Balance and Inorder lists.

Syntax:

set balance
bootstrapserver
canonical
inorder
lease-expire-interval
lease-time-default
ping-time
support-bootp
support-unlisted-clients
used-ip-address-expire-interval

balance *subnet_group_name*
Adds or moves a subnet group to the Balance list. Addresses will be assigned in a round robin fashion from all the subnets associated with the group(s) defined within a subnet group, according to their priority.

subnet_group_name
Specifies the name of the subnet group to which this subnet belongs.

Valid Values: An existing subnet group name

Default Value: None

Example:

```
DHCP Server config> set balance
     Enter the subnet group name []? group1
```

bootstrapserver *scope [subnet-name] [class-name] [client-name] address*
Specifies whether or not the DHCP server specifies a bootstrap server for clients. If you want the DHCP server to specify a bootstrap server, you should define the IP address of the server. This parameter can be specified within the global, subnet, class or client scope.

scope Specifies the scope of the bootstrapserver parameter.

Valid Values:

- class-global
- class-subnet
- client-global
- client-subnet
- global

- subnet

Default Value: None

subnet-name
Valid if the scope is *subnet*, *class-subnet* or *client-subnet*. Indicates the name of the subnet for which the bootstrap server is being specified.

Valid Values: An existing subnet name

Default Value: None

class-name
Valid if the scope is *class-global* or *class-subnet*. Indicates the name of the class for which the bootstrap server is being specified.

Valid Values: An existing class name

Default Value: None

client-name
Valid if the scope is *client-global* or *client-subnet*. Indicates the name of the client for which the bootstrap server is being specified.

Valid Values: An existing client name

Default Value: None

IP address of the server
Specifies the IP address of the bootstrap server.

Valid Values: Any valid IP address in dotted decimal format

Default Value: None

Example:

```
DHCP Server config> set bootstrap-server class-global
     Enter the class name []? ClassA
     Enter the IP address of the server []? 100.100.100.100
```

Example:

```
DHCP Server config> set bootstrap-server class-subnet
     Enter the subnet name []? subA
     Enter the class name []? ClassA
     Enter the IP address of the server []? 100.100.100.100
```

Example:

```
DHCP Server config> set bootstrap-server client-global
     Enter the client name []? ClientA
     Enter the IP address of the server []? 100.100.100.100
```

Example:

```
DHCP Server config> set bootstrap-server client-subnet
     Enter the subnet name []? subA
     Enter the client name []? ClientA
     Enter the IP address of the server []? 100.100.100.100
```

Example:

```
DHCP Server config> set bootstrap-server global
        Enter the IP address of the server []? 100.100.100.100
```

Example:

```
DHCP Server config> set bootstrap-server subnet
     Enter the subnet name []? subA
     Enter the IP address of the server []? 100.100.100.100
```

canonical *scope [subnet-name] [class-name] [client-name] value*

Specifies whether the DHCP server will transform MAC addresses to canonical format.

MAC addresses for Ethernet/802.3 clients are stored in the canonical (byte starts with least significant bit) format. MAC addresses for Token-Ring clients are stored in the non-canonical (byte starts with most significant bit) format. This parameter should be used when the DHCP server is on one media type (Token-Ring or Ethernet/802.3), the client is on the other media type and there is a translational bridge between the two. When this parameter is set to *yes*, the DHCP server will cause the client's MAC address to be flipped from either canonical to non-canoncial or non-canonical to canonical. Since the DHCP server does not know which format the MAC address is originally in, setting this parameter to *yes* will just flip the address. Canonical can be set within the global, subnet, class or client scope.

scope Specifies the scope of the bootstrapserver parameter.

Valid Values:

- class-global
- class-subnet
- client-global
- client-subnet
- global
- subnet

Default Value: None

subnet-name

Valid if the scope is *subnet*, *class-subnet* or *client-subnet*. Indicates the name of the subnet for which canonical is being specified.

Valid Values: An existing subnet name

Default Value: None

class-name

Valid if the scope is *class-global* or *class-subnet*. Indicates the name of the class for which canonical is being specified.

Valid Values: An existing class name

Default Value: None

client-name

Valid if the scope is *client-global* or *client-subnet*. Indicates the name of the client for which canonical is being specified.

Valid Values: An existing client name

Default Value: None

value Specifies whether MAC addresses are to be transformed to canonical format

Valid Values: yes, no

Default Value: no, if the **scope** is *global*. Otherwise, the default value is determined by the scope hierarchy. See “Concepts and Terminology” on page 391 for an explanation of scope.

Example:

```
DHCP Server config> set canonical class-global
 Enter the class name []? ClassA
 Would you like MAC addresses to be transformed to canonical format? [No] yes
```

Example:

```
DHCP Server config> set canonical class-subnet
 Enter the subnet name []? subA
 Enter the class name []? ClassA
 Would you like MAC addresses to be transformed to canonical format? [No] yes
```

Example:

```
DHCP Server config> set canonical client-global
 Enter the client name []? ClientA
 Would you like MAC addresses to be transformed to canonical format? [No] yes
```

Example:

```
DHCP Server config> set canonical client-subnet
 Enter the subnet name []? subA
 Enter the client name []? ClientA
 Would you like MAC addresses to be transformed to canonical format? [No] yes
```

Example:

```
DHCP Server config> set canonical global
 Would you like MAC addresses to be transformed to canonical format? [No] yes
```

Example:

```
DHCP Server config> set canonical subnet
 Enter the subnet name []? subA
 Would you like MAC addresses to be transformed to canonical format? [No] yes
```

inorder *label-list*

Adds or moves a subnet group to the Inorder list. Addresses will be assigned from the subnets in a subnet group in order of the priority assigned to that subnet.

subnet_group_name

Specifies the subnet group to which this subnet belongs.

Valid Values: An existing subnet group name

Default Value: None

Example:

```
DHCP Server config> set inorder
     Enter the subnet group name []? g2
```

lease-expire-interval *time length*

Specifies the interval at which the lease condition of all the addresses in the address pool is examined to determine which leases have expired. The lease expire interval can only be set at the global level.

time Specifies the unit of time measurement.

Valid Values: seconds, minutes, hours

Default Value: None

length Specifies how long the interval will be.

Valid Values: 15 seconds - 12 hours

Default Value:

- 15 (if the time unit is seconds)
- 1 (if the time unit is minutes)
- 1 (if the time unit is hours)

Example:

```
DHCP Server config> set lease-expire-interval seconds
     How long is the interval in seconds (max:59) [15]? 59
```

Example:

```
DHCP Server config> set lease-expire-interval minutes
     How long is the interval in minutes (max:59) [1]? 45
```

Example:

```
DHCP Server config> set lease-expire-interval hours
     How long is the interval in hours (max:12) [1]? 2
```

lease-time-default *time length*

Specifies the default lease duration for the leases issued by the DHCP Server. An interval of infinity means that leases will never expire. The lease time default can only be set at the global level.

time Specifies the unit of time measurement.

Valid Values: minutes, hours, days, weeks, months, years, infinity

Default Value: None

length Specifies how long the interval will be.

Valid Values: 3 minutes - infinity

Default Value:

- 3 (if the time unit is minutes)
- 1 (if the time unit is hours)
- 1 (if the time unit is days)
- 1 (if the time unit is months)
- 1 (if the time unit is years)

Example:

```
DHCP Server config> set lease-time-default minutes
     How long is the interval in minutes (max:59) [3]? 2
```

Example:

```
DHCP Server config> set lease-time-default hours
     How long is the interval in hours (max:23) [1]? 45
```

Example:

```
DHCP Server config> set lease-time-default days
     How long is the interval in days (max:6) [1]? 2
```

Example:

```
DHCP Server config> set lease-time-default weeks
     How long is the interval in weeks (max:3) [1]? 1
```

Example:

```
DHCP Server config> set lease-time-default months
     How long is the interval in months (max:11) [1]? 3
```

Example:

```
DHCP Server config> set lease-time-default years
     How long is the interval in years (max:10) [1]? 3
```

Example:

```
DHCP Server config> set lease-time-default infinity
```

ping-time *time length*

Before assigning an IP address, the DHCP server tests to be sure the IP address is not in use. This value specifies how long the DHCP server will wait for a ping response before marking the address available. A value of 0 disables pings, resulting in the DHCP server not testing an address before assigning it.

time Specifies the unit of time measurement.

Valid Values: seconds

Default Value: None

length Specifies how long the interval will be.

Valid Values: 0 - 5 seconds

Default Value: 1

Example:

```
DHCP Server config> set ping-time seconds
     How long is the interval in seconds (max:5) [1]? 10
```

support-bootp *value*

Specifies whether the server will respond to requests from BOOTP clients. If the DHCP server was previously configured to support BOOTP clients and has been reconfigured to not support BOOTP clients, the address binding for any BOOTP clients that was established before the reconfiguration will be maintained until the BOOTP client sends another request (when it is restarting). At that time, the server will not respond, and the binding will be removed. This parameter can only be set at the global level.

Valid Values: yes or no

Default Value: no

Example:

```
DHCP Server config> set support-bootp
     Would you like the server to support BOOTP clients? [No] 10
```

support-unlisted-clients *scope [subnet-name] [class-name] value*

Specifies whether the server will respond to requests from DHCP clients other than those whose client IDs are specifically listed in this configuration. This parameter has several possible values:

scope Specifies the scope of the **support-unlisted-clients** parameter.

Valid Values:

- class-global
- class-subnet
- global
- subnet

Default Value: None

subnet-name

Valid is the scope is *subnet*, *class-subnet*, or *client-subnet*. Indicates the name of the subnet for which this parameter is being specified.

Valid Values: An existing subnet name

Default Value: None

class-name

Valid is the scope is *class-global*, or*class-subnet*. Indicates the name of the class for which this parameter is being specified.

Valid Values: An existing class name

Default Value: None

value

yes DHCP server should respond to any client no matter the type or if its configured.

no DHCP server will respond only to requests from DHCP clients that are configured.

bootp DHCP server will support unlisted BOOTP clients but not unlisted DHCP clients

dhcp DHCP server will respond to unlisted DHCP clients but not unlisted BOOTP clients.

Valid Values: yes, no, bootp, dhcp

Default Value: yes, if the **scope** is *global*. Otherwise, the default value is determined by the scope hierarchy. See “Concepts and Terminology” on page 391 for an explanation of scope.

Example:

```
DHCP Server config> set support-unlisted-clients class-global yes
     Enter the class name []? ClassA
```

Example:

```
DHCP Server config> set support-unlisted-clients class-subnet no
     Enter the subnet name []? subA
     Enter the class name []? ClassA
```

Example:

```
DHCP Server config> set support-unlisted-clients global bootp
```

Example:

```
DHCP Server config> set support-unlisted-clients subnet dhcp
     Enter the subnet name []? subA
```

used-ip-address-expire-interval *time length*

Specifies the interval the server will hold an in-use IP address before making the address available for assignment. Before the server allocates an IP address, it pings the address to make sure it is not already in use on the network. The server then marks the in-use address reserved. This parameter specifies how long an in-use address is kept as reserved before making the address available for assignment. This parameter can only be set at the global level.

time Specifies the unit of time measurement.

Valid Values: seconds, minutes, hours, days, weeks, months, years, infinity

Default Value: None

length Specifies how long the interval will be.

Valid Values: 30 seconds - infinity

Default Value:

- 30 (if the time unit is seconds)
- 15 (if the time unit is minutes)
- 1 (if the time unit is hours)
- 1 (if the time unit is days)
- 1 (if the time unit is months)
- 1 (if the time unit is years)

Example:

```
DHCP Server config> set used-ip-address-expire-interval seconds
     How long is the interval in seconds (max:59) [30]? 2
```

Example:

```
DHCP Server config> set used-ip-address-expire-interval minutes
     How long is the interval in minutes (max:59) [15]? 2
```

Example:

```
DHCP Server config> set used-ip-address-expire-interval hours
     How long is the interval in hours (max:23) [1]? 5
```

Example:

```
DHCP Server config> set used-ip-address-expire-interval days
     How long is the interval in days (max:6) [1]? 2
```

Example:

```
DHCP Server config> set used-ip-address-expire-interval weeks
     How long is the interval in weeks (max:3) [1]? 1
```

Example:

```
DHCP Server config> set used-ip-address-expire-interval months
     How long is the interval in months (max:11) [1]? 3
```

Example:

```
DHCP Server config> set used-ip-address-expire-interval years
     How long is the interval in years (max:10) [1]? 3
```

Example:

```
DHCP Server config> set used-ip-address-expire-interval infinity
```

Accessing the DHCP Server Monitoring Environment

Use the following procedure to access the DHCP server *monitoring* process.

1. At the OPCON prompt, enter **talk 5**. For example:

```
* talk 5
Config>
```

After you enter the **talk 5** command, the CONFIG prompt (+) displays on the terminal. If the prompt does not appear when you first enter configuration, press **Return** again.

2. At the + prompt, enter the **feature dhcp-server** command to get to the `DHCP Server>` prompt.

DHCP Server Monitoring Commands

Table 60. DHCP Server Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Disable	Dynamically disables the DHCP server.
Enable	Dynamically enables the DHCP server.
List	Displays parameters for classes, clients, globals, subnets, and vendor-options.
Reset	Dynamically resets the DHCP Server configuration.
Request	
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Disable

Use the **disable** command to dynamically disable the DHCP server.

Syntax:

disable	dhcp

Enable

Use the **enable** command to dynamically enable the DHCP server.

Syntax:

enable	dhcp

List

Use the **list** command to list configuration information about a class, client, global parameters, subnets or vendor-option and any associated options. See “List” on page 427 for examples of the **list** command.

Syntax:

list	class
	client
	global
	option
	subnet
	vendor-option

Reset

Use the **reset** command to dynamically reset the DHCP Server configuration.

Syntax:

reset dhcp

Example:

```
DHCP Server> reset dhcp
You are about to reset the DHCP Server. Clients who have been granted a lease by this
  server will need to renew it.
Are you sure you want to continue? [No]: y
DHCP Server has been reset
DHCP Server>
```

Request

Use the **request** command to display admin information.

Syntax:

request clientid
delete
ipquery
poolquery
stats
status

clientid *client_id*
Displays information for a client.

client_id
Indicates the identifier of the client.

Valid Values: An existing client id

Default Value: None

Example:

```
DHCP Server> request clientid
     Enter the client name []? 0020351FB371

     Client id: 1-0x0020351FB371
     Status: BOUND
     Address last assigned: 192.9.200.10
     Most recent lease time: 16:41:25 December 3, 1998
     Proxy flag: FALSE
     Hostname: Win-XY-1
     Domain name: city.net
```

delete *address*
Deletes a lease for a specific client's IP address.

address
Indicates the IP address of the client to be deleted.

Valid Values: Any valid IP address of an existing client

Default Value: None

Example:

```
DHCP Server> request delete
        Enter the client's IP address []? 194.3.200.10
```

ipquery *address*

Displays information for an IP address.

Example:

```
DHCP Server>req ipquery 192.168.8.3
IP address:       192.168.8.3
Status:           RECLAIMED
Lease time:       86400 seconds
Start time:       Not Leased
Last time leased: 04:16:33  March 9, 1999
DHCP Server>
```

poolquery *address*

Displays information for a pool of IP addresses.

address

Indicates an IP address in the pool to be displayed.

Valid Values: Any valid IP address in the pool to be displayed

Default Value: None

Example:

```
DHCP Server> request poolquery

Enter the client's IP address []? 194.3.200.10
     IP address:            194.3.200.10
     Status:                LEASED
     Lease time:            86400 seconds
     Start time:            16:41:25 December 3, 1998
     Last time leased:      16:41:25 December 3, 1998
     Client id:             1-0x0020351FB371
     Hostname:              Win-XY-1
     Domain name:           city.net
     IP address:            194.3.200.11
     Status:                STOCKED
     IP address:            194.3.200.12
     Status:                STOCKED
```

stats Displays statistics information about the pool of addresses administered by the server. The statistics include: discover packets processed, discover packets with no response, offers made, leases granted, negative acknowledgments (NAKs), informs processed, including informs plus acknowledgments (ACKs), renewals, releases, BOOTP clients processed, proxyARec updated attempted, unsupported packets. Syntax: request stats

Example:

```
DHCP Server> request stats
       Number of DISCOVER requests received:      8
       Number of OFFER responses sent:            4
       Number of ACK responses sent:              3
       Number of NACK responses sent:             0
       Number of RELEASE requests received:       0
       Number of DECLINE packets received:        0
       Number of INFORM requests received:        0
       Number of BOOTP requests received:         0
       Number of requests received via proxy:     0
       Number of UNSUPPORTED requests received:   0
       Total number of request/responses:        15
       Number of lease expirations:               0
```

status Displays information about the address pools.

Example:

```
DHCP Server> request status

     IP address:            194.3.200.10
     Status:                LEASED
     Lease time:            86400 seconds
     Start time:            16:41:25 December 3, 1998
     Last time leased:      16:41:25 December 3, 1998
     Client id:             1-0x0020351FB371
     Hostname:              Win-XY-1
     Domain name:           city.net

     IP address:            194.3.200.11
     Status:                STOCKED

     IP address:            194.3.200.12
     Status:                STOCKED

     IP address:            194.3.200.10
     Status:                STOCKED
```

DHCP Server Monitoring Commands (Talk 6)

Chapter 30. Configuring and Monitoring VCRM

Virtual Circuit Resource Manager (VCRM) is a feature that supports Resource ReSerVation Protocol (RSVP), which is described in “Using RSVP” and “Configuring and Monitoring RSVP” in the *Protocol Configuration and Monitoring Reference Volume 1*. Based upon the reservation request from RSVP, VCRM creates the connection for the data flow over the physical interface. To do this, VCRM must first determine whether enough bandwidth exists to accommodate the reservation.

Note: If you are using WAN interfaces such as frame relay or X.25, you need to set the line speed so that VCRM knows how much bandwidth is available. The procedure for setting the line speed is described in the Frame Relay and X.25 interface configuration and monitoring chapters of the *Software User's Guide*.

If the interface is ATM SVC, VCRM maps RSVP QoS requests to SVC setup requests. The RSVP reservation request succeeds if the SVC setup succeeds. VCRM ensures that there is adequate buffer space for the QoS packets and that these packets are sent over the correct SVC for transmission.

If the interface is not ATM, such as PPP link, LAN, or WAN, VCRM uses software queuing of the QoS and best-effort packets to prioritize the packets on the outbound link.

This chapter includes the following sections:

Accessing the VCRM Configuration Environment

To access the VCRM configuration environment, enter the following command at the `Config>` prompt:

```
Config> feature vcrm
VC & Resource Management config console
  --Currently no configurable objects.
Config>
```

The purpose of the message displayed is to indicate that VCRM cannot be separately configured. Enabling RSVP enables VCRM, which obtains its parameters from the RSVP configuration.

Accessing the VCRM Monitoring Environment

To access the VCRM monitoring environment, type

```
* t 5
```

Then, enter the following command at the **+** prompt:

```
+ feature VCRM
VCRM console
VCRM Console>
```

The VCRM Console> prompt appears.

VCRM Monitoring Commands

This section describes the VCRM monitoring commands. Enter these commands at the VCRM Console> prompt.

Table 61. VCRM Monitoring Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page xxix.
Clear	Resets the queue statistics.
Queue	Shows non-ATM software queuing statistics.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page xxix.

Clear

Use the **clear** command to reset the software queue statistics.

Syntax:

clear

See the **queue** command for an example of the **clear** command.

Queue

Use the **queue** command to show the software queuing of the traffic flows that are not ATM.

Syntax:

queue

The following list defines the terms used in displaying the non-ATM software queues:

Quota Amount of bandwidth reserved. Originally, best-effort (B.E.) has all the quotas. When a reservation is made, the reserved bandwidth (b/w) is shifted from the B.E. quota to the QoS quota.

Max-q Maximum queue length, stated in packets.

Curr-q
Current queue length, stated in packets.

In quota
Packets or kilobytes sent within the allocated bandwidth.

Outside quota
Packets or kilobytes sent outside of the allocated bandwidth, when idle bandwidth was available.

Packets/bytes dropped
Packets or bytes dropped by software queueing.

DLC packets/bytes dropped

Packets or bytes dropped by DLC after the packets have gone through the software queue.

Example:

```
*t 5

+feature vcrm
VCRM console
VCRM Console>?
CLEAR
QUEUE
EXIT
VCRM Console>queue
Flow-control Queues at sys-clock 346781 Second:
---------
Intf    B.E. Quota:       10000 Kbps            QoS Quota:        0     Kbps
0/Eth   B.E. Max-q        0                     QoS Max-q         0
        B.E. curr-q       0                     QoS curr-q        0
        B.E. pkts/Kbytes sent:                  QoS pkts/Kbytes sent:
        in quota:        54169/   3926          in quota:            0/      0
        outside quota:       0/      0          outside quota:       0/      0
        B.E. pkts/bytes dropped: 0/0            QoS pkts/bytes dropped:  0/0
        DLC pkts/bytes dropped:  B.E.: 0/0      QoS: 0/0
Intf    B.E. Quota:       2048  Kbps            QoS Quota:        0     Kbps
2/PPP   B.E. Max-q        0                     QoS Max-q         0
        B.E. curr-q       0                     QoS curr-q        0
        B.E. pkts/Kbytes sent:                  QoS pkts/Kbytes sent:
        in quota:           62/      6          in quota:            0/      0
        outside quota:       0/      0          outside quota:       0/      0
        B.E. pkts/bytes dropped: 0/0            QoS pkts/bytes dropped:  0/0
        DLC pkts/bytes dropped:  B.E.: 0/0      QoS: 0/0
Intf    B.E. Quota:       2032  Kbps            QoS Quota:        16    Kbps
3/FR    B.E. Max-q        1                     QoS Max-q         1
        B.E. curr-q       0                     QoS curr-q        0
        B.E. pkts/Kbytes sent:                  QoS pkts/Kbytes sent:
        in quota:        53160/   4920          in quota:       346596/  31886
        outside quota:       0/      0          outside quota:       0/      0
        B.E. pkts/bytes dropped: 0/0            QoS pkts/bytes dropped:  0/0
        DLC pkts/bytes dropped:  B.E.: 0/0      QoS: 0/0
Intf    B.E. Quota:       2048  Kbps            QoS Quota:        0     Kbps
4/PPP   B.E. Max-q        1                     QoS Max-q         1
        B.E. curr-q       0                     QoS curr-q        0
        B.E. pkts/Kbytes sent:                  QoS pkts/Kbytes sent:
        in quota:           66/      6          in quota:          109/     1
        outside quota:       0/      0          outside quota:       0/
        B.E. pkts/bytes dropped: 0/0            QoS pkts/bytes dropped:  0/0
        DLC pkts/bytes dropped:  B.E.: 0/0      QoS: 0/0

Max total queue length=1;  current total length=0
VCRM Console>clear
Flow-control Queues cleared at sys-clock 346786 Second:
-------
VCRM Console>
```

Monitoring VCRM (Talk 5)

Appendix. Remote AAA Attributes

This section contains the remote AAA Attributes use by Radius, TACACS and TACACS+ servers.

Radius

IBM Vendor ID: 211

Authorization Attributes

Standard Drafted

TUNNEL_TYPE	64
TUNNEL_MEDIUM_TYPE	65
TUNNEL_CLIEN_TYPE	66
TUNNEL_SERVER_EP	67
TUNNEL_CONN_ID	68
TUNNEL_PASSWORD	69

values

TUNNEL_TYPE		integer
3	L2TP	
TUNNEL_MEDIUM_TYPE		integer
1	IP	
TUNNEL_SERVER_EP		string
	ip address	

IBM Vendor Specific

NAS_TUNNEL_PASSWORD	101
CALLBACK_FLAGS	210
ENCRYPTION	211
HOSTNAME	213
DIALOUT	214
SUBNETMASK	215
PRIVILEGE	216

Keywords

Keywords are used for Radius servers that allow the entry of vendor specific fields `<keyword>=<value>`.

KWD_CALLBACK_FLAGS	CBF
KWD_ENCRYPTION	ENC
KWD_HOSTNAME	HSN
KWD_DIALOUT	DOF
KWD_SUBNETMASK	SNM

KWD_PRIVELGE	PRV

Values

PRIVILEGE:

ADMIN	
OPER	
MONITOR	

CALLBACKFLAGS

REQ	required callback
ROAM	roaming callback

DIALOUT

TRUE	enable dialout for this user
FALSE	disable dialout for this user
ONLY	only allow dialout for this user (not dial in)

TACACS+

Authentication

Authorization

PPP service=ppp protocol=ip
LOGIN service=shell cmd=null pri_lvl*0

Standard TACACS+ Attributes

- service
- protocol
- cmd
- addr
- timeout
- priv_lvl
- callback-dialstring

IBM Specific Attributes

encryption_key	16 hex characters
dial_out	TRUE FALSE ONLY

Accounting

- task_id
- start_time
- stop_time
- elasped_time
- timezone
- event
- reason
- bytes
- bytes_in
- bytes_out
- paks

paks_in
paks_out
status
err_msg

List of Abbreviations

AARP AppleTalk Address Resolution Protocol

ABR area border router

ack acknowledgment

AIX Advanced Interactive Executive

AMA arbitrary MAC addressing

AMP active monitor present

ANSI American National Standards Institute

AP2 AppleTalk Phase 2

APPN Advanced Peer-to-Peer Networking

ARE all-routes explorer

ARI ATM real interface

ARI/FCI
address recognized indicator/frame copied indicator

ARP Address Resolution Protocol

AS autonomous system

ASBR autonomous system boundary router

ASCII American National Standard Code for Information Interchange

ASN.1 abstract syntax notation 1

ASRT adaptive source routing transparent

ASYNC
asynchronous

ATCP AppleTalk Control Protocol

ATP AppleTalk Transaction Protocol

AUI attachment unit interface

AVI ATM virtual interface

ayt are you there

BAN Boundary Access Node

BBCM Bridging Broadcast Manager

BECN backward explicit congestion notification

BGP Border Gateway Protocol

BNC bayonet Niell-Concelman

BNCP Bridging Network Control Protocol

BOOTP
BOOT protocol

BPDU bridge protocol data unit

bps bits per second

BR bridging/routing

BRS bandwidth reservation

BSD Berkeley software distribution

BTP BOOTP relay agent

BTU basic transmission unit

CAM content-addressable memory

CCITT Consultative Committee on International Telegraph and Telephone

CD collision detection

CGWCON
Gateway Console

CIDR Classless Inter-Domain Routing

CIP Classical IP

CIR committed information rate

CLNP Connectionless-Mode Network Protocol

CPU central processing unit

CRC cyclic redundancy check

CRS configuration report server

CTS clear to send

CUD call user data

DAF destination address filtering

DB database

DBsum
database summary

DCD data channel received line signal detector

DCE data circuit-terminating equipment

DCS Directly connected server

DDLC dual data-link controller

DDN Defense Data Network

DDP Datagram Delivery Protocol

DDT Dynamic Debugging Tool

DHCP Dynamic Host Configuration Protocol

dir directly connected

DL data link

DLC data link control

DLCI data link connection identifier

DLS data link switching

DLSw data link switching

DMA direct memory access

DNA Digital Network Architecture

DNCP DECnet Protocol Control Protocol

DNIC Data Network Identifier Code

DoD Department of Defense

DOS Disk Operating System

DR designated router

DRAM Dynamic Random Access Memory

DSAP destination service access point

DSE data switching equipment

DSE data switching exchange

DSR data set ready

DSU data service unit

DTE data terminal equipment

DTR data terminal ready

Dtype destination type

DVMRP
Distance Vector Multicast Routing Protocol

E1 2.048 Mbps transmission rate

EDEL end delimiter

EDI error detected indicator

EGP Exterior Gateway Protocol

EIA Electronics Industries Association

ELAN Emulated LAN

ELAP EtherTalk Link Access Protocol

ELS Event Logging System

ESI End system identifier

EST Eastern Standard Time

Eth Ethernet

fa-ga functional address-group address

FCS frame check sequence

FECN forward explicit congestion notification

FIFO first in, first out

FLT filter library

FR Frame Relay

FRL Frame Relay

FTP File Transfer Protocol

GMT Greenwich Mean Time

GOSIP
Government Open Systems Interconnection Profile

GTE General Telephone Company

GWCON
Gateway Console

HDLC high-level data link control

HEX hexadecimal

HPR high-performance routing

HST TCP/IP host services

HTF host table format

IBD Integrated Boot Device

ICMP Internet Control Message Protocol

ICP Internet Control Protocol

ID identification

IDP Initial Domain Part

IDP Internet Datagram Protocol

IEEE Institute of Electrical and Electronics Engineers

Ifc# interface number

IGP interior gateway protocol

InARP Inverse Address Resolution Protocol

IP Internet Protocol

IPCP IP Control Protocol

IPPN IP Protocol Network

IPX Internetwork Packet Exchange

IPXCP IPX Control Protocol

ISDN integrated services digital network

ISO International Organization for Standardization

Kbps kilobits per second

LAC L2TP Network Access Concentrator

LAN local area network

LAPB link access protocol-balanced

LAT local area transport

LCP Link Control Protocol

LED light-emitting diode

LF largest frame; line feed

LIS Logical IP subnet

LLC logical link control

LLC2 logical link control 2

LMI local management interface

LNS L2TP Network Server

LRM LAN reporting mechanism

LS link state

LSA link state advertisement

LSB least significant bit

LSI LAN shortcuts interface

LSreq link state request

LSrxl link state retransmission list

LU logical unit

MAC medium access control

Mb megabit

MB megabyte

Mbps megabits per second

MBps megabytes per second

MC multicast

MCF MAC filtering

MIB Management Information Base

MIB II Management Information Base II

MILNET
military network

MOS Micro Operating System

MOSDBG
Micro Operating System Debugging Tool

MOSPF
Open Shortest Path First with multicast extensions

MSB most significant bit

MSDU MAC service data unit

MRU maximum receive unit

MTU maximum transmission unit

nak not acknowledged

NBMA Non-Broadcast Multiple Access

NBP Name Binding Protocol

NBR neighbor

NCP Network Control Protocol

NCP Network Core Protocol

NetBIOS
Network Basic Input/Output System

NHRP Next Hop Resolution Protocol

NIST National Institute of Standards and Technology

NPDU Network Protocol Data Unit

NRZ non-return-to-zero

NRZI non-return-to-zero inverted

NSAP Network Service Access Point

NSF National Science Foundation

NSFNET
National Science Foundation NETwork

NVCNFG
nonvolatile configuration

OPCON
Operator Console

OSI open systems interconnection

OSICP
OSI Control Protocol

OSPF Open Shortest Path First

OUI organization unique identifier

PC personal computer

PCR peak cell rate

PDN public data network

PING Packet internet groper

PDU protocol data unit

PID process identification

P-P Point-to-Point

PPP Point-to-Point Protocol

PROM programmable read-only memory

PU physical unit

PVC permanent virtual circuit

RAM random access memory

RD route descriptor

REM ring error monitor

REV receive

RFC Request for Comments

RI ring indicator; routing information

RIF routing information field

RII routing information indicator

RIP Routing Information Protocol

RISC reduced instruction-set computer

RNR receive not ready

ROM read-only memory

ROpcon
Remote Operator Console

RPS ring parameter server

RTMP Routing Table Maintenance Protocol

RTP RouTing update Protocol

RTS request to send

Rtype route type

rxmits retransmissions

rxmt retransmit

SAF source address filtering

SAP service access point

SAP Service Advertising Protocol

SCR Sustained cell rate

SCSP Server Cache Synchronization Protocol

sdel start delimiter

SDLC SDLC relay, synchronous data link control

seqno sequence number

SGID sever group id

SGMP Simple Gateway Monitoring Protocol

SL serial line

SMP standby monitor present

SMTP Simple Mail Transfer Protocol

SNA Systems Network Architecture

SNAP Subnetwork Access Protocol

SNMP Simple Network Management Protocol

SNPA subnetwork point of attachment

SPF OSPF intra-area route

SPE1 OSPF external route type 1

SPE2 OSPF external route type 2

SPIA OSPF inter-area route type

SPID service profile ID

SPX Sequenced Packet Exchange

SQE signal quality error

SRAM static random access memory

SRB source routing bridge

SRF specifically routed frame

SRLY SDLC relay

SRT source routing transparent

SR-TB
source routing-transparent bridge

STA static

STB spanning tree bridge

STE spanning tree explorer

STP shielded twisted pair; spanning tree protocol

SVC switched virtual circuit

TB transparent bridge

TCN topology change notification

TCP Transmission Control Protocol

TCP/IP
Transmission Control Protocol/Internet Protocol

TEI terminal point identifier

TFTP Trivial File Transfer Protocol

TKR token ring

TMO timeout

TOS type of service

TSF transparent spanning frames

TTL time to live

TTY teletypewriter

TX transmit

UA unnumbered acknowledgment

UDP User Datagram Protocol

UI unnumbered information

UTP unshielded twisted pair

VCC Virtual Channel Connection

VINES VIrtual NEtworking System

VIR variable information rate

VL virtual link

VNI Virtual Network Interface

VR virtual route

WAN wide area network

WRS WAN restoral/reroute

X.25 packet-switched networks

X.251 X.25 physical layer

X.252 X.25 frame layer

X.253 X.25 packet layer

XID exchange identification

XNS Xerox Network Systems

XSUM checksum

ZIP AppleTalk Zone Information Protocol

ZIP2 AppleTalk Zone Information Protocol 2

ZIT Zone Information Table

Glossary

This glossary includes terms and definitions from:

- The *American National Standard Dictionary for Information Systems* , ANSI X3.172-1990, copyright 1990 by the American National Standards Institute (ANSI). Copies may be purchased from the American National Standards Institute, 11 West 42nd Street, New York, New York 10036. Definitions are identified by the symbol (A) after the definition.
- The ANSI/EIA Standard—440-A, *Fiber Optic Terminology* Copies may be purchased from the Electronic Industries Association, 2001 Pennsylvania Avenue, N.W., Washington, DC 20006. Definitions are identified by the symbol (E) after the definition.
- The *Information Technology Vocabulary* developed by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Commission (ISO/IEC JTC1/SC1). Definitions of published parts of this vocabulary are identified by the symbol (I) after the definition; definitions taken from draft international standards, committee drafts, and working papers being developed by ISO/IEC JTC1/SC1 are identified by the symbol (T) after the definition, indicating that final agreement has not yet been reached among the participating National Bodies of SC1.
- The *IBM Dictionary of Computing* , New York: McGraw-Hill, 1994.
- Internet Request for Comments: 1208, *Glossary of Networking Terms*
- Internet Request for Comments: 1392, *Internet Users' Glossary*
- The *Object-Oriented Interface Design: IBM Common User Access Guidelines* , Carmel, Indiana: Que, 1992.

The following cross-references are used in this glossary:

Contrast with:
This refers to a term that has an opposed or substantively different meaning.

Synonym for:
This indicates that the term has the same meaning as a preferred term, which is defined in its proper place in the glossary.

Synonymous with:
This is a backward reference from a defined term to all other terms that have the same meaning.

See: This refers the reader to multiple-word terms that have the same last word.

See also:
This refers the reader to terms that have a related, but not synonymous, meaning.

A

AAL. ATM Adaptation Layer, the layer that adapts user data to/from the ATM network by adding/removing headers and segmenting/reassembling the data into/from cells.

AAL-5. ATM Adaptation Layer 5, one of several standard AALs. AAL-5 was designed for data communications, and is used by LAN Emulation and Classical IP.

abstract syntax. A data specification that includes all distinctions that are needed in data transmissions, but that omits (abstracts) other details such as those that depend on specific computer architectures. See also *abstract syntax notation 1 (ASN.1)* and *basic encoding rules (BER)*.

abstract syntax notation 1 (ASN.1). The Open Systems Interconnection (OSI) method for abstract syntax specified in the following standards:

- ITU-T Recommendation X.208 (1988) | ISO/IEC 8824: 1990
- ITU-T Recommendation X.680 (1994) | ISO/IEC 8824-1: 1994

See also *basic encoding rules (BER)*.

ACCESS. In the Simple Network Management Protocol (SNMP), the clause in a Management Information Base (MIB) module that defines the minimum level of support that a managed node provides for an object.

acknowledgment. (1) The transmission, by a receiver, of acknowledge characters as an affirmative response to a sender. (T) (2) An indication that an item sent was received.

active. (1) Operational. (2) Pertaining to a node or device that is connected or is available for connection to another node or device.

active monitor. In a token-ring network, a function performed at any one time by one ring station that

initiates the transmission of tokens and provides token error recovery facilities. Any active adapter on the ring has the ability to provide the active monitor function if the current active monitor fails.

address. In data communication, the unique code assigned to each device, workstation, or user connected to a network.

address mapping table (AMT). A table, maintained within the AppleTalk router, that provides a current mapping of node addresses to hardware addresses.

address mask. For internet subnetworking, a 32-bit mask used to identify the subnetwork address bits in the host portion of an IP address. Synonymous with *subnet mask* and *subnetwork mask*.

address resolution. (1) A method for mapping network-layer addresses to media-specific addresses. (2) See also *Address Resolution Protocol (ARP)* and *AppleTalk Address Resolution Protocol (AARP)*.

Address Resolution Protocol (ARP). (1) In the Internet suite of protocols, the protocol that dynamically maps an IP address to an address used by a supporting metropolitan or local area network such as Ethernet or token-ring. (2) See also *Reverse Address Resolution Protocol (RARP)*.

addressing. In data communication, the way in which a station selects the station to which it is to send data.

adjacent nodes. Two nodes connected together by at least one path that connects no other node. (T)

Administrative Domain. A collection of hosts and routers, and the interconnecting networks, managed by a single administrative authority.

Advanced Peer-to-Peer Networking (APPN). An extension to SNA featuring (a) greater distributed network control that avoids critical hierarchical dependencies, thereby isolating the effects of single points of failure; (b) dynamic exchange of network topology information to foster ease of connection, reconfiguration, and adaptive route selection; (c) dynamic definition of network resources; and (d) automated resource registration and directory lookup. APPN extends the LU 6.2 peer orientation for end-user services to network control and supports multiple LU types, including LU 2, LU 3, and LU 6.2.

Advanced Peer-to-Peer Networking (APPN) end node. A node that provides a broad range of end-user services and supports sessions between its local control point (CP) and the CP in an adjacent network node. It uses these sessions to dynamically register its resources with the adjacent CP (its network node server), to send and receive directory search requests, and to obtain management services. An APPN end node can also attach to a subarea network as a peripheral node or to other end nodes.

Advanced Peer-to-Peer Networking (APPN) network. A collection of interconnected network nodes and their client end nodes.

Advanced Peer-to-Peer Networking (APPN) network node. A node that offers a broad range of end-user services and that can provide the following:

- Distributed directory services, including registration of its domain resources to a central directory server
- Topology database exchanges with other APPN network nodes, enabling network nodes throughout the network to select optimal routes for LU-LU sessions based on requested classes of service
- Session services for its local LUs and client end nodes
- Intermediate routing services within an APPN network

Advanced Peer-to-Peer Networking (APPN) node. An APPN network node or an APPN end node.

agent. A system that assumes an agent role.

alert. A message sent to a management services focal point in a network to identify a problem or an impending problem.

all-stations address. In communications, synonym for *broadcast address*.

American National Standards Institute (ANSI). An organization consisting of producers, consumers, and general interest groups, that establishes the procedures by which accredited organizations create and maintain voluntary industry standards in the United States. (A)

analog. (1) Pertaining to data consisting of continuously variable physical quantities. (A) (2) Contrast with *digital*.

AppleTalk. A network protocol developed by Apple Computer, Inc. This protocol is used to interconnect network devices, which can be a mixture of Apple and non-Apple products.

AppleTalk Address Resolution Protocol (AARP). In AppleTalk networks, a protocol that (a) translates AppleTalk node addresses into hardware addresses and (b) reconciles addressing discrepancies in networks that support more than one set of protocols.

AppleTalk Transaction Protocol (ATP). In AppleTalk networks, a protocol that provides client/server request and response functions for hosts accessing the Zone Information Protocol (ZIP) for zone information.

APPN network. See *Advanced Peer-to-Peer Networking (APPN) network*.

APPN network node. See *Advanced Peer-to-Peer Networking (APPN) network node*.

arbitrary MAC addressing (AMA). In DECnet architecture, an addressing scheme used by DECnet Phase IV-Prime that supports universally administered addresses and locally administered addresses.

area. In Internet and DECnet routing protocols, a subset of a network or gateway grouped together by definition of the network administrator. Each area is self-contained; knowledge of an area's topology remains hidden from other areas.

asynchronous (ASYNC). Pertaining to two or more processes that do not depend upon the occurrence of specific events such as common timing signals. (T)

ATM. Asynchronous Transfer Mode, a connection-oriented, high-speed networking technology based on cell switching.

ATMARP. ARP in Classical IP.

attachment unit interface (AUI). In a local area network, the interface between the medium attachment unit and the data terminal equipment within a data station. (I) (A)

Attribute Value Pair (AVP). A uniform method of encoding message types and bodies. This method maximizes the extensibility while permitting interoperability of L2TP.

authentication failure. In the Simple Network Management Protocol (SNMP), a trap that may be generated by an authentication entity when a requesting client is not a member of the SNMP community.

autonomous system. In TCP/IP, a group of networks and routers under one administrative authority. These networks and routers cooperate closely to propagate network reachability (and routing) information among themselves using an interior gateway protocol of their choice.

autonomous system number. In TCP/IP, a number assigned to an autonomous system by the same central authority that also assigns IP addresses. The autonomous system number makes it possible for automated routing algorithms to distinguish autonomous systems.

B

backbone. (1) In a local area network multiple-bridge ring configuration, a high-speed link to which the rings are connected by means of bridges or routers. A backbone may be configured as a bus or as a ring. (2) In a wide area network, a high-speed link to which nodes or data switching exchanges (DSEs) are connected.

backbone network. A central network to which smaller networks, normally of lower speed, connect. The backbone network usually has a much higher capacity than the networks it helps interconnect or is a wide-area network (WAN) such as a public packet-switched datagram network.

backbone router. (1) A router used to transmit data between areas. (2) One in a series of routers that is used to interconnect networks into a larger internet.

Bandwidth. The bandwidth of an optical link designates the information-carrying capacity of the link and is related to the maximum bit rate that a fiber link can support.

basic transmission unit (BTU). In SNA, the unit of data and control information passed between path control components. A BTU can consist of one or more path information units (PIUs).

baud. In asynchronous transmission, the unit of modulation rate corresponding to one unit interval per second; that is, if the duration of the unit interval is 20 milliseconds, the modulation rate is 50 baud. (A)

bootstrap. (1) A sequence of instructions whose execution causes additional instructions to be loaded and executed until the complete computer program is in storage. (T) (2) A technique or device designed to bring itself into a desired state by means of its own action, for example, a machine routine whose first few instructions are sufficient to bring the rest of itself into the computer from an input device. (A)

Border Gateway Protocol (BGP). An Internet Protocol (IP) routing protocol used between domains and autonomous systems.

border router. In Internet communications, a router, positioned at the edge of an autonomous system, that communicates with a router that is positioned at the edge of a different autonomous system.

bridge. A functional unit that interconnects multiple LANs (locally or remotely) that use the same logical link control protocol but that can use different medium access control protocols. A bridge forwards a frame to another bridge based on the medium access control (MAC) address.

bridge identifier. An 8-byte field, used in a spanning tree protocol, composed of the MAC address of the port with the lowest port identifier and a user-defined value.

bridging. In LANs, the forwarding of a frame from one LAN segment to another. The destination is specified by the medium access control (MAC) sublayer address encoded in the destination address field of the frame header.

broadcast. (1) Transmission of the same data to all destinations. (T) (2) Simultaneous transmission of data to more than one destination. (3) Contrast with *multicast*.

broadcast address. In communications, a station address (eight 1's) reserved as an address common to all stations on a link. Synonymous with *all-stations address*.

C

cache. (1) A special-purpose buffer storage, smaller and faster than main storage, used to hold a copy of instructions and data obtained from main storage and likely to be needed next by the processor. (T) (2) A buffer storage that contains frequently accessed instructions and data; it is used to reduce access time. (3) An optional part of the directory database in network nodes where frequently used directory information may be stored to speed directory searches. (4) To place, hide, or store in a cache.

call request packet. (1) A call supervision packet that a data terminal equipment (DTE) transmits to ask that a connection for a call be established throughout the network. (2) In X.25 communications, a call supervision packet transmitted by a DTE to ask for a call establishment through the network.

canonical address. In LANs, the IEEE 802.1 format for the transmission of medium access control (MAC) addresses for token-ring and Ethernet adapters. In canonical format, the least significant (rightmost) bit of each address byte is transmitted first. Contrast with *noncanonical address*.

carrier. An electric or electromagnetic wave or pulse train that may be varied by a signal bearing information to be transmitted over a communication system. (T)

carrier detect. Synonym for *received line signal detector (RLSD)*.

carrier sense. In a local area network, an ongoing activity of a data station to detect whether another station is transmitting. (T)

carrier sense multiple access with collision detection (CSMA/CD). A protocol that requires carrier sense and in which a transmitting data station that detects another signal while transmitting, stops sending, sends a jam signal, and then waits for a variable time before trying again. (T) (A)

CCITT. International Telegraph and Telephone Consultative Committee. This was an organization of the International Telecommunication Union (ITU). On 1 March 1993 the ITU was reorganized, and responsibilities for standardization were placed in a subordinate organization named the Telecommunication Standardization Sector of the Telecommunication Union (ITU-TS). "CCITT" continues to be used for recommendations that were approved before the reorganization.

channel. (1) A path along which signals can be sent, for example, data channel, output channel. (A) (2) A functional unit, controlled by the processor, that handles the transfer of data between processor storage and local peripheral equipment.

channel service unit (CSU). A unit that provides the interface to a digital network. The CSU provides line conditioning (or equalization) functions, which keep the signal's performance consistent across the channel bandwidth; signal reshaping, which constitutes the binary pulse stream; and loopback testing, which includes the transmission of test signals between the CSU and the network carrier's office channel unit. See also *data service unit (DSU)*.

checksum. (1) The sum of a group of data associated with the group and used for checking purposes. (T) (2) In error detection, a function of all bits in a block. If the written and calculated sums do not agree, an error is indicated. (3) On a diskette, data written in a sector for error detection purposes; a calculated checksum that does not match the checksum of data written in the sector indicates a bad sector. The data are either numeric or other character strings regarded as numeric for the purpose of calculating the checksum.

circuit switching. (1) A process that, on demand, connects two or more data terminal equipment (DTEs) and permits the exclusive use of a data circuit between them until the connection is released. (I) (A) (2) Synonymous with *line switching*.

class A network. In Internet communications, a network in which the high-order (most significant) bit of the IP address is set to 0 and the host ID occupies the three low-order octets.

class B network. In Internet communications, a network in which the two high-order (most significant and next-to-most significant) bits of the IP address are set to 1 and 0, respectively, and the host ID occupies the two low-order octets.

class of service (COS). A set of characteristics (such as route security, transmission priority, and bandwidth) used to construct a route between session partners. The class of service is derived from a mode name specified by the initiator of a session.

client. (1) A functional unit that receives shared services from a server. (T) (2) A user.

client/server. In communications, the model of interaction in distributed data processing in which a program at one site sends a request to a program at another site and awaits a response. The requesting program is called a client; the answering program is called a server.

clocking. (1) In binary synchronous communication, the use of clock pulses to control synchronization of

data and control characters. (2) A method of controlling the number of data bits sent on a telecommunication line in a given time.

collision. An unwanted condition that results from concurrent transmissions on a channel. (T)

collision detection. In carrier sense multiple access with collision detection (CSMA/CD), a signal indicating that two or more stations are transmitting simultaneously.

Committed information rate. The maximum amount of data in bits that the network agrees to deliver.

community. In the Simple Network Management Protocol (SNMP), an administrative relationship between entities.

community name. In the Simple Network Management Protocol (SNMP), a string of octets identifying a community.

compression. (1) The process of eliminating gaps, empty fields, redundancies, and unnecessary data to shorten the length of records or blocks. (2) Any encoding to reduce the number of bits used to represent a given message or record.

configuration. (1) The manner in which the hardware and software of an information processing system are organized and interconnected. (T) (2) The devices and programs that make up a system, subsystem, or network.

configuration database (CDB). A database that stores the configuration parameters of one or several devices. It is prepared and updated using the configuration program.

configuration file. A file that specifies the characteristics of a system device or network.

configuration parameter. A variable in a configuration definition, the values of which can characterize the relationship of a product to other products in the same network or can define characteristics of the product itself.

configuration report server (CRS). In the IBM Token-Ring Network Bridge Program, the server that accepts commands from the LAN Network Manager (LNM) to get station information, set station parameters, and remove stations on its ring. This server also collects and forwards configuration reports generated by stations on its ring. The configuration reports include the new active monitor reports and the nearest active upstream neighbor (NAUN) reports.

congestion. See *network congestion*.

connection. In data communication, an association established between functional units for conveying information. (I) (A)

control point (CP). (1) A component of an APPN or LEN node that manages the resources of that node. In an APPN node, the CP is capable of engaging in CP-CP sessions with other APPN nodes. In an APPN network node, the CP also provides services to adjacent end nodes in the APPN network. (2) A component of a node that manages resources of that node and optionally provides services to other nodes in the network. Examples are a system services control point (SSCP) in a type 5 subarea node, a network node control point (NNCP) in an APPN network node, and an end node control point (ENCP) in an APPN or LEN end node. An SSCP and an NNCP can provide services to other nodes.

control point management services (CPMS). A component of a control point, consisting of management services function sets, that provides facilities to assist in performing problem management, performance and accounting management, change management, and configuration management. Capabilities provided by the CPMS include sending requests to physical unit management services (PUMS) to test system resources, collecting statistical information (for example, error and performance data) from PUMS about the system resources, and analyzing and presenting test results and statistical information collected about the system resources. Analysis and presentation responsibilities for problem determination and performance monitoring can be distributed among multiple CPMSs.

control point management services unit (CP-MSU). The message unit that contains management services data and flows between management services function sets. This message unit is in general data stream (GDS) format. See also *management services unit (MSU)* and *network management vector transport (NMVT)*.

D

D-bit. Delivery-confirmation bit. In X.25 communications, the bit in a data packet or call-request packet that is set to 1 if end-to-end acknowledgment (delivery confirmation) is required from the recipient.

daemon. A program that runs unattended to perform a standard service. Some daemons are triggered automatically to perform their task; others operate periodically.

data carrier detect (DCD). Synonym for *received line signal detector (RLSD)*.

data circuit. (1) A pair of associated transmit and receive channels that provide a means of two-way data communication. (I) (2) In SNA, synonym for *link connection*. (3) See also *physical circuit* and *virtual circuit*.

Notes:

1. Between data switching exchanges, the data circuit may include data circuit-terminating equipment (DCE), depending on the type of interface used at the data switching exchange.
2. Between a data station and a data switching exchange or data concentrator, the data circuit includes the data circuit-terminating equipment at the data station end, and may include equipment similar to a DCE at the data switching exchange or data concentrator location.

data circuit-terminating equipment (DCE). In a data station, the equipment that provides the signal conversion and coding between the data terminal equipment (DTE) and the line. (I)

Notes:

1. The DCE may be separate equipment or an integral part of the DTE or of the intermediate equipment.
2. A DCE may perform other functions that are usually performed at the network end of the line.

data link connection identifier (DLCI). The numeric identifier of a frame-relay subport or PVC segment in a frame-relay network. Each subport in a single frame-relay port has a unique DLCI. The following table, excerpted from the American National Standards Institute (ANSI) Standard T1.618 and the International Telegraph and Telephone Consultative Committee (ITU-T/CCITT) Standard Q.922, indicates the functions associated with certain DLCI values:

DLCI Values	Function
0	in-channel signaling
1–15	reserved
16–991	assigned using frame-relay connection procedures
992–1007	layer 2 management of frame-relay bearer service
1008–1022	reserved
1023	in-channel layer management

data link control (DLC). A set of rules used by nodes on a data link (such as an SDLC link or a token ring) to accomplish an orderly exchange of information.

data link control (DLC) layer. In SNA, the layer that consists of the link stations that schedule data transfer over a link between two nodes and perform error control for the link. Examples of data link control are SDLC for serial-by-bit link connection and data link control for the System/370 channel.

Note: The DLC layer is usually independent of the physical transport mechanism and ensures the integrity of data that reaches the higher layers.

data link layer. In the Open Systems Interconnection reference model, the layer that provides services to transfer data between entities in the network layer over a communication link. The data link layer detects and possibly corrects errors that may occur in the physical layer. (T)

data link level. (1) In the hierarchical structure of a data station, the conceptual level of control or processing logic between high level logic and the data link that maintains control of the data link. The data link level performs such functions as inserting transmit bits and deleting receive bits; interpreting address and control fields; generating, transmitting, and interpreting commands and responses; and computing and interpreting frame check sequences. See also *packet level* and *physical level*. (2) In X.25 communications, synonym for *frame level*.

data link switching (DLSw). A method of transporting network protocols that use IEEE 802.2 logical link control (LLC) type 2. SNA and NetBIOS are examples of protocols that use LLC type 2. See also *encapsulation* and *spoofing*.

data packet. In X.25 communications, a packet used for the transmission of user data on a virtual circuit at the DTE/DCE interface.

data service unit (DSU). A device that provides a digital data service interface directly to the data terminal equipment. The DSU provides loop equalization, remote and local testing capabilities, and a standard EIA/CCITT interface.

data set ready (DSR). Synonym for *DCE ready*.

data switching exchange (DSE). The equipment installed at a single location to provide switching functions, such as circuit switching, message switching, and packet switching. (I)

data terminal equipment (DTE). That part of a data station that serves as a data source, data sink, or both. (I) (A)

data terminal ready (DTR). A signal to the modem used with the EIA 232 protocol.

data transfer rate. The average number of bits, characters, or blocks per unit time passing between corresponding equipment in a data transmission system. (I)

Notes:

1. The rate is expressed in bits, characters, or blocks per second, minute, or hour.
2. Corresponding equipment should be indicated; for example, modems, intermediate equipment, or source and sink.

datagram. (1) In packet switching, a self-contained packet, independent of other packets, that carries information sufficient for routing from the originating data terminal equipment (DTE) to the destination DTE without relying on earlier exchanges between the DTEs

and the network. (I) (2) In TCP/IP, the basic unit of information passed across the Internet environment. A datagram contains a source and destination address along with the data. An Internet Protocol (IP) datagram consists of an IP header followed by the transport layer data. (3) See also *packet* and *segment*.

Datagram Delivery Protocol (DDP). In AppleTalk networks, a protocol that provides network connectivity by means of connectionless socket-to-socket delivery service on the internet layer.

DCE ready. In the EIA 232 standard, a signal that indicates to the data terminal equipment (DTE) that the local data circuit-terminating equipment (DCE) is connected to the communication channel and is ready to send data. Synonymous with *data set ready (DSR)*.

DECnet. A network architecture that defines the operation of a family of software modules, databases, and hardware components typically used to tie Digital Equipment Corporation systems together for resource sharing, distributed computation, or remote system configuration. DECnet network implementations follow the Digital Network Architecture (DNA) model.

default. Pertaining to an attribute, condition, value, or option that is assumed when none is explicitly specified. (I)

dependent LU requester (DLUR). An APPN end node or an APPN network node that owns dependent LUs, but requests that a dependent LU server provide the SSCP services for those dependent LUs.

designated router. A router that informs end nodes of the existence and identity of other routers. The selection of the designated router is based upon the router with the highest priority. When several routers share the highest priority, the router with the highest station address is selected.

destination node. The node to which a request or data is sent.

destination port. The 8-port asynchronous adapter that serves as a connection point with a serial service.

destination service access point (DSAP). In SNA and TCP/IP, a logical address that allows a system to route data from a remote device to the appropriate communications support. Contrast with *source service access point (SSAP)*.

device. A mechanical, electrical, or electronic contrivance with a specific purpose.

digital. (1) Pertaining to data that consist of digits. (T) (2) Pertaining to data in the form of digits. (A) (3) Contrast with *analog*.

Digital Network Architecture (DNA). The model for all DECnet hardware and software implementations.

direct memory access (DMA). The system facility that allows a device on the Micro Channel bus to get direct access to the system or bus memory without the intervention of the system processor.

directory. A table of identifiers and references to the corresponding items of data. (I) (A)

directory service (DS). An application service element that translates the symbolic names used by application processes into the complete network addresses used in an OSI environment. (T)

directory services (DS). A control point component of an APPN node that maintains knowledge of the location of network resources.

disable. To make nonfunctional.

disabled. (1) Pertaining to a state of a processing unit that prevents the occurrence of certain types of interruptions. (2) Pertaining to the state in which a transmission control unit or audio response unit cannot accept incoming calls on a line.

domain. (1) That part of a computer network in which the data processing resources are under common control. (T) (2) In Open Systems Interconnection (OSI), a part of a distributed system or a set of managed objects to which a common policy applies. (3) See *Administrative Domain* and *domain name*.

domain name. In the Internet suite of protocols, a name of a host system. A domain name consists of a sequence of subnames separated by a delimiter character. For example, if the fully qualified domain name (FQDN) of a host system is `ralvm7.vnet.ibm.com`, each of the following is a domain name:

- `ralvm7.vnet.ibm.com`
- `vnet.ibm.com`
- `ibm.com`

domain name server. In the Internet suite of protocols, a server program that supplies name-to-address translation by mapping domain names to IP addresses. Synonymous with *name server*.

Domain Name System (DNS). In the Internet suite of protocols, the distributed database system used to map domain names to IP addresses.

dotted decimal notation. The syntactical representation for a 32-bit integer that consists of four 8-bit numbers written in base 10 with periods (dots) separating them. It is used to represent IP addresses.

dump. (1) Data that has been dumped. (T) (2) To copy the contents of all or part of virtual storage for the purpose of collecting error information.

dynamic reconfiguration (DR). The process of changing the network configuration (peripheral PUs and

LUs) without regenerating complete configuration tables or deactivating the affected major node.

Dynamic Routing. Routing using learned routes rather than routes statically configured at initialization.

E

echo. In data communication, a reflected signal on a communications channel. For example, on a communications terminal, each signal is displayed twice, once when entered at the local terminal and again when returned over the communications link. This allows the signals to be checked for accuracy.

EIA 232. In data communication, a specification of the Electronic Industries Association (EIA) that defines the interface between data terminal equipment (DTE) and data circuit-terminating equipment (DCE), using serial binary data interchange.

Electronic Industries Association (EIA). An organization of electronics manufacturers that advances the technological growth of the industry, represents the views of its members, and develops industry standards.

EIA unit. A unit of measure, established by the Electronic Industries Association, equal to 44.45 millimeters (1.75 inches).

encapsulation. (1) In communications, a technique used by layered protocols by which a layer adds control information to the protocol data unit (PDU) from the layer it supports. In this respect, the layer encapsulates the data from the supported layer. In the Internet suite of protocols, for example, a packet would contain control information from the physical layer, followed by control information from the network layer, followed by the application protocol data. (2) See also *data link switching*.

encode. To convert data by the use of a code in such a manner that reconversion to the original form is possible. (T)

end node (EN). (1) See *Advanced Peer-to-Peer Networking (APPN) end node* and *low-entry networking (LEN) end node*. (2) In communications, a node that is frequently attached to a single data link and cannot perform intermediate routing functions.

entry point (EP). In SNA, a type 2.0, type 2.1, type 4, or type 5 node that provides distributed network management support. It sends network management data about itself and the resources it controls to a focal point for centralized processing, and it receives and executes focal-point initiated commands to manage and control its resources.

Ethernet. A 10-Mbps baseband local area network that allows multiple stations to access the transmission medium at will without prior coordination, avoids contention by using carrier sense and deference, and resolves contention by using collision detection and delayed retransmission. Ethernet uses carrier sense multiple access with collision detection (CSMA/CD).

exception. An abnormal condition such as an I/O error encountered in processing a data set or a file.

exception response (ER). In SNA, a protocol requested in the form-of-response-requested field of a request header that directs the receiver to return a response only if the request is unacceptable as received or cannot be processed; that is, a negative response, but not a positive response, can be returned. Contrast with *definite response* and *no response*.

exchange identification (XID). A specific type of basic link unit that is used to convey node and link characteristics between adjacent nodes. XIDs are exchanged between link stations before and during link activation to establish and negotiate link and node characteristics, and after link activation to communicate changes in these characteristics.

explicit route (ER). In SNA, a series of one or more transmission groups that connect two subarea nodes. An explicit route is identified by an origin subarea address, a destination subarea address, an explicit route number, and a reverse explicit route number. Contrast with *virtual route (VR)*.

explorer frame. See *explorer packet*.

explorer packet. In LANs, a packet that is generated by the source host and that traverses the entire source routing part of a LAN, gathering information on the possible paths available to the host.

exterior gateway. In Internet communications, a gateway on one autonomous system that communicates with another autonomous system. Contrast with *interior gateway*.

Exterior Gateway Protocol (EGP). In the Internet suite of protocols, a protocol, used between domains and autonomous systems, that enables network reachability information to be advertised and exchanged. IP network addresses in one autonomous system are advertised to another autonomous system by means of EGP-participating routers. An example of an EGP is the Border Gateway Protocol (BGP). Contrast with Interior Gateway Protocol (IGP).

F

fax. Hardcopy received from a facsimile machine. Synonymous with ***telecopy.***

File Transfer Protocol (FTP). In the Internet suite of protocols, an application layer protocol that uses TCP and Telnet services to transfer bulk-data files between machines or hosts.

flash memory. A data storage device that is programmable, erasable, and does not require continuous power. The chief advantage of flash memory over other programmable and erasable data storage devices is that it can be reprogrammed without being removed from the circuit board.

flow control. (1) In SNA, the process of managing the rate at which data traffic passes between components of the network. The purpose of flow control is to optimize the rate of flow of message units with minimum congestion in the network; that is, to neither overflow the buffers at the receiver or at intermediate routing nodes, nor leave the receiver waiting for more message units. (2) See also *pacing*.

fragment. See *fragmentation*.

fragmentation. (1) The process of dividing a datagram into smaller parts, or fragments, to match the capabilities of the physical medium over which it is to be transmitted. (2) See also *segmenting*.

frame. (1) In Open Systems Interconnection architecture, a data structure pertaining to a particular area of knowledge and consisting of slots that can accept the values of specific attributes and from which inferences can be drawn by appropriate procedural attachments. (T) (2) The unit of transmission in some local area networks, including the IBM Token-Ring Network. It includes delimiters, control characters, information, and checking characters. (3) In SDLC, the vehicle for every command, every response, and all information that is transmitted using SDLC procedures.

frame level. Synonymous with *data link level*. See *link level*.

frame relay. (1) An interface standard describing the boundary between a user's equipment and a fast-packet network. In frame-relay systems, flawed frames are discarded; recovery comes end-to-end rather than hop-by-hop. (2) A technique derived from the integrated services digital network (ISDN) D channel standard. It assumes that connections are reliable and dispenses with the overhead of error detection and control within the network.

front-end processor. A processor such as the IBM 3745 or 3174, that relieves a main frame from the communication control tasks.

G

gateway. (1) A functional unit that interconnects two computer networks with different network architectures. A gateway connects networks or systems of different architectures. A bridge interconnects networks or systems with the same or similar architectures. (T) (2) In the IBM Token-Ring Network, a device and its associated software that connect a local area network to another local area network or a host that uses different logical link protocols. (3) In TCP/IP, synonym for *router*.

general data stream (GDS). The data stream used for conversations in LU 6.2 sessions.

general data stream (GDS) variable. A type of RU substructure that is preceded by an identifier and a length field and includes either application data, user control data, or SNA-defined control data.

H

header. (1) System-defined control information that precedes user data. (2) The portion of a message that contains control information for the message such as one or more destination fields, name of the originating station, input sequence number, character string indicating the type of message, and priority level for the message.

heap memory. The amount of RAM used to dynamically allocate data structures.

Hello. A protocol used by a group of cooperating, trusting routers to allow them to discover minimal delay routes.

hello message. (1) A message sent periodically to establish and test reachability between routers or between routers and hosts. (2) In the Internet suite of protocols, a message defined by the Hello protocol as an Interior Gateway Protocol (IGP).

heuristic. Pertaining to exploratory methods of problem solving in which solutions are discovered by evaluation of the progress made toward the final result.

high-level data link control (HDLC). In data communication, the use of a specified series of bits to control data links in accordance with the International Standards for HDLC: ISO 3309 Frame Structure and ISO 4335 Elements of Procedures.

high-performance routing (HPR). An addition to the Advanced Peer-to-Peer Networking (APPN) architecture that enhances data routing performance and reliability, especially when using high-speed links.

hop. (1) In APPN, a portion of a route that has no intermediate nodes. It consists of only a single transmission group connecting adjacent nodes. (2) To the routing layer, the logical distance between two nodes in a network.

hop count. (1) A metric or measure of distance between two points. (2) In Internet communications, the number of routers that a datagram passes through on its way to its destination. (3) In SNA, a measure of the number of links to be traversed in a path to a destination.

host. In the Internet suite of protocols, an end system. The end system can be any workstation; it does not have to be a mainframe.

hub (intelligent). A wiring concentrator, such as the IBM 8260, that provides bridging and routing functions for LANs with different cables and protocols.

hysteresis. The amount the temperature must change past the set alert threshold before the alert condition is cleared.

I

I-frame. Information frame.

information (I) frame. A frame in I format used for numbered information transfer.

input/output channel. In a data processing system, a functional unit that handles transfer of data between internal and peripheral equipment. (I) (A)

Integrated Digital Network Exchange (IDNX). A processor integrating voice, data, and image applications. It also manages the transmission resources, and connects to multiplexers and network management support systems. It allows integration of equipment from different vendors.

integrated services digital network (ISDN). A digital end-to-end telecommunication network that supports multiple services including, but not limited to, voice and data.

Note: ISDNs are used in public and private network architectures.

interface. (1) A shared boundary between two functional units, defined by functional characteristics, signal characteristics, or other characteristics, as appropriate. The concept includes the specification of the connection of two devices having different functions. (T) (2) Hardware, software, or both, that links systems, programs, or devices.

interior gateway. In Internet communications, a gateway that communicates only with its own autonomous system. Contrast with *exterior gateway*.

Interior Gateway Protocol (IGP). In the Internet suite of protocols, a protocol used to propagate network reachability and routing information within an autonomous system. Examples of IGPs are Routing Information Protocol (RIP) and Open Shortest Path First (OSPF).

intermediate node. A node that is at the end of more than one branch. (T)

intermediate session routing (ISR). A type of routing function within an APPN network node that provides session-level flow control and outage reporting for all sessions that pass through the node but whose end points are elsewhere.

International Organization for Standardization (ISO). An organization of national standards bodies from various countries established to promote development of standards to facilitate international exchange of goods and services, and develop cooperation in intellectual, scientific, technological, and economic activity.

International Telecommunication Union (ITU). The specialized telecommunication agency of the United Nations, established to provide standardized communication procedures and practices, including frequency allocation and radio regulations worldwide.

internet. A collection of networks interconnected by a set of routers that allow them to function as a single, large network. See also *Internet*.

Internet. The internet administered by the Internet Architecture Board (IAB), consisting of large national backbone networks and many regional and campus networks all over the world. The Internet uses the Internet suite of protocols.

Internet address. See *IP address*.

Internet Architecture Board (IAB). The technical body that oversees the development of the Internet suite of protocols that are known as TCP/IP.

Internet Control Message Protocol (ICMP). The protocol used to handle errors and control messages in the Internet Protocol (IP) layer. Reports of problems and incorrect datagram destinations are returned to the original datagram source. ICMP is part of the Internet Protocol.

Internet Control Protocol (ICP). The VIrtual NEtworking System (VINES) protocol that provides exception notifications, metric notifications, and PING support. See also *RouTing update Protocol (RTP)*.

Internet Engineering Task Force (IETF). The task force of the Internet Architecture Board (IAB) that is responsible for solving the short-term engineering needs of the Internet.

Internetwork Packet Exchange (IPX). (1) The network protocol used to connect Novell's servers, or any workstation or router that implements IPX, with other workstations. Although similar to the Internet Protocol (IP), IPX uses different packet formats and terminology. (2) See also *Xerox Network Systems (XNS)*.

Internet Protocol (IP). A connectionless protocol that routes data through a network or interconnected networks. IP acts as an intermediary between the higher protocol layers and the physical network. However, this

protocol does not provide error recovery and flow control and does not guarantee the reliability of the physical network.

interoperability. The capability to communicate, execute programs, or transfer data among various functional units in a way that requires the user to have little or no knowledge of the unique characteristics of those units. (T)

intra-area routing. In Internet communications, the routing of data within an area.

Inverse Address Resolution Protocol (InARP). In the Internet suite of protocols, the protocol used for locating a protocol address through the known hardware address. In a frame-relay context, the data link connection identifier (DLCI) is synonymous with the known hardware address.

IPPN. The interface that other protocols can use to transport data over IP.

IP address. The 32-bit address defined by the Internet Protocol, standard 5, Request for Comments (RFC) 791. It is usually represented in dotted decimal notation.

IP datagram. In the Internet suite of protocols, the fundamental unit of information transmitted through an internet. It contains source and destination addresses, user data, and control information such as the length of the datagram, the header checksum, and flags indicating whether the datagram can be or has been fragmented.

IP router. A device in an IP internet that is responsible for making decisions about the paths over which network traffic will flow. Routing protocols are used to gain information about the network and to determine the best route over which the datagram should be forwarded toward the final destination. The datagrams are routed based on IP destination addresses.

IPXWAN. A Novell protocol that is used to exchange router-to-router information before exchanging standard Internetwork Packet Exchange (IPX) routing information and traffic over wide area networks (WANs).

L

L2TP Access Concentrator (LAC). A device attached to one or more public service telephone network (PSTN) or ISDN lines capable of handling both PPP operation and of the L2TP protocol. The LAC implements the media over which L2TP operates. L2TP passes the traffic to one or more L2TP Network Servers (LNS). L2TP can tunnel any protocol carried by the PPP network.

L2TP Network Server (LNS). An LNS operates on any platform capable that can be a PPP end station. The LNS handles the server side of the L2TP protocol. Since L2TP relies only on the single media over which L2TP tunnels arrive, the LNS has only a single LAN or WAN interface, yet is still able to terminate calls arriving from any the full range of PPP interfaces supported by a LAC. These include asynchronous ISDN, synchronous ISDN, V.120, and other types of connections.

LAN bridge server (LBS). In the IBM Token-Ring Network Bridge Program, the server that keeps statistical information about frames forwarded between two or more rings (through a bridge). The LBS sends these statistics to the appropriate LAN managers through the LAN reporting mechanism (LRM).

LAN Emulation (LE). An ATM Forum standard that supports legacy LAN applications over ATM networks.

LAN Emulation Client (LEC). A LAN Emulation component that represents users of the Emulated LAN.

LAN Emulation Configuration Server (LECS). A LAN Emulation Service component that centralizes and disseminates configuration data.

LAN Emulation Server (LES). A LAN Emulation Service component that resolves LAN Destinations to ATM Addresses.

LAN Network Manager (LNM). An IBM licensed program that enables a user to manage and monitor LAN resources from a central workstation.

LAN segment. (1) Any portion of a LAN (for example, a bus or ring) that can operate independently, but that is connected to other parts of the network by means of bridges. (2) A ring or bus network without bridges.

layer. (1) In network architecture, a group of services that is complete from a conceptual point of view, that is one out of a set of hierarchically arranged groups, and that extends across all systems that conform to the network architecture. (T) (2) In the Open Systems Interconnection reference model, one of seven conceptually complete, hierarchically arranged groups of services, functions, and protocols, that extend across all open systems. (T) (3) In SNA, a grouping of related functions that are logically separate from the functions in other groups. Implementation of the functions in one layer can be changed without affecting functions in other layers.

LE. LAN Emulation. An ATM Forum standard that supports legacy LAN applications over ATM networks.

LEC. LAN Emulation Client. A LAN Emulation component that represents users of the Emulated LAN.

LECS. LAN Emulation Configuration Server. A LAN Emulation Service component that centralizes and disseminates configuration data.

LES. LAN Emulation Server. A LAN Emulation Service component that resolves LAN Destinations to ATM Addresses.

line switching. Synonym for *circuit switching*.

link. The combination of the link connection (the transmission medium) and two link stations, one at each end of the link connection. A link connection can be shared among multiple links in a multipoint or token-ring configuration.

link access protocol balanced (LAPB). A protocol used for accessing an X.25 network at the link level. LAPB is a duplex, asynchronous, symmetric protocol, used in point-to-point communication.

link-attached. (1) Pertaining to devices that are connected to a controlling unit by a data link. (2) Contrast with *channel-attached*. (3) Synonymous with *remote*.

link connection. (1) The physical equipment providing two-way communication between one link station and one or more other link stations; for example, a telecommunication line and data circuit-terminating equipment (DCE). (2) In SNA, synonymous with *data circuit*.

link level. (1) A part of Recommendation X.25 that defines the link protocol used to get data into and out of the network across the full-duplex link connecting the subscriber's machine to the network node. LAP and LAPB are the link access protocols recommended by the CCITT. (2) See *data link level*.

link-state. In routing protocols, the advertised information about the usable interfaces and reachable neighbors of a router or network. The protocol's topological database is formed from the collected link-state advertisements.

link station. (1) The hardware and software components within a node representing a connection to an adjacent node over a specific link. For example, if node A is the primary end of a multipoint line that connects to three adjacent nodes, node A will have three link stations representing the connections to the adjacent nodes. (2) See also *adjacent link station (ALS)*.

local. (1) Pertaining to a device accessed directly without use of a telecommunication line. (2) Contrast with *remote*. (3) Synonym for *channel-attached*.

local area network (LAN). (1) A computer network located on a user's premises within a limited geographical area. Communication within a local area network is not subject to external regulations; however, communication across the LAN boundary may be subject to some form of regulation. (T) (2) A network in which a set of devices are connected to one another for communication and that can be connected to a larger network. (3) See also *Ethernet* and *token ring*. (4) Contrast with *metropolitan area network (MAN)* and *wide area network (WAN)*.

local bridging. A function of a bridge program that allows a single bridge to connect multiple LAN segments without using a telecommunication link. Contrast with *remote bridging*.

local management interface (LMI). See *local management interface (LMI) protocol*.

local management interface (LMI) protocol. In NCP, a set of frame-relay network management procedures and messages used by adjacent frame-relay nodes to exchange line status information over DLCI X'00'. NCP supports both the American National Standards Institute (ANSI) and International Telegraph and Telephone Consultative Committee (ITU-T/CCITT) versions of LMI protocol. These standards refer to LMI protocol as *link integrity verification tests (LIVT)*.

locally administered address. In a local area network, an adapter address that the user can assign to override the universally administered address. Contrast with *universally administered address*.

logical channel. In packet mode operation, a sending channel and a receiving channel that together are used to send and receive data over a data link at the same time. Several logical channels can be established on the same data link by interleaving the transmission of packets.

logical link. A pair of link stations, one in each of two adjacent nodes, and their underlying link connection, providing a single link-layer connection between the two nodes. Multiple logical links can be distinguished while they share the use of the same physical media connecting two nodes. Examples are 802.2 logical links used on local area network (LAN) facilities and LAP E logical links on the same point-to-point physical link between two nodes. The term logical link also includes the multiple X.25 logical channels that share the use of the access link from a DTE to an X.25 network.

logical link control (LLC). The data link control (DLC) LAN sublayer that provides two types of DLC operation for the orderly exchange of information. The first type is connectionless service, which allows information to be sent and received without establishing a link. The LLC sublayer does not perform error recovery or flow control for connectionless service. The second type is connection-oriented service, which requires establishing a link prior to the exchange of information. Connection-oriented service provides sequenced information transfer, flow control, and error recovery.

logical link control (LLC) protocol. In a local area network, the protocol that governs the exchange of transmission frames between data stations independently of how the transmission medium is

shared. (T) The LLC protocol was developed by the IEEE 802 committee and is common to all LAN standards.

logical link control (LLC) protocol data unit. A unit of information exchanged between link stations in different nodes. The LLC protocol data unit contains a destination service access point (DSAP), a source service access point (SSAP), a control field, and user data.

logical unit (LU). A type of network accessible unit that enables users to gain access to network resources and communicate with each other.

loopback test. A test in which signals from a tester are looped at a modem or other network element back to the tester for measurements that determine or verify the quality of the communications path.

low-entry networking (LEN). A capability of nodes to attach directly to one another using basic peer-to-peer protocols to support multiple and parallel sessions between logical units.

low-entry networking (LEN) end node. A LEN node receiving network services from an adjacent APPN network node.

low-entry networking (LEN) node. A node that provides a range of end-user services, attaches directly to other nodes using peer protocols, and derives network services implicitly from an adjacent APPN network node, that is, without the direct use of CP-CP sessions.

M

Management Information Base (MIB). (1) A collection of objects that can be accessed by means of a network management protocol. (2) A definition for management information that specifies the information available from a host or gateway and the operations allowed. (3) In OSI, the conceptual repository of management information within an open system.

management station. In Internet communications, the system responsible for managing all, or a portion of, a network. The management station communicates with network management agents that reside in the managed node by means of a network management protocol, such as the Simple Network Management Protocol (SNMP).

mapping. The process of converting data that is transmitted in one format by the sender into the data format that can be accepted by the receiver.

mask. (1) A pattern of characters used to control retention or elimination of portions of another pattern of characters. (I) (A) (2) To use a pattern of characters to control retention or elimination of portions of another pattern of characters. (I) (A)

maximum transmission unit (MTU). In LANs, the largest possible unit of data that can be sent on a given physical medium in a single frame. For example, the MTU for Ethernet is 1500 bytes.

medium access control (MAC). In LANs, the sublayer of the data link control layer that supports medium-dependent functions and uses the services of the physical layer to provide services to the logical link control (LLC) sublayer. The MAC sublayer includes the method of determining when a device has access to the transmission medium.

medium access control (MAC) protocol. In a local area network, the protocol that governs access to the transmission medium, taking into account the topological aspects of the network, in order to enable the exchange of data between data stations. (T)

medium access control (MAC) sublayer. In a local area network, the part of the data link layer that applies a medium access method. The MAC sublayer supports topology-dependent functions and uses the services of the physical layer to provide services to the logical link control sublayer. (T)

metric. In Internet communications, a value, associated with a route, which is used to discriminate between multiple exit or entry points to the same autonomous system. The route with the lowest metric is preferred.

metropolitan area network (MAN). A network formed by the interconnection of two or more networks which may operate at higher speed than those networks, may cross administrative boundaries, and may use multiple access methods. (T) Contrast with *local area network (LAN)* and *wide area network (WAN)*.

MIB. (1) MIB module. (2) Management Information Base.

MIB object. Synonym for *MIB variable*.

MIB variable. In the Simple Network Management Protocol (SNMP), a specific instance of data defined in a MIB module. Synonymous with *MIB object*.

MIB view. In the Simple Network Management Protocol (SNMP), the collection of managed objects, known to the agent, that is visible to a particular community.

MILNET. The military network that was originally part of ARPANET. It was partitioned from ARPANET in 1984. MILNET provides a reliable network service for military installations.

modem (modulator/demodulator). (1) A functional unit that modulates and demodulates signals. One of

the functions of a modem is to enable digital data to be transmitted over analog transmission facilities. (T) (A) (2) A device that converts digital data from a computer to an analog signal that can be transmitted on a telecommunication line, and converts the analog signal received to data for the computer.

modulo. (1) Pertaining to a modulus; for example, 9 is equivalent to 4 modulo 5. (2) See also *modulus*.

modulus. A number, such as a positive integer, in a relationship that divides the difference between two related numbers without leaving a remainder; for example, 9 and 4 have a modulus of 5 (9 - 4 = 5; 4 - 9 = -5; and 5 divides both 5 and -5 without leaving a remainder).

monitor. (1) A device that observes and records selected activities within a data processing system for analysis. Possible uses are to indicate significant departure from the norm, or to determine levels of utilization of particular functional units. (T) (2) Software or hardware that observes, supervises, controls, or verifies operations of a system. (A) (3) The function required to initiate the transmission of a token on the ring and to provide soft-error recovery in case of lost tokens, circulating frames, or other difficulties. The capability is present in all ring stations.

multicast. (1) Transmission of the same data to a selected group of destinations. (T) (2) A special form of broadcast in which copies of a packet are delivered to only a subset of all possible destinations.

multiple-domain support (MDS). A technique for transporting management services data between management services function sets over LU-LU and CP-CP sessions. See also *multiple-domain support message unit (MDS-MU)*.

multiple-domain support message unit (MDS-MU). The message unit that contains management services data and flows between management services function sets over the LU-LU and CP-CP sessions used by multiple-domain support. This message unit, as well as the actual management services data that it contains, is in general data stream (GDS) format. See also *control point management services unit (CP-MSU)*, *management services unit (MSU)*, and *network management vector transport (NMVT)*.

N

Name Binding Protocol (NBP). In AppleTalk networks, a protocol that provides name translation function from the AppleTalk entity (resource) name (character string) into an AppleTalk IP address (16-bit number) on the transport layer.

name resolution. In Internet communications, the process of mapping a machine name to the corresponding Internet Protocol (IP) address. See also *Domain Name System (DNS)*.

name server. In the Internet suite of protocols, synonym for *domain name server*.

nearest active upstream neighbor (NAUN). In the IBM Token-Ring Network, the station sending data directly to a given station on the ring.

neighbor. A router on a common subnetwork that has been designated by a network administrator to receive routing information.

NetBIOS. Network Basic Input/Output System. A standard interface to networks, IBM personal computers (PCs), and compatible PCs, that is used on LANs to provide message, print-server, and file-server functions. Application programs that use NetBIOS do not need to handle the details of LAN data link control (DLC) protocols.

network. (1) A configuration of data processing devices and software connected for information interchange. (2) A group of nodes and the links interconnecting them.

Network Access Server (NAS). A device providing temporary, on-demand network access to users. This access is point-to-point using PSTN or ISDN lines.

network accessible unit (NAU). A logical unit (LU), physical unit (PU), control point (CP), or system services control point (SSCP). It is the origin or the destination of information transmitted by the path control network. Synonymous with *network addressable unit*.

network address. According to ISO 7498-3, a name, unambiguous within the OSI environment, that identifies a set of network service access points.

network addressable unit (NAU). Synonym for *network accessible unit*.

network architecture. The logical structure and operating principles of a computer network. (T)

Note: The operating principles of a network include those of services, functions, and protocols.

network congestion. An undesirable overload condition caused by traffic in excess of what a network can handle.

network identifier. (1) In TCP/IP, that part of the IP address that defines a network. The length of the network ID depends on the type of network class (A, B, or C). (2) A 1- to 8-byte customer-selected name or an 8-byte IBM-registered name that uniquely identifies a specific subnetwork.

Network Information Center (NIC). In Internet communications, local, regional, and national groups

throughout the world who provide assistance, documentation, training, and other services to users.

network layer. In Open Systems Interconnection (OSI) architecture, the layer that is responsible for routing, switching, and link-layer access across the OSI environment.

network management. The process of planning, organizing, and controlling a communication-oriented data processing or information system.

network management station. In the Simple Network Management Protocol (SNMP), a station that executes management application programs that monitor and control network elements.

network management vector transport (NMVT). A management services request/response unit (RU) that flows over an active session between physical unit management services and control point management services (SSCP-PU session).

network manager. A program or group of programs that is used to monitor, manage, and diagnose the problems of a network.

network node (NN). See *Advanced Peer-to-Peer Networking (APPN) network node*.

network user address (NUA). In X.25 communications, the X.121 address containing up to 15 binary code digits.

node. (1) In a network, a point at which one or more functional units connect channels or data circuits. (I) (2) Any device, attached to a network, that transmits and receives data.

noncanonical address. In LANs, a format for the transmission of medium access control (MAC) addresses for token-ring adapters. In noncanonical format, the most significant (leftmost) bit of each address byte is transmitted first. Contrast with *canonical address*.

Non-Return-to-Zero Changes-on-Ones Recording (NRZ-1). A recording method in which the ones are represented by a change in the condition of magnetization, and zeros are represented by the absence of change. Only the one signals are explicitly recorded. (Previously called ***non-return-to-zero inverted***, NRZI, recording.)

nonseed router. In AppleTalk networks, a router that acquires network number range and zone list information from a seed router attached to the same network.

O

Open Shortest Path First (OSPF). In the Internet suite of protocols, a function that provides intradomain information transfer. An alternative to the Routing Information Protocol (RIP), OSPF allows the lowest-cost routing and handles routing in large regional or corporate networks.

Open Systems Interconnection (OSI). (1) The interconnection of open systems in accordance with standards of the International Organization for Standardization (ISO) for the exchange of information. (T) (A) (2) The use of standardized procedures to enable the interconnection of data processing systems.

Note: OSI architecture establishes a framework for coordinating the development of current and future standards for the interconnection of computer systems. Network functions are divided into seven layers. Each layer represents a group of related data processing and communication functions that can be carried out in a standard way to support different applications.

Open Systems Interconnection (OSI) architecture. Network architecture that adheres to that particular set of ISO standards that relates to Open Systems Interconnection. (T)

Open Systems Interconnection (OSI) reference model. A model that describes the general principles of the Open Systems Interconnection, as well as the purpose and the hierarchical arrangement of its seven layers. (T)

origin. An external logical unit (LU) or application program from which a message or other data originates. See also *destination*.

orphan circuit. A non-configured circuit whose availability is learned dynamically.

P

pacing. (1) A technique by which a receiving component controls the rate of transmission of a sending component to prevent overrun or congestion. (2) See also *flow control*, *receive pacing*, *send pacing*, *session-level pacing*, and *virtual route (VR) pacing*.

packet. In data communication, a sequence of binary digits, including data and control signals, that is transmitted and switched as a composite whole. The data, control signals, and, possibly, error control information are arranged in a specific format. (I)

packet internet groper (PING). (1) In Internet communications, a program used in TCP/IP networks to test the ability to reach destinations by sending the destinations an Internet Control Message Protocol (ICMP) echo request and waiting for a reply. (2) In communications, a test of reachability.

packet loss ratio. The probability that a packet will not reach its destination or not reach it within a specified time.

packet mode operation. Synonym for *packet switching*.

packet switching. (1) The process of routing and transferring data by means of addressed packets so that a channel is occupied only during transmission of a packet. On completion of the transmission, the channel is made available for transfer of other packets. (I) (2) Synonymous with *packet mode operation*. See also *circuit switching*.

parallel bridges. A pair of bridges connected to the same LAN segment, creating redundant paths to the segment.

parallel transmission groups. Multiple transmission groups between adjacent nodes, with each group having a distinct transmission group number.

path. (1) In a network, any route between any two nodes. A path may include more than one branch. (T) (2) The series of transport network components (path control and data link control) that are traversed by the information exchanged between two network accessible units. See also *explicit route (ER)*, *route extension*, and *virtual route (VR)*.

path control (PC). The function that routes message units between network accessible units in the network and provides the paths between them. It converts the basic information units (BIUs) from transmission control (possibly segmenting them) into path information units (PIUs) and exchanges basic transmission units containing one or more PIUs with data link control. Path control differs by node type: some nodes (APPN nodes, for example) use locally generated session identifiers for routing, and others (subarea nodes) use network addresses for routing.

path cost. In link-state routing protocols, the sum of the link costs along the path between two nodes or networks.

path information unit (PIU). A message unit consisting of a transmission header (TH) alone, or a TH followed by a basic information unit (BIU) or a BIU segment.

pattern-matching character. A special character such as an asterisk (*) or a question mark (?) that can be used to represent one or more characters. Any character or set of characters can replace a pattern-matching character. Synonymous with *global character* and *wildcard character*.

permanent virtual circuit (PVC). In X.25 and frame-relay communications, a virtual circuit that has a logical channel permanently assigned to it at each data terminal equipment (DTE). Call-establishment protocols are not required. Contrast with *switched virtual circuit (SVC)*.

physical circuit. A circuit established without multiplexing. See also *data circuit*. Contrast with *virtual circuit*.

physical layer. In the Open Systems Interconnection reference model, the layer that provides the mechanical, electrical, functional, and procedural means to establish, maintain, and release physical connections over the transmission medium. (T)

physical unit (PU). (1) The component that manages and monitors the resources (such as attached links and adjacent link stations) associated with a node, as requested by an SSCP via an SSCP-PU session. An SSCP activates a session with the physical unit in order to indirectly manage, through the PU, resources of the node such as attached links. This term applies to type 2.0, type 4, and type 5 nodes only. (2) See also *peripheral PU* and *subarea PU*.

ping command. The command that sends an Internet Control Message Protocol (ICMP) echo-request packet to a gateway, router, or host with the expectation of receiving a reply.

Point-to-Point Protocol (PPP). A protocol that provides a method for encapsulating and transmitting packets over serial point-to-point links.

polling. (1) On a multipoint connection or a point-to-point connection, the process whereby data stations are invited, one at a time, to transmit. (I) (2) Interrogation of devices for such purposes as to avoid contention, to determine operational status, or to determine readiness to send or receive data. (A)

port. (1) An access point for data entry or exit. (2) A connector on a device to which cables for other devices such as display stations and printers are attached. (3) The representation of a physical connection to the link hardware. A port is sometimes referred to as an adapter; however, there can be more than one port on an adapter. There may be one or more ports controlled by a single DLC process. (4) In the Internet suite of protocols, a 16-bit number used to communicate between TCP or the User Datagram Protocol (UDP) and a higher-level protocol or application. Some protocols, such as File Transfer Protocol (FTP) and Simple Mail Transfer Protocol (SMTP), use the same well-known port number in all TCP/IP implementations. (5) An abstraction used by transport protocols to distinguish among multiple destinations within a host machine. (6) Synonymous with *socket*.

port number. In Internet communications, the identification of an application entity to the transport service.

private branch exchange (PBX). A private telephone exchange for transmission of calls to and from the public telephone network.

problem determination. The process of determining the source of a problem; for example, a program component, machine failure, telecommunication facilities, user or contractor-installed programs or equipment, environmental failure such as a power loss, or user error.

program temporary fix (PTF). A temporary solution or bypass of a problem diagnosed by IBM in a current unaltered release of the program.

protocol. (1) A set of semantic and syntactic rules that determine the behavior of functional units in achieving communication. (I) (2) In Open Systems Interconnection architecture, a set of semantic and syntactic rules that determine the behavior of entities in the same layer in performing communication functions. (T) (3) In SNA, the meanings of, and the sequencing rules for, requests and responses used for managing the network, transferring data, and synchronizing the states of network components. Synonymous with *line control discipline* and *line discipline*. See *bracket protocol* and *link protocol*.

protocol data unit (PDU). A unit of data specified in a protocol of a given layer and consisting of protocol control information of this layer, and possibly user data of this layer. (T)

R

Rapid Transport Protocol (RTP) connection. In high-performance routing (HPR), the connection established between the endpoints of the route to transport session traffic.

reachability. The ability of a node or a resource to communicate with another node or resource.

read-only memory (ROM). Memory in which stored data cannot be modified by the user except under special conditions.

real-time processing. The manipulation of data that are required, or generated, by some process while the process is in operation. Usually the results are used to influence the process, and perhaps related processes, while it is occurring.

reassembly. In communications, the process of putting segmented packets back together after they have been received.

receive not ready (RNR). In communications, a data link command or response that indicates a temporary condition of being unable to accept incoming frames.

receive not ready (RNR) packet. See *RNR packet*.

received line signal detector (RLSD). In the EIA 232 standard, a signal that indicates to the data terminal equipment (DTE) that it is receiving a signal from the remote data circuit-terminating equipment (DCE). Synonymous with *carrier detect* and *data carrier detect (DCD)*.

Recognized Private Operating Agency (RPOA). Any individual, company, or corporation, other than a government department or service, that operates a telecommunication service and is subject to the obligations undertaken in the Convention of the International Telecommunication Union and in the Regulations; for example, a communication common carrier.

reduced instruction-set computer (RISC). A computer that uses a small, simplified set of frequently used instructions for rapid execution.

remote. (1) Pertaining to a system, program, or device that is accessed through a telecommunication line. (2) Synonym for *link-attached*. (3) Contrast with *local*.

remote bridging. The function of a bridge that allows two bridges to connect multiple LANs using a telecommunication link. Contrast with *local bridging*.

Remote Execution Protocol (REXEC). A protocol that allows the execution of a command or program on any host in the network. The local host receives the results of the command execution.

Request for Comments (RFC). In Internet communications, the document series that describes a part of the Internet suite of protocols and related experiments. All Internet standards are documented as RFCs.

reset. On a virtual circuit, reinitialization of data flow control. At reset, all data in transit are eliminated.

reset request packet. In X.25 communications, a packet transmitted by the data terminal equipment (DTE) to the data circuit-terminating equipment (DCE) to request that a virtual call or a permanent virtual circuit be reset. The reason for the request can also be specified in the packet.

ring. See *ring network*.

ring network. (1) A network in which every node has exactly two branches connected to it and in which there are exactly two paths between any two nodes. (T) (2) A network configuration in which devices are connected by unidirectional transmission links to form a closed path.

ring segment. A section of a ring that can be isolated (by unplugging connectors) from the rest of the ring. See *LAN segment*.

rlogin (remote login). A service, offered by Berkeley UNIX-based systems, that allows authorized users of one machine to connect to other UNIX systems across an internet and interact as if their terminals were connected directly. The rlogin software passes information about the user's environment (for example, terminal type) to the remote machine.

RNR packet. A packet used by a data terminal equipment (DTE) or by a data circuit-terminating equipment (DCE) to indicate a temporary inability to accept additional packets for a virtual call or permanent virtual circuit.

root bridge. The bridge that is the root of a spanning tree formed between other active bridges in the bridging network. The root bridge originates and transmits bridge protocol data units (BPDUs) to other active bridges to maintain the spanning tree topology. It is the bridge with the highest priority in the network.

route. (1) An ordered sequence of nodes and transmission groups (TGs) that represent a path from an origin node to a destination node traversed by the traffic exchanged between them. (2) The path that network traffic uses to get from source to destination.

route bridge. A function of an IBM bridge program that allows two bridge computers to use a telecommunication link to connect two LANs. Each bridge computer is connected directly to one of the LANs, and the telecommunication link connects the two bridge computers.

route extension (REX). In SNA, the path control network components, including a peripheral link, that make up the portion of a path between a subarea node and a network addressable unit (NAU) in an adjacent peripheral node. See also *explicit route (ER)*, *path*, and *virtual route (VR)*.

Route Selection control vector (RSCV). A control vector that describes a route within an APPN network. The RSCV consists of an ordered sequence of control vectors that identify the TGs and nodes that make up the path from an origin node to a destination node.

router. (1) A computer that determines the path of network traffic flow. The path selection is made from several paths based on information obtained from specific protocols, algorithms that attempt to identify the shortest or best path, and other criteria such as metrics or protocol-specific destination addresses. (2) An attaching device that connects two LAN segments, which use similar or different architectures, at the reference model network layer. (3) In OSI terminology, a function that determines a path by which an entity can be reached. (4) In TCP/IP, synonymous with *gateway*. (5) Contrast with *bridge*.

routing. (1) The assignment of the path by which a message is to reach its destination. (2) In SNA, the forwarding of a message unit along a particular path through a network, as determined by parameters carried in the message unit, such as the destination network address in a transmission header.

routing domain. In Internet communications, a group of intermediate systems that use a routing protocol so that the representation of the overall network is the same within each intermediate system. Routing domains are connected to each other by exterior links.

Routing Information Protocol (RIP). In the Internet suite of protocols, an interior gateway protocol used to exchange intradomain routing information and to determine optimum routes between internet hosts. RIP determines optimum routes on the basis of route metrics, not link transmission speed.

routing loop. A situation that occurs when routers circulate information among themselves until convergence occurs or until the networks involved are considered unreachable.

routing protocol. A technique used by a router to find other routers and to remain up to date about the best way to get to reachable networks.

routing table. A collection of routes used to direct datagram forwarding or to establish a connection. The information is passed among routers to identify network topology and destination feasibility.

Routing Table Maintenance Protocol (RTMP). In AppleTalk networks, a protocol that provides routing information generation and maintenance on the transport layer by means of the AppleTalk routing table. The AppleTalk routing table directs packet transmission through the internet from source socket to destination socket.

RouTing update Protocol (RTP). The VIrtual NEtworking System (VINES) protocol that maintains the routing database and allows the exchange of routing information between VINES nodes. See also *Internet Control Protocol (ICP)*.

rsh. A variant of the rlogin command that invokes a command interpreter on a remote UNIX machine and passes the command-line arguments to the command interpreter, skipping the login step completely.

S

SAP. See service access point.

seed router. In AppleTalk networks, a router that maintains configuration data (network range numbers and zone lists, for example) for the network. Each network must have at least one seed router. The seed router must be initially set up using the configurator tool. Contrast with *nonseed router*.

segment. (1) A section of cable between components or devices. A segment may consist of a single patch cable, several patch cables that are connected, or a combination of building cable and patch cables that are connected. (2) In Internet communications, the unit of transfer between TCP functions in different machines. Each segment contains control and data fields; the current byte-stream position and actual data bytes are identified along with a checksum to validate received data.

segmenting. In OSI, a function performed by a layer to map one protocol data unit (PDU) from the layer it supports into multiple PDUs.

sequence number. In communications, a number assigned to a particular frame or packet to control the transmission flow and receipt of data.

Serial Line Internet Protocol (SLIP). A protocol used over a point-to-point connection between two IP hosts over a serial line, for example, a serial cable or an RS232 connection into a modem, over a telephone line.

server. A functional unit that provides shared services to workstations over a network; for example, a file server, a print server, a mail server. (T)

service access point (SAP). (1) In Open Systems Interconnection (OSI) architecture, the point at which the services of a layer are provided by an entity of that layer to an entity of the next higher layer. (T) (2) A logical point made available by an adapter where information can be received and transmitted. A single service access point can have many links terminating in it.

Service Advertising Protocol (SAP). In Internetwork Packet Exchange (IPX), a protocol that provides the following:

- A mechanism that allows IPX servers on an internet to advertise their services by name and type. Servers using this protocol have their name, service type, and address recorded in all file servers running NetWare.
- A mechanism that allows a workstation to broadcast a query to discover the identities of all servers of all types, all servers of a specific type, or the nearest server of a specific type.
- A mechanism that allows a workstation to query any file server running NetWare to discover the names and addresses of all servers of a specific type.

session. (1) In network architecture, for the purpose of data communication between functional units, all the activities which take place during the establishment, maintenance, and release of the connection. (T) (2) A logical connection between two network accessible units (NAUs) that can be activated, tailored to provide various protocols, and deactivated, as requested. Each session is uniquely identified in a transmission header (TH) accompanying any transmissions exchanged during the session. (3) In L2TP, L2TP creates a session when an end-to-end PPP connection is attempted between a dial user and the LNS; regardless of whether the user initiates the session or the LNS initiates an outbound call. The datagrams for th session are sent over the tunnel between the LAC and LNS. The LNS and LAC maintain the state information for each user attached to an LAC.

Simple Network Management Protocol (SNMP). In the Internet suite of protocols, a network management protocol that is used to monitor routers and attached networks. SNMP is an application layer protocol. Information on devices managed is defined and stored in the application's Management Information Base (MIB).

SNA management services (SNA/MS). The services provided to assist in management of SNA networks.

socket. (1) An endpoint for communication between processes or application programs. (2) The abstraction provided by the University of California's Berkeley Software Distribution (commonly called Berkeley UNIX or BSD UNIX) that serves as an endpoint for communication between processes or applications.

source route bridging. In LANs, a bridging method that uses the routing information field in the IEEE 802.5 medium access control (MAC) header of a frame to determine which rings or token-ring segments the frame must transit. The routing information field is inserted into the MAC header by the source node. The information in the routing information field is derived from explorer packets generated by the source host.

source routing. In LANs, a method by which the sending station determines the route the frame will follow and includes the routing information with the frame. Bridges then read the routing information to determine whether they should forward the frame.

source service access point (SSAP). In SNA and TCP/IP, a logical address that allows a system to send data to a remote device from the appropriate communications support. Contrast with *destination service access point (DSAP)*.

spanning tree. In LAN contexts, the method by which bridges automatically develop a routing table and update that table in response to changing topology to ensure that there is only one route between any two LANs in the bridged network. This method prevents packet looping, where a packet returns in a circuitous route back to the sending router.

sphere of control (SOC). The set of control point domains served by a single management services focal point.

sphere of control (SOC) node. A node directly in the sphere of control of a focal point. A SOC node has exchanged management services capabilities with its

focal point. An APPN end node can be a SOC node if it supports the function to exchange management services capabilities.

split horizon. A technique for minimizing the time to achieve network convergence. A router records the interface over which it received a particular route and does not propagate its information about the route back over the same interface.

spoofing. For data links, a technique in which a protocol initiated from an end station is acknowledged and processed by an intermediate node on behalf of the final destination. In IBM 6611 data link switching, for example, SNA frames are encapsulated into TCP/IP packets for transport across a non-SNA wide area network, unpacked by another IBM 6611, and passed to the final destination. A benefit of spoofing is the prevention of end-to-end session timeouts.

standard MIB. In the Simple Network Management Protocol (SNMP), a MIB module that is located under the management branch of the Structure of Management Information (SMI) and that is considered a standard by the Internet Engineering Task Force (IETF).

static route. The route between hosts, networks, or both that is manually entered into a routing table.

station. An input or output point of a system that uses telecommunication facilities; for example, one or more systems, computers, terminals, devices, and associated programs at a particular location that can send or receive data over a telecommunication line.

StreetTalk. In the VIrtual NEtworking System (VINES), a unique network-wide naming and addressing system that allows users to locate and access any resource on the network without knowing the network topology. See also *Internet Control Protocol (ICP)* and *RouTing update Protocol (RTP)*.

Structure of Management Information (SMI). (1) In the Simple Network Management Protocol (SNMP), the rules used to define the objects that can be accessed by means of a network management protocol. (2) In OSI, the set of standards relating to management information. The set includes the *Management Information Model* and the *Guidelines for the Definition of Managed Objects*

subarea. A portion of the SNA network consisting of a subarea node, attached peripheral nodes, and associated resources. Within a subarea node, all network accessible units (NAUs), links, and adjacent link stations (in attached peripheral or subarea nodes) that are addressable within the subarea share a common subarea address and have distinct element addresses.

subnet. (1) In TCP/IP, a part of a network that is identified by a portion of the IP address. (2) Synonym for *subnetwork*.

subnet address. In Internet communications, an extension to the basic IP addressing scheme where a portion of the host address is interpreted as the local network address.

subnet mask. Synonym for *address mask*.

subnetwork. (1) Any group of nodes that have a set of common characteristics, such as the same network ID. (2) Synonymous with *subnet*.

Subnetwork Access Protocol (SNAP). In LANs, a 5-byte protocol discriminator that identifies the non-IEEE standard protocol family to which a packet belongs. The SNAP value is used to differentiate between protocols that use `$AA` as their service access point (SAP) value.

subnetwork mask. Synonym for *address mask*.

subsystem. A secondary or subordinate system, usually capable of operating independently of, or asynchronously with, a controlling system. (T)

switched virtual circuit (SVC). An X.25 circuit that is dynamically established when needed. The X.25 equivalent of a switched line. Contrast with *permanent virtual circuit (PVC)*.

synchronous. (1) Pertaining to two or more processes that depend upon the occurrence of specific events such as common timing signals. (T) (2) Occurring with a regular or predictable time relationship.

Synchronous Data Link Control (SDLC). (1) A discipline conforming to subsets of the Advanced Data Communication Control Procedures (ADCCP) of the American National Standards Institute (ANSI) and High-level Data Link Control (HDLC) of the International Organization for Standardization, for managing synchronous, code-transparent, serial-by-bit information transfer over a link connection. Transmission exchanges may be duplex or half-duplex over switched or nonswitched links. The configuration of the link connection may be point-to-point, multipoint, or loop. (I) (2) Contrast with *binary synchronous communication (BSC)*.

SYNTAX. In the Simple Network Management Protocol (SNMP), a clause in the MIB module that defines the abstract data structure that corresponds to a managed object.

system. In data processing, a collection of people, machines, and methods organized to accomplish a set of specific functions. (I) (A)

system configuration. A process that specifies the devices and programs that form a particular data processing system.

system services control point (SSCP). A component within a subarea network for managing the configuration, coordinating network operator and

problem determination requests, and providing directory services and other session services for users of the network. Multiple SSCPs, cooperating as peers with one another, can divide the network into domains of control, with each SSCP having a hierarchical control relationship to the physical units and logical units within its own domain.

Systems Network Architecture (SNA). The description of the logical structure, formats, protocols, and operational sequences for transmitting information units through, and controlling the configuration and operation of, networks. The layered structure of SNA allows the ultimate origins and destinations of information, that is, the users, to be independent of and unaffected by the specific SNA network services and facilities used for information exchange.

T

TCP/IP. (1) Transmission Control Protocol/Internet Protocol. (2) A UNIX-like/Ethernet-based system-interconnect protocol originally developed by the US Department of Defense. TCP/IP facilitated ARPANET (Advanced Research Projects Agency Network), a packet-switched research network for which layer 4 was TCP and layer 3, IP.

Telnet. In the Internet suite of protocols, a protocol that provides remote terminal connection service. It allows users of one host to log on to a remote host and interact as directly attached terminal users of that host.

threshold. (1) In IBM bridge programs, a value set for the maximum number of frames that are not forwarded across a bridge due to errors, before a “threshold exceeded” occurrence is counted and indicated to network management programs. (2) An initial value from which a counter is decremented to 0, or a value to which a counter is incremented or decremented from an initial value.

throughput class. In packet switching, the speed at which data terminal equipment (DTE) packets travel through the packet switching network.

time to live (TTL). A technique used by best-effort delivery protocols to inhibit endlessly looping packets. The packet is discarded if the TTL counter reaches 0.

timeout. (1) An event that occurs at the end of a predetermined period of time that began at the occurrence of another specified event. (I) (2) A time interval allotted for certain operations to occur; for example, response to polling or addressing before system operation is interrupted and must be restarted.

token. (1) In a local area network, the symbol of authority passed successively from one data station to another to indicate the station temporarily in control of the transmission medium. Each data station has an opportunity to acquire and use the token to control the medium. A token is a particular message or bit pattern that signifies permission to transmit. (T) (2) In LANs, a sequence of bits passed from one device to another along the transmission medium. When the token has data appended to it, it becomes a frame.

token ring. (1) According to IEEE 802.5, network technology that controls media access by passing a token (special packet or frame) between media-attached stations. (2) IEEE 802.5 network with a ring topology that passes tokens from one attaching ring station (node) to another. (3) See also *local area network (LAN)*.

token-ring network. (1) A ring network that allows unidirectional data transmission between data stations, by a token passing procedure, such that the transmitted data return to the transmitting station. (T) (2) A network that uses a ring topology, in which tokens are passed in a circuit from node to node. A node that is ready to send can capture the token and insert data for transmission.

topology. In communications, the physical or logical arrangement of nodes in a network, especially the relationships among nodes and the links between them.

topology database update (TDU). A message about a new or changed link or node that is broadcast among APPN network nodes to maintain the network topology database, which is fully replicated in each network node. A TDU contains information that identifies the following:

- The sending node
- The node and link characteristics of various resources in the network
- The sequence number of the most recent update for each of the resources described.

trace. (1) A record of the execution of a computer program. It exhibits the sequences in which the instructions were executed. (A) (2) For data links, a record of the frames and bytes transmitted or received.

transceiver (transmitter-receiver). In LANs, a physical device that connects a host interface to a local area network, such as Ethernet. Ethernet transceivers contain electronics that apply signals to the cable and that sense collisions.

Transmission Control Protocol (TCP). A communications protocol used in the Internet and in any network that follows the U.S. Department of Defense standards for internetwork protocol. TCP provides a reliable host-to-host protocol between hosts in packet-switched communications networks and in interconnected systems of such networks. It uses the Internet Protocol (IP) as the underlying protocol.

Transmission Control Protocol/Internet Protocol (TCP/IP). A set of communications protocols that support peer-to-peer connectivity functions for both local and wide area networks.

transmission group (TG). (1) A connection between adjacent nodes that is identified by a transmission group number. (2) In a subarea network, a single link or a group of links between adjacent nodes. When a transmission group consists of a group of links, the links are viewed as a single logical link, and the transmission group is called a *multilink transmission group (MLTG)*. A *mixed-media multilink transmission group (MMMLTG)* is one that contains links of different medium types (for example, token-ring, switched SDLC, nonswitched SDLC, and frame-relay links). (3) In an APPN network, a single link between adjacent nodes. (4) See also *parallel transmission groups*.

transmission header (TH). Control information, optionally followed by a basic information unit (BIU) or a BIU segment, that is created and used by path control to route message units and to control their flow within the network. See also *path information unit*.

transparent bridging. In LANs, a method for tying individual local area networks together through the medium access control (MAC) level. A transparent bridge stores the tables that contain MAC addresses so that frames seen by the bridge can be forwarded to another LAN if the tables indicate to do so.

transport layer. In the Open Systems Interconnection reference model, the layer that provides a reliable end-to-end data transfer service. There may be relay open systems in the path. (T) See also *Open Systems Interconnection reference model*.

trap. In the Simple Network Management Protocol (SNMP), a message sent by a managed node (agent function) to a management station to report an exception condition.

Tunnel. A tunnel is defined by an LNS-LAC pair. The tunnel carries PPP datagrams between the LAC and the LNS. A single tunnel can multiplex many sessions. A control connection operating over the same tunnel controls the establishment, release, and maintenance of all sessions and of the tunnel itself.

tunneling. To treat a transport network as though it were a single communication link or LAN. See also *encapsulation*.

T1. In the United States, a 1.544-Mbps public access line. It is available in twenty-four 64-Kbps channels. The European version (E1) transmits 2.048 Mbps.

U

universally administered address. In a local area network, the address permanently encoded in an adapter at the time of manufacture. All universally administered addresses are unique. Contrast with *locally administered address*.

User Datagram Protocol (UDP). In the Internet suite of protocols, a protocol that provides unreliable, connectionless datagram service. It enables an application program on one machine or process to send a datagram to an application program on another machine or process. UDP uses the Internet Protocol (IP) to deliver datagrams.

V

V.24. In data communication, a specification of the CCITT that defines the list of definitions for interchange circuits between data terminal equipment (DTE) and data circuit-terminating equipment (DCE).

V.25. In data communication, a specification of the CCITT that defines the automatic answering equipment and parallel automatic calling equipment on the General Switched Telephone Network, including procedures for disabling of echo controlled devices for both manually and automatically established calls.

V.34. An ITU-T Recommendation for modem communication over standard commercially available voice-grade 33.6-Kbps (and slower) channels.

V.35. In data communication, a specification of the CCITT that defines the list of definitions for interchange circuits between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) at various data rates.

V.36. In data communication, a specification of the CCITT that defines the list of definitions for interchange circuits between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) at rates of 48, 56, 64, or 72 kilobits per second.

version. A separately licensed program that usually has significant new code or new function.

VINES. VIrtual NEtworking System.

virtual circuit. (1) In packet switching, the facilities provided by a network that give the appearance to the user of an actual connection. (T) See also *data circuit*. Contrast with *physical circuit*. (2) A logical connection established between two DTEs.

virtual connection. In frame relay, the return path of a potential connection.

virtual link. In Open Shortest Path First (OSPF), a point-to-point interface that connects border routers that are separated by a non-backbone transit area. Because area routers are part of the OSPF backbone, the virtual link connects the backbone. The virtual links ensure that the OSPF backbone does not become discontinuous.

VIrtual NEtworking System (VINES). The network operating system and network software from Banyan Systems, Inc. In a VINES network, virtual linking allows all devices and services to appear to be directly connected to each other, when they may actually be thousands of miles apart. See also *StreetTalk*.

virtual route (VR). (1) In SNA, either (a) a logical connection between two subarea nodes that is physically realized as a particular explicit route or (b) a logical connection that is contained wholly within a subarea node for intranode sessions. A virtual route between distinct subarea nodes imposes a transmission priority on the underlying explicit route, provides flow control through virtual route pacing, and provides data integrity through sequence numbering of path information units (PIUs). (2) Contrast with *explicit route (ER)*. See also *path* and *route extension (REX)*.

W

wide area network (WAN). (1) A network that provides communication services to a geographic area larger than that served by a local area network or a metropolitan area network, and that may use or provide public communication facilities. (T) (2) A data communication network designed to serve an area of hundreds or thousands of miles; for example, public and private packet-switching networks, and national telephone networks. (3) Contrast with *local area network (LAN)* and *metropolitan area network (MAN)*.

wildcard character. Synonym for *pattern-matching character*.

X

X.21. An International Telegraph and Telephone Consultative Committee (CCITT) recommendation for a general-purpose interface between data terminal equipment and data circuit-terminating equipment for synchronous operations on a public data network.

X.25. (1) An International Telegraph and Telephone Consultative Committee (CCITT) recommendation for the interface between data terminal equipment and packet-switched data networks. (2) See also *packet switching*.

Xerox Network Systems (XNS). The suite of internet protocols developed by the Xerox Corporation. Although similar to TCP/IP protocols, XNS uses different packet formats and terminology. See also *Internetwork Packet Exchange (IPX)*.

Z

zone. In AppleTalk networks, a subset of nodes within an internet.

Zone Information Protocol (ZIP). In AppleTalk networks, a protocol that provides zone management service by maintaining a mapping of the zone names and network numbers across the internet on the session layer.

zone information table (ZIT). A listing of network numbers and their associated zone name mappings in the internet. This listing is maintained by each internet router in an AppleTalk internet.

Index

A

B

C

D

E

F

G

I

K

L

M

N

O

P

Q

R

Readers' Comments — We'd Like to Hear from You

Nways Multiprotocol Routing Services
Using and Configuring Features
Version 3.3

Publication No. SC30-3992-01

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

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

Name _____ Address _____

Company or Organization _____

Phone No. _____

Readers' Comments — We'd Like to Hear from You
SC30-3992-01

IBM

Cut or Fold
Along Line

Fold and Tape **Please do not staple** Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Corporation
Design & Information Development
Department CGF/Bldg. 656
PO Box 12195
Research Triangle Park, NC 27709-9990

Fold and Tape **Please do not staple** Fold and Tape

SC30-3992-01

Cut or Fold
Along Line

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SC30-3992-01

Spine information:

Nways Multiprotocol Routing Services

MRS V3.3 Using Features