Distributed Routing Software

Routing Protocols User’s Guide

Part Number: AA-QL2DE-TE

December 1997

This guide explains how to configure and monitor the Distributed Routing Software routing protocols shipped with your RouteAbout Access router.

Revision/Update Information: This is a revised manual.
Software Version: Distributed Routing Software V3.0

Digital Equipment Corporation
Maynard, Massachusetts

Digital Equipment Corporation makes no representations that the use of its products in the manner described in this publication will not infringe on existing or future patent rights, nor do the descriptions contained in this publication imply the granting of licenses to make, use, or sell equipment or software in accordance with the description.

Possession, use, or copying of the software described in this publication is authorized only pursuant to a valid written license from DIGITAL or an authorized sublicensor.

The following are trademarks of Digital Equipment Corporation: clearVISN, DEC, DECnet, DECswitch, DIGITAL, OpenVMS, PATHWORKS, RouteAbout, ThinWire, VAX, VAXcluster, VMS, VT, and the DIGITAL logo.

The following are third-party trademarks:

Apollo is a registered trademark of Apollo Computer, Inc., a subsidiary of Hewlett-Packard Company.

AppleTalk, EtherTalk, and LocalTalk are registered trademarks of Apple Computer, Inc.

Banyan and Vines are registered trademarks of Banyan Systems, Inc.

BSD is a trademark of the University of California, Berkeley, CA.

IBM is a registered trademark of International Business Machines Corporation.

Intel is a registered trademark of Intel Corporation.

Lotus Notes is a registered trademark of Lotus Development Corporation.

MS-DOS and Windows 95 are registered trademarks, and Windows NT is a trademark of Microsoft Corporation.

NetBIOS is a trademark of Micro Computer Systems, Inc.

NetWare and Novell are registered trademarks of Novell, Inc.

Proteon, ProNET, and TokenVIEW are registered trademarks of Proteon, Inc.

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

All other trademarks and registered trademarks are the property of their respective holders.

This manual was produced by DIGITAL Information Design.

Contents

Preface

1 Getting Started

2 Configuring and Monitoring AppleTalk Phase 1

3 Configuring and Monitoring AppleTalk Phase 2

4 Configuring and Monitoring ARP

5 Configuring and Monitoring BGP4

6 Configuring and Monitoring Bandwidth Reservation

7 Configuring and Monitoring DNA IV

8 Configuring and Monitoring DVMRP

9 Configuring and Monitoring IP

10 Configuring and Monitoring IPX

11 Configuring and Monitoring OSI/DNA V

12 Configuring and Monitoring OSPF

13 Configuring and Monitoring PIM

14 Configuring and Monitoring SNMP

15 Configuring and Monitoring RMON

16 Using the DIGITAL Trace Facility

A SNMP Objects

B Packet Sizes

C Comparison of Protocols

D DIGITAL MIB Support

Figures

Tables

Preface

Objectives

This *Routing Protocols User’s Guide* explains how to configure and monitor the protocol software shipped with your bridging router.

Audience

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

For further information on each of the protocols discussed in this guide, refer to the *Routing Protocols Reference Guide*.

Using This Guide

The following table helps you locate information in this guide:

If You Want Information About...	See Chapter or Appendix...
• Summary of Document Contents • Related Documentation • Document Set Structure • Documentation Conventions	Preface
• Accessing Protocol Configuration and Console Processes • Accessing the Protocol Configuration Process • Accessing the Protocol Console Process • Accessing the Digital Trace Facility	**1** Getting Started
• AppleTalk Phase 1 and AppleTalk Phase 2 • Accessing the AppleTalk Phase I Configuration Environment • Basic Configuration Procedures • AppleTalk Phase I Configuration and Console Commands	**2** Configuring and Monitoring AppleTalk Phase 1
• AppleTalk Phase I and AppleTalk Phase II • Accessing the AppleTalk Phase II Configuration Environment • Basic Configuration Procedures • AppleTalk Phase II Configuration and Console Commands	**3** Configuring and Monitoring AppleTalk Phase 2
• Accessing the ARP Configuration Environment • ARP Configuration and Console Commands	**4** Configuring and Monitoring ARP
• Border Gateway Protocol Review • How BGP Works • Setting Up BGP • Sample Policy Definitions • BGP Configuration and Console Commands	**5** Configuring and Monitoring BGP4

If You Want Information About...	See Chapter or Appendix...
• Displaying the Bandwidth Reservation Configuration Prompt • Displaying the Bandwidth Reservation Monitoring Prompt • Bandwidth Reservation Configuration and Console Commands	**6** Configuring and Monitoring Bandwidth Reservation
• Accessing the NCP Environment • Configuring DNA IV in an OSI/DNA V Environment • NCP Configuration and Console Commands	**7** Configuring and Monitoring DNA IV
• Accessing the DVMRP Configuration Environment • DVMRP Commands	**8** Configuring and Monitoring DVMRP
• Accessing the IP Configuration and Console Environments • Basic Configuration Procedures • The BOOTP Forwarding Process • IP Configuration and Console Commands	**9** Configuring and Monitoring IP
• Accessing the IPX Configuration Environment • IPX Configuration and Console Commands	**10** Configuring and Monitoring IPX
• Accessing the OSI Configuration Environment • Basic Configuration Procedure • Configuring OSI Over an Ethernet, Token Ring, or FDDI LAN • Configuring OSI Over X.25 or Frame Relay • Enabling Compression on DLM and DA Circuits • Configuring OSI Over a Serial Line • Configuring a DNA V Router for a DNA IV Environment • DNA IV and DNA V Algorithm Considerations • OSI Commands	**11** Configuring and Monitoring OSI/DNA V

If You Want Information About...	See Chapter or Appendix...
• Accessing the OSPF Configuration and Console Environments • Basic Configuration Procedures • OSPF Configuration and Console Commands	**12** Configuring and Monitoring OSPF
• Configuring for Protocol Independent Multicasting • Accessing the Interface Configuration Process • Accessing the PIM Console Environment • PIM Configuration and Console Commands	**13** Configuring and Monitoring PIM
• Accessing the SNMP Configuration Environment • SNMP Configuration and Console Commands	**14** Configuring and Monitoring SNMP
• Overview of Remote Monitoring Feature • Basic configuration procedures • Accessing RMON configuration and console environments • RMON Configuration and Console Commands	**15** Configuring and Monitoring RMON
• Overview of the DIGITAL Trace Facility • Accessing DTF • Router Tracepoints	**16** Using the DIGITAL Trace Facility
• SNMP Objects • Packet Sizes • Comparison of Protocols • DIGITAL MIB Support	**A** **B** **C** **D**

Using Related Documentation

DIGITAL Documents

This Document...	**Describes...**
RouteAbout Access EI Installation EK-DEXBR-IN	Installation and use of the RouteAbout Access EI router.
RouteAbout Access EW Installation EK-DEX2R-IN	Installation and use of the RouteAbout Access EW router.
RouteAbout Access TW Installation EK-DEWTR-IN	Installation and use of the RouteAbout Access TW router.
RouteAbout Central EI Installation EK-DEZBR-IN	Installation and use of the RouteAbout Central EI router.
RouteAbout Central EP Installation EK-DEZPR-IN	Installation and use of the RouteAbout Central EP router.
RouteAbout Central EW Installation EK-DEZ8R-IN	Installation and use of the RouteAbout Central EW router.
Bridging Configuration Guide AA-QL29E-TE	The configuration and monitoring procedures for bridging methods. Bridging features that enhance system performance.
clearVISN Router Configurator User’s Guide AA-R08YB-TE	The graphic user interface application which enables you to create and load a basic configuration for the bridging router.
DTF (DIGITAL Trace Facility) User Guide AA-R85DA-TE	How to install and use the DIGITAL Trace Facility, which enables you to trace packets within the protocol layers of the bridging router.
Event Logging System Messages Guide AA-QL2AE-TE	How events are logged, how to interpret Event Logging System (ELS) messages. Provides a description of each ELS message with a corresponding corrective action.

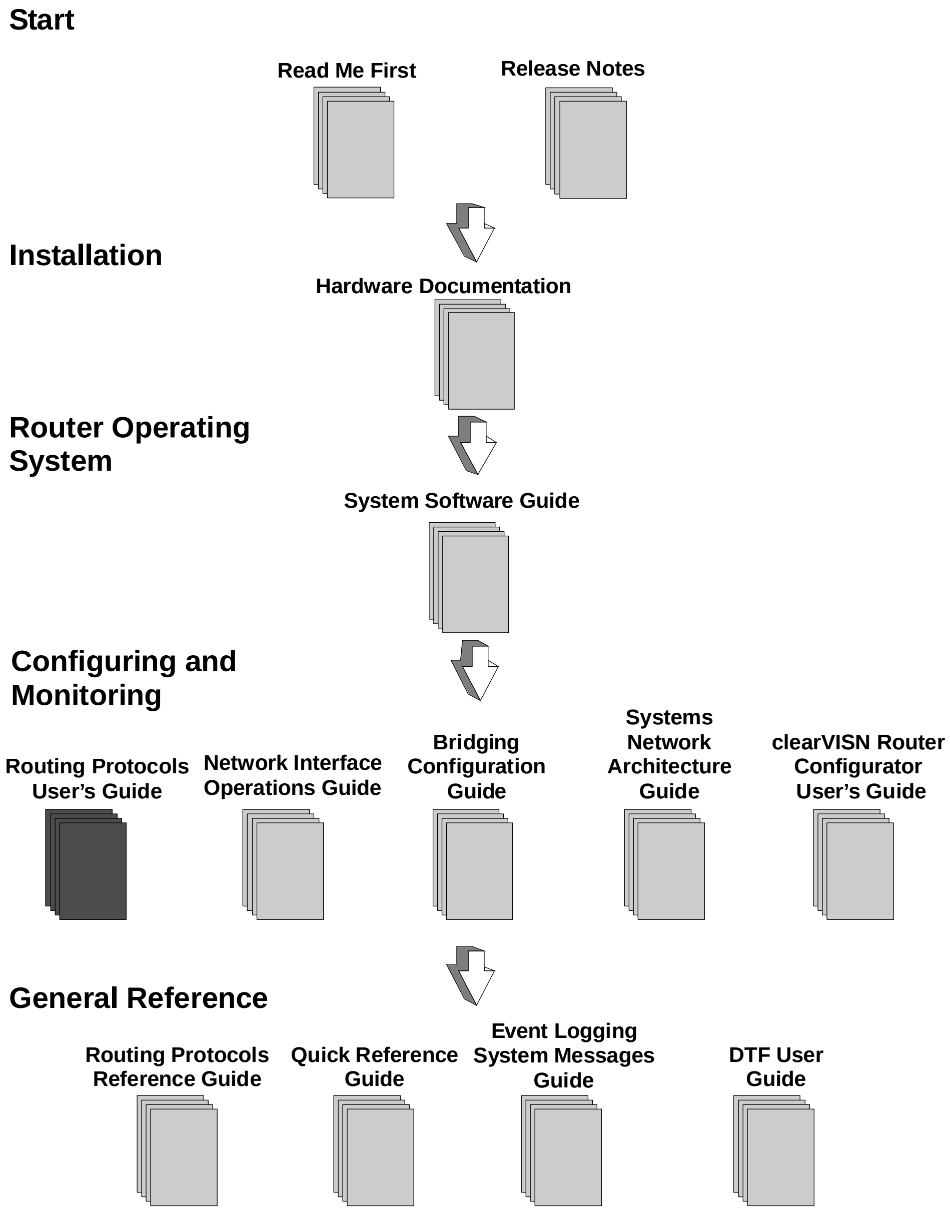
Network Interface Operations Guide AA-QL2BE-TE	Configuring and monitoring the network interfaces in the Distributed Routing Software bridging router.

This Document...	Describes...
Quick Reference Guide AA-R7QAA-TE	How to configure and monitor the main protocols, features and interfaces, and lists the associated commands.
Routing Protocols Reference Guide AA-QL2CE-TE	Reference information about the micro-operating system structure, and the protocols and interfaces supported by bridging routers.
Systems Network Architecture Guide AA-QU5SC-TE	SNA interfaces and protocols for the Distributed Routing Software System.
System Software Guide AA-QL2EE-TE	Installing, configuring, and operating the Distributed Routing Software system software.

Document Set Structure

Figure 1 shows the structure of the documentation set.

**Figure 1	Document Set Structure**



Start
Read Me First
Release Notes
Installation
Hardware Documentation
Router Operating System
System Software Guide
Configuring and Monitoring
Routing Protocols User's Guide
Network Interface Operations Guide
Bridging Configuration Guide
Systems Network Architecture Guide
clearVISN Router Configurator User's Guide
General Reference
Routing Protocols Reference Guide
Quick Reference Guide
Event Logging System Messages Guide
DTF User Guide

LKG-10590-97C

Conventions

The following conventions are used in this guide:

`Monospace type` Monospace type in examples indicates system output or user input.

`Boldface type` Boldface type in examples indicates user input. Boldface type is also used for file names and command names within text.

lowercase italics Lowercase italics in command syntax or examples indicate variables for which either the user or the system supplies a value.

[ ] Brackets enclose operands or symbols that are either optional or conditional. Specify the operand and value if you want the condition to apply. Do not type the brackets in the line of code.

`key` A key name in bold type indicates that you press the specified key.

`Ctrl/X` Indicates that you hold the Ctrl key while pressing the key specified by the *X*. The server displays the key combination as ^X.

underscore Characters underlined in a command listing represent the fewest number of characters you must enter to identify that command to the interpreter.

2-3 In the Index, page reference numbers in bold type indicate a reference to a command description.

Symbols

C M

The configuring and monitoring chapters contain a description of all commands you can use to configure and monitor the protocol, feature, or interface.

C means you use the command to configure the router. You access configuration commands after you enter **talk 6** at the * prompt. Configuration commands change the router’s nonvolatile database; a router restart is necessary to activate the change.

M means you use the command to monitor and dynamically configure the router. You access monitoring commands after you enter **talk 5** at the * prompt. Changes made in this mode take effect immediately, but are not made in the router’s nonvolatile database (and therefore not preserved after a router restart).

C M means you use the command both to configure and to monitor the router.

Note: **Talk 5** monitoring commands are also referred to as console commands in this guide. **Talk 6** configuration commands are sometimes referred to as config commands.

Commands

Figure 2 shows the components of a command description.

Figure 2 Command Components

Command Name

Description of commands.

Syntax: command-name
parameter 1 . . .
parameter 2 . . .

parameter 1 ***option***

Description of parameter and options.

Example:

```
command name  parameter
Prompt [Default value]?options
```

Command Name The name of the command followed by an overview description.

Syntax: The command followed by each parameter you can configure using that command. If an ellipsis follows a parameter, you need to enter additional information (*options*). When you enter a command, you can save time by typing only the underlined letters.

parameter Description of each parameter.

option Information (in italics) you must enter with the command and parameter.

Example: An example of how you enter that command and its parameter.

Entering Commands

Instead of being prompted for options, you can save time by entering the complete command on one line. For example, you can enter the **set framesize** command shown in Figure 3 as follows:

```
set framesize 2048
```

If you abbreviate the command using the underlined letters, you can enter

```
s f 2048
```

Figure 3 Set Framesize Command

Set

Configures frame size and local address.

Syntax: set
framesize . . .
parameter 2 . . .

framesize ***1024*** or ***2048*** or ***4096***

The size of the network-layer portion of frames transmitted and received on the interface.

Example:

```
set framesize
Framesize in bytes (1024/2048/4096) [1024]? 2048
```

Accepting the Current Setting

When the software prompts you for information, the current setting appears in brackets [ ]. To accept the information in the brackets, press **Return**. In this example, the current setting is 1024.

```
Framesize in bytes (1024/2048/4096) [1024]?
```

Correspondence

Documentation Comments

If you have comments or suggestions about his document, send them to the DIGITAL Network Products Business Organization.

Attn:	Documentation Project Manager
E-MAIL:	doc_quality@lkg.mts.dec.com

Online Services

To locate product-specific information, refer to the DIGITAL Network Products Home Page on the World Wide Web, located at the following addresses:

North America	http://www.networks.digital.com
Europe	http://www.networks.europe.digital.com
Asia Pacific	http://www.networks.digital.com.au

How to Order Additional Documentation

To order additional documentation, use the following information:

To Order:	**Contact:**
By Telephone	USA (except Alaska, New Hampshire, and Hawaii): 1-800-DIGITAL (1-800-344-4825) Alaska, New Hampshire, and Hawaii: 1-603-884-6660 Canada: 1-800-267-6215
Electronically (USA only)	Dial 1-800-DEC-DEMO (For assistance, call 1-800-DIGITAL)
By Mail (USA and Puerto Rico)	DIGITAL EQUIPMENT CORPORATION P.O. Box CS2008 Nashua, New Hampshire 03061 (Place prepaid orders from Puerto Rico with the local DIGITAL subsidiary: 809-754-7575)
By Mail (Canada)	DIGITAL EQUIPMENT of CANADA LTD. 940 Belfast Road Ottawa, Ontario, Canada K1G 4C2 Attn.: A&SG Business Manager
Internationally	DIGITAL EQUIPMENT CORPORATION Attn.: A&SG Business Manager c/o local DIGITAL subsidiary or approved distributor
Internally	U.S. Software Supply Business (SSB) DIGITAL EQUIPMENT CORPORATION 8 Cotton Road Nashua, New Hampshire 03063

1

Getting Started

This chapter explains how to access the processes required to configure and monitor the protocol software shipped with your bridging router and to trace the activity on the ports. If you want your bridging router to initialize automatically (without a console terminal), refer to Chapter 1 of the *System Software Guide* for information about the EasyStart process.

For further information about the protocols discussed in this guide, refer to the *Routing Protocols Reference Guide*.

1.1 Accessing Protocol Configuration and Console Processes

All protocols described in this guide have commands that are executed by doing one of the following:

- Accessing the protocol configuration process to initially configure and enable the protocol as well as perform later configuration changes
- Accessing the protocol console process to monitor information about each protocol or make temporary configuration changes

The procedures for accessing these processes is basically the same for all protocols. The next sections describe these procedures.

1.1.1 Accessing the Protocol Configuration Process

Each protocol configuration process is accessed through the router’s CONFIG process. CONFIG is the second-level process of the router user interface that lets you communicate with third-level processes. Protocol processes are examples of third-level processes.

The CONFIG command interface is made up of levels that are called modes. Protocol configuration command interfaces are modes of the CONFIG interface. Each protocol configuration interface has its own prompt. For example, the prompt for the TCP/IP protocol command interface is `IP config>`.

1.1 Accessing Protocol Configuration and Console Processes

In general, the procedure for accessing the protocol configuration processes is as follows:

- Enter the CONFIG command process from OPCON and obtain the CONFIG prompt.
- Enter the desired protocol configuration process (with its own prompt) from the CONFIG prompt using the **protocol** command.

The following sections describe these procedures in more detail.

1.1.1.1 Entering the CONFIG Process

To enter the CONFIG command process from OPCON and obtain the CONFIG prompt:

1. At the OPCON prompt, enter the **status** command to find the pid (process ID) of CONFIG.

```
* status

     Pid    Name    Status  TTY    Comments
      1    COpCon    IOW    TTY0
      2    Monitr    DET     --
      3    Tasker    IDL     --
      4    MOSDDT    DET     --
      5    CGWCon    IOW     --
      6    Config    IOW     --
      7    ROpCon    IOW    TTY1    janb
      8    ROpCon    RDY    TTY2
```

2. Enter the OPCON **talk** command and the pid for CONFIG. The pid for CONFIG is 6.

```
* talk 6
```

The console displays the CONFIG prompt (`Config>`). If the prompt does not appear, press Return again.

1.1.1.2 Entering the Desired Protocol Configuration Process

To enter the desired protocol configuration process from the CONFIG prompt:

1. At the CONFIG prompt, enter the **list configuration** command to see the numbers and names of the protocols available for the router. For example:

```
Config>list configuration
Hostname: [none]
Maximum packet size: [autoconfigured]
Maximum number of global buffers: [autoconfigured]
Number of Restarts before a Reload/Dump: 64
Logging disposition: detached
Console inactivity timer (minutes): 0
Physical console login: disabled
Modem control: disabled
Contact person for this node: [none]
Location of this node: [none]

Configurable Protocols:
Num Name  Protocol
0   IP    DOD-IP
3   ARP   Address Resolution
4   DN    DNA Phase IV
6   VIN   Banyan Vines
7   IPX   NetWare IPX
8   OSI   ISO CLNP/ESIS/ISIS
9   DVM   Distance Vector Multicast Routing Protocol
10  BGP   Border Gateway Protocol
11  SNMP  Simple Network Management Protocol
12  OSPF  Open SPF-Based Routing Protocol
14  APL   AppleTalk
22  AP2   AppleTalk Phase 2
23  ASRT  Adaptive Source Routing Transparent Enhanced Bridge
24  HST   TCP/IP Host Services
27  PIM   Protocol Independent Multicast

Configurable Features:
Num Name  Feature
0   WRS   WAN Restoral
1   BRS   Bandwidth Reservation
2   MCF   MAC Filtering
4   X25S  X25 Switching
```

2. From the CONFIG prompt, enter the **protocol** command with the number or *short name* of the protocol you want to configure. You can obtain the protocol number and short name from the **list configuration** command display. The following example shows the command for accessing the IP protocol configuration process by the protocol short name:

   ```
   Config> protocol IP
   ```

   The protocol configuration prompt then displays on the console. This example shows the IP protocol prompt `IP config>`:

   ```
   IP config>
   ```

   You can achieve the same result by entering the **protocol** command followed by the protocol “number.” In the following example, the command was entered to access the IP protocol configuration process by the protocol number:

   ```
   Config> protocol 0
   ```

   The protocol configuration prompt then displays on the console.

   ```
   IP Config>
   ```

You can now begin entering that protocol’s configuration commands. See the corresponding protocol section of this guide for more information about specific protocol configuration commands.

In summary, the **protocol** command lets you enter the configuration process for the protocol software installed in your router. The **protocol** command enters a protocol’s command process. After entering the **protocol** command, the prompt of the specified protocol appears. From the prompt, you can enter commands specific to that protocol.

1.1.1.3 Exiting the Protocol Configuration Process

After configuring or changing the protocol configuration process, exit the protocol configuration process:

1. Return to the CONFIG process by entering the protocol **exit** command. For example:

   ```
   IP Config> exit
   ```

2. Return to the OPCON process by entering the OPCON intercept character (Ctrl/P). For example:

```
Config> ^p
*
```

1.1.1.4 Restarting the Router

Changes that you make to the protocol parameters through CONFIG do not take effect until you restart the router or reload the router software. The only exception is for certain DNA IV **NCP set** commands.

Note: The changes you make through CONFIG are retained in a configuration database in nonvolatile memory. They are retained during powerdowns and are recalled when you restart the router.

To restart the router, enter the OPCON **restart** command. For example:

```
* restart

Are you sure you want to restart the router? (Yes or No): yes
```

1.1.2 Accessing the Protocol Console Process

To view information about the protocol or to change parameters at the console, you must access and use the protocol console process. Protocol console command interfaces are modes of the GWCON interface. Within the GWCON mode, each protocol console interface has its own prompt. For example, the prompt for the TCP/IP protocol is `IP>`.

In general, the procedure for accessing the protocol console processes is as follows:

- Enter the GWCON command process from OPCON and obtain the GWCON prompt.
- Enter the desired protocol console process from the GWCON prompt using the **protocol** command.

The next sections describe these procedures in more detail.

1.1 Accessing Protocol Configuration and Console Processes

1.1.2.1 Entering the GWCON Command Process

The general process for entering the GWCON process from OPCON and obtaining the GWCON prompt is as follows:

1. Enter the **status** command to find the pid (process ID) of GWCON. For example:

```
* status
Pid    Name    Status  TTY    Comments
 1    COpCon    IOW    TTY0
 2    Monitr    DET     --
 3    Tasker    IDL     --
 4    MOSDDT    DET     --
 5    CGWCon    IOW     --
 6    Config    IOW     --
 7    ROpCon    IOW    TTY1    janb
 8    ROpCon    RDY    TTY2
*
```

2. At the OPCON prompt, enter the OPCON **talk** command and the pid number for GWCON. For example:

```
* talk 5
```

The GWCON prompt (+) then displays on the console. If the prompt does not appear, press Return again.

1.1.2.2 Entering a Protocol Console Process

To enter a protocol console process from the GWCON prompt:

1. At the GWCON prompt, enter the **configuration** command to see the protocols and networks configured for the bridge. For example:

```
+ configuration

Portable MC68040 C Gateway [not configured] S/N 452
V15.1[]
Boot ROM version 0.4
Watchdog timer enabled
Auto-boot switch enabled
Manufacturing rest enabled
Manufacturing test disabled
Console baud rate: 0

Num Name  Protocol
0   IP    DOD-IP
3   ARP   Address Resolution
7   IPX   NetWare IPX
11  SNMP  Simple Network Management Protocol
14  APL   AppleTalk
```

```
22  AP2   AppleTalk Phase 2
23  ASRT  Adaptive Source Routing Transparent Enhanced Bridge

Num Name  Feature
2   MCF   MAC Filtering


7 Networks:
Net Interface  MAC/Data-Link         Hardware                  State
0   FDDI/0     IEEE 802.2/FDDI       WGE200 FDDI                Up
1   Eth/0      Ethernet/IEEE 802.3                              Up
2   Eth/1      Ethernet/IEEE 802.3                              Up
3   Eth/2      Ethernet/IEEE 802.3                              Up
4   Eth/3      Ethernet/IEEE 802.3                              Down
5   Eth/4      Ethernet/IEEE 802.3                              Down
6   Eth/5      Ethernet/IEEE 802.3                              Down
```

2. Enter the GWCON **protocol** command with the protocol number or short name of the desired protocol displayed in the configuration information. In the following example, the command was entered for accessing the IP protocol console process:

   ```
   + protocol 0
   ```

   OR

   ```
   + protocol IP
   ```

   The protocol console prompt then displays on the console. This example shows the IP protocol console prompt (`IP>`):

   ```
   IP>
   ```

You can now begin entering that protocol's console commands. See the corresponding protocol section of this guide for more information about specific protocol console commands.

1.1.2.3 Exiting the Protocol Console Process

To exit the protocol console process and return to the OPCON process:

1. Return to the GWCON process by entering the protocol **exit** command. For example:

   ```
   IP> exit
   ```

2. Return to the OPCON process by entering the OPCON intercept character (Ctrl/P). For example:

```
+ ^p
*
```

1.1.2.4 Protocol Names and Numbers

Table 1–1 lists the numbers that you enter along with the **protocol** command when accessing a specific protocol configuration or console process.

Table 1–1 Protocol Numbers and Names

Protocol Number	Protocol Short Name	Accesses the Following Protocol Process
0	IP	Internet Protocol
3	ARP	Address Resolution Protocol
4	DN	DNA - a subset of Network Control Program
7	IPX	Novell NetWare Internetwork Packet Exchange
8	OSI	ISO Open Systems Interconnect Connectionless Network Layer Protocol / ESIS / ISIS
9	DVM	Distance Vector Multicast Routing Protocol
10	BGP	Border Gateway Protocol
11	SNMP	Simple Network Management Protocol
12	OSPF	Open Shortest Path First
14	APL	AppleTalk Phase 1
20	SDLC	SDLC Relay
22	AP2	AppleTalk Phase 2
23	ASRT	Adaptive Source Routing Transparent Bridge
24	HST	TCP/IP Host Services
27	PIM	Protocol Independent Multicast

1.2 Accessing the DIGITAL Trace Facility (DTF)

The DIGITAL Trace Facility (DTF) is a host-based facility that allows the tracing of packets as they traverse through the protocol layers within the router. DTF supports DIGITAL UNIX Alpha, ULTRIX, Linux OpenVMS, and Windows NT host platforms.

DTF is included with the clearVISN Router Configurator software. It is in the ***install-directory*\tools\supported\dtf\** subdirectory.

The latest versions of the DTF documentation and installation kits for each host platform are available over the Internet, and can be downloaded from the following World Wide Web locations:

- **http://www.networks.digital.com**
- **http://www.networks.europe.digital.com** (Europe)
- **http://www.networks.digital.com.au** (Asia Pacific)

Use the search feature to find the DTF Installation Kit. For more information about using DTF with your bridging router, refer to Chapter 16.

2

Configuring and Monitoring AppleTalk Phase 1

This chapter describes the AppleTalk configuration and console commands.

For more information about AppleTalk, refer to the *Routing Protocols Reference Guide*.

2.1 AppleTalk Phase 1 and AppleTalk Phase 2

Your router provides separate packet forwarders to support both AppleTalk Phase 1 (which is described in this chapter and is also referred to as *AppleTalk* or *APL*) and its enhancement, AppleTalk Phase 2 (AP2). The difference between Phase 1 and Phase 2 is that Phase 2 removes the Phase 1 restriction of a maximum number of 254 concurrently active AppleTalk devices on one network. You can now assign more than one network number to a single AppleTalk network. The size of the range of network numbers assigned to a network determines the maximum number of concurrently active AppleTalk devices that the network can support (253 devices per network number).

To allow Phase 1 hosts to transparently communicate with Phase 2 hosts, you must enter the AppleTalk Phase 2 configuration process on the router running AP2 and then enable the AppleTalk Phase 1/Phase 2 translation process through that router's AP2 **enable translation** configuration command. For more information about the **enable translation** command and AppleTalk Phase 2, refer to Chapter 3.

In addition to providing the Phase 1/Phase 2 translation process function, this router now acts as both a Phase 1 and Phase 2 router on whatever interfaces these protocols are configured. Routing information is passed between Phase 1 and Phase 2 networks through the translation process resulting in a (logically) single internet.

2.2 Accessing the AppleTalk Phase 1 Configuration Environment

For information about accessing the AppleTalk Phase 1 configuration environment, see Chapter 1.

2.3 Basic Configuration Procedures

This section outlines the initial steps you must perform to configure and run the AppleTalk Phase 1 protocol. The command sections of this chapter describe how you can make further configuration changes. Before any configuration changes can take effect, you must restart the router.

A complete configuration example appears near the end of this section.

2.3.1 Enabling Router Parameters

When you configure a router to forward AppleTalk Phase 1 packets, you must enable certain parameters regardless of the number or type of interfaces in the router. If you have multiple routers transferring AppleTalk Phase 1 packets, specify these parameters for each router.

- **Globally Enable AppleTalk Phase 1** – To begin, you must globally enable the AppleTalk Phase 1 software using the AppleTalk Phase 1 **enable apl** configuration command. If the router displays an error in this step, there is no AppleTalk Phase 1 software present in your router. You should contact your DIGITAL Customer Services representative.
- **Enable Specific Interfaces** – You must then enable the specific interfaces over which you want AppleTalk Phase 1 to send and receive packets. Use the **enable interface** *interface number* command to do this.

2.3.2 Setting Network Parameters

For each network and interface that sends and receives AppleTalk Phase 1 packets, you must specify certain parameters. These parameters are described below.

Note: After specifying these parameters, use the AppleTalk Phase 1 **list** configuration command to view the results of the configuration.

- **Set the Network Numbers** – AppleTalk Phase 1 network numbers are 16-bit integers, specified in decimal, in the range of 1 to 65535. (Network 0 is illegal.) Each physical network must have a unique network number. Although you may have multiple routers with interfaces on one network, you need only configure the network number on one router. The configured router, called the *seed router*, dynamically sends the network number to the connected routers through the RTMP routing protocol. If you do not configure the network number on a router, the router can learn it from other seed routers on the network.

To connect the interface numbers to the network numbers, use the **set net-number** *interface-number APL-network-number* command while in the AppleTalk Phase 1 configuration process.

- **Set the Node Addresses** – AppleTalk Phase 1 node numbers are 1-byte integers, specified in decimal, in the range of 1 to 254. You can configure a node address for each interface on a network that sends and receives AppleTalk Phase 1 packets. This is the node address used for the Link Access Protocol, such as the Ethernet Link Access Protocol. If you do not configure a node address, it is selected automatically. However, for management purposes, it is desirable to use a fixed node address for a router.

  To set the node addresses, use the AppleTalk Phase 1 **set node-number interface-number** *number* configuration command.

- **Set a Zone Name** – You can configure a zone name for each network in the internetwork. In addition, you can configure the zone name for a given network in any router connected to that network. Only the seed router needs to contain the zone name information for a connected network. Attached routers dynamically acquire the zone name from adjacent routers using the ZIP protocol. Apple recommends that for a given network, you choose the same seed router for the network number and the zone name.

  To set the zone name for each network number, use the AppleTalk Phase 1 **set zone interface-number** *node-name* configuration command.

- **Set the Routing Table Size** – The size of the routing table is configurable. The routing table limits the size by setting the amount of available memory. If there are more than 194 active networks, the router sends the Routing Table Maintenance Protocol (RTMP) data as multiple packets.

  To set the routing table size, use the AppleTalk Phase 1 **set nnets** *number* configuration command.

2.3 Basic Configuration Procedures

- **Set the Network Packet Header Size** – You can specify whether the router uses short or long header DDP packets on each network in certain situations. For example, you can configure the router to generate short headers for packets destined for a host on a directly connected network. If not configured, the router defaults to long DDP headers, as recommended by Apple.

  To set the network header packet size, use the AppleTalk Phase 1 **set ddp-header** *name interface-number* configuration command.

2.3.2.1 Sample Configuration Procedure for AppleTalk Phase 1

The following procedure demonstrates the configuration of AppleTalk Phase 1 on a RouteAbout Access EI bridging router.

In this procedure, each parameter that has a default value is given that value. For parameters that have no default (for example, addresses), the procedure uses an arbitrary value.

Note: You should check all values carefully and change them as necessary for your individual implementation.

The procedure uses the following steps to configure the AppleTalk Phase 1 protocol on router interfaces:

1. Talk to the router.

   ```
   * t 6
   ```

2. Configure the AppleTalk Phase 1 protocol.

   ```
   Config> protocol APL
   ```

3. Enable APL and the interfaces on which APL will run.

   ```
   APL Protocol User Configuration
   APL Config> enable APL
   APL Config> enable interface 1
   ```

4. Set the network number.

   ```
   APL Config> set NET-Number
   Which interface [0]? 1
   Network number (1-65535, or 0 to delete) []? 3233
   APL Config>
   ```

5. Set the node number.

```
APL Config> set NODE-NUMBER
Which interface [0]? 1
Node Number (1-254, or 0 to delete) []? 233
APL Config>
```

6. Set the zone name.

```
APL Config> set ZONE
Which interface [0]? 1
Zone name []? zone_name
APL Config>
```

7. Set the routing table size.

```
APL Config> set NNETS 32
APL Config>
```

8. Set the network packet header.

```
APL Config> set ddp-header long
Which interface [0]? 1
APL Config>
```

Note: You must restart the router to put these values in effect.

2.3.3 Configuration Restrictions

There are no configuration restrictions on Phase 1 networks. All network numbers, however, must be either Phase 1 only or Phase 2 only. A physical network can contain both Phase 1 and Phase 2 hosts as long as the router is configured with different network numbers.

With Phase 2 networks, when Phase1/Phase 2 translation is enabled, a network can belong to only one zone. The reason for this restriction is that the Phase 1 Zone Information Protocol (ZIP) maps a network number into a single zone. Without the single zone restriction, a Phase 1 ZIP query cannot be properly processed.

2.4 AppleTalk Phase 1 Configuration and Console Commands

This section summarizes and explains the AppleTalk Phase 1 configuration and console commands.

The AppleTalk Phase 1 configuration commands allow you to specify network parameters for router interfaces that transmit AppleTalk Phase 1 packets. The information you specify with the configuration commands becomes active when you restart the router.

The AppleTalk Phase 1 console commands allow you to view the parameters and statistics of the interfaces and networks that transmit AppleTalk Phase 1 packets. Console commands display configuration values for the physical, frame, and packet levels. You can optionally view the values for all three protocol levels at once.

Enter the AppleTalk Phase 1 configuration commands at the `APL config>` prompt.

Enter the AppleTalk Phase 1 console commands at the `APL>` prompt.

Table 2–1 AppleTalk Phase 1 Configuration and Console Commands Summary

Command	Tasks	Function
? (Help)	Configuring and Monitoring	Lists the AppleTalk Phase 1 configuration and console commands or lists the options associated with specific commands.
Counters	Monitoring	Displays the number of packet overflows.
Disable	Configuring	Disables checksum or takedown, disables a specified interface, or globally disables AppleTalk Phase 1.
Dump	Monitoring	Displays the contents of the routing table.
Enable	Configuring	Enables checksum or takedown, enables a specified interface, or globally enables AppleTalk Phase 1.
Interface	Monitoring	Lists the addresses of all interfaces in the router.
List	Configuring	Displays the current AppleTalk Phase 1 configuration.
Set	Configuring	Sets the DDP header, network number, node number, size of the routing table, and zone parameters.
Exit	Configuring and Monitoring	Exits the AppleTalk Phase 1 configuration or console process and returns to the CONFIG environment.

2.4 AppleTalk Phase 1 Configuration and Console Commands

? (Help) C M

Lists the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example:

```
?
DISABLE
ENABLE
EXIT
LIST
SET
```

Example:

```
set ?
DDP-HEADER
NET-NUMBER
NODE-NUMBER
NNETS
ZONE
```

Counters M

Displays the number of packet overflows on each network that sends and receives AppleTalk Phase 1 packets. This command displays the number of times the AppleTalk Phase 1 forwarder input queue was full when packets were received from the specified network.

Syntax: counters

Example:

```
counters
APL input packet overflows
    Net   Count
   Eth/0  4
   TKR/0  0
```

2.4 AppleTalk Phase 1 Configuration and Console Commands

Disable C

Disables the checksum generation or takedown, disables a specified interface, or globally disables the AppleTalk Phase 1 protocol.

```
Syntax:     disable
                        apl
                        checksum
                        interface . . .
                        takedown
```

apl

Disables the AppleTalk Phase 1 packet forwarder as a whole.

Example:

```
disable apl
```

checksum

Specifies that the router does not compute the checksum in the packets it generates. This is the default. The router verifies checksums on all packets that it forwards.

Example:

```
disable checksum
```

interface *interface#*

Disables all AppleTalk Phase 1 functions on the specified interface.

Example:

```
disable interface 3
```

takedown

Prevents ZIP takedown and bringup packets from affecting the routers network numbers and zone names. This is the default for security reasons.

Example:

```
disable takedown
```

2.4 AppleTalk Phase 1 Configuration and Console Commands

Dump M

Obtains routing table information about the interfaces on the router that forwards AppleTalk Phase 1 packets.

Syntax: dump

Example:

```
dump
Dest Net    Cost    State    Next hop    Source   Zone
    3        0      Dir        3/0         APL    ″Blue″
   72        0      Dir        3/0         AP2    ″Green″
   13        1      Good       3/13        AP2    ″Fuchsia″
   63        2      Good       3/13        APL     NIL
   42        3      Suspct     3/13        APL    ″Orange″
5 entries used out of 32
```

Dest Net Specifies the destination network number in decimal.

Cost Specifies the number of router hops to this destination network.

State Specifies the state of the entry in the routing table. It includes the following:

- **Dir** – Indicates that the router is directly connected to the routing table, the interface is enabled and up, and the network number is known.
- **Good** – Indicates that an RTMP packet containing a good tuple for this network was heard in the last 20 seconds.
- **Suspct** – Indicates that no RTMP tuple was received for this network in the last 20 seconds.
- **Bad** – Indicates that no RTMP tuple was received for this network in the last 40 seconds. RTMP packets with tuples listing this network as unreachable are sent for 20 seconds, then the network is deleted from the RTMP routing table. (For more information, refer to the RTMP chapter in *Inside AppleTalk* by Gursharan S. Sidhu, First Edition.)

Next hop Specifies the next hop for packets going to networks that are not directly connected. For directly connected networks, the node number is 0.

Source Specifies the originating router type for that routing table entry. APL indicates an AppleTalk Phase 1 router. AP2 indicates AppleTalk Phase 2.

Note: If the Phase 1/Phase 2 translation gateway process was not enabled, you do not see the source column.

Zone Specifies the human-understandable name for that network. The zone name is enclosed in double quotes in case there are embedded spaces or nonprinting characters. If the zone name contains characters beyond the 7-bit ASCII character set (they are 8-bit), the zone name that is displayed depends on the characteristics of your console terminal. If there is no zone name known for this network, the name NIL (without quotes) is displayed.

Note: At the bottom of the display is the number of entries used and the total available. If all the entries are used, it is likely that the routing table is not large enough. Use the AppleTalk Phase 1 **set nnets** configuration command to increase the size.

Enable C

Enables the checksum generation or takedown, enables a specified interface, or globally enables the AppleTalk Phase 1 protocol.

Syntax: enable

apl
checksum
interface . . .
takedown

apl

Allows the router to send AppleTalk Phase 1 packets over all of the interfaces.

Example:

```
enable apl
```

checksum

Specifies that the router does not compute the checksum in the packets it generates. The router verifies checksums on all packets that it forwards.

Example:

```
enable checksum
```

interface ***interface#***

Allows the router to send and receive AppleTalk Phase 1 packets over the specified interface.

Example:

```
enable interface 3
```

takedown

Allows any node on the AppleTalk Phase 1 internetwork to use ZIP takedown and bringup packets to change network numbers and zone names for the router.

Example:

```
enable takedown
```

Interface M

Displays the addresses of all the interfaces in the router on which AppleTalk Phase 1 is enabled. If the interface is present in the router but it is disabled, this command shows that status.

Syntax: interface

Example:

```
interface
Interface    Addresses
  Eth/0      3/29
  TKR/0       APL not enabled
```

2.4 AppleTalk Phase 1 Configuration and Console Commands

List C

Displays the current AppleTalk Phase 1 configuration. In the example, the router is a seed router on networks 13 and 22. It chooses a node number dynamically on net 22 and on interface 2. It also learns the network number and zone name from a seed router on interface 2.

Syntax: list

Example:

```
list
APL globally  enabled
Checksumming  disabled
Takedown      disabled
Table size    32


List of configured interfaces:

Interface           DFLT DDP hdr    APL address        Zone
   0                long              13/4           ”Jupiter”
   1                short             22/0           ”Neptune”
   2                long               0/0            ””
```

APL globally Indicates whether AppleTalk Phase 1 is globally enabled or disabled.

Checksumming Indicates whether checksum is enabled or disabled.

Takedown Indicates whether takedown is enabled or disabled.

Table size Indicates the size of the table.

List of configured interfaces Lists each interface number and its associated DDP header, APL address, and zone name. The zone name is enclosed in double quotes in case there are embedded spaces or nonprinting characters.

2.4 AppleTalk Phase 1 Configuration and Console Commands

Set C

Defines specific AppleTalk Phase 1 parameters, including the DDP header, network number, node number, size of the routing table, and zone.

Syntax: set

ddp-header long . . .
ddp-header short . . .
net-number . . .
node-number . . .
nnets . . .
zone . . .

ddp-header long ***interface#***

Specifies long DDP headers for packets sent on that interface number. This is the default and is recommended by Apple.

Example:

```
set ddp-header long 2
```

ddp-header short ***interface#***

Specifies short DDP headers for packets sent on that interface number. Use this only for compatibility with software that does not support long DDP headers.

Example:

```
set ddp-header short 2
```

net-number ***interface# AppleTalk Phase 1-net#***

Assigns an AppleTalk Phase 1 network number to the associated directly connected network. This router is a seed for the network number. If it is not set, it is learned from a seed router. Setting the network number to 0 restores it to the unseeded state.

Example:

```
set net-number 1 33
```

node-number ***interface# node#***

Specifies the number of the interface. This is optional. The default is auto configure. Setting the node number to 0 restores auto configuration.

Example:

```
set node-number 0 12
```

2.4 AppleTalk Phase 1 Configuration and Console Commands

nnets ***size***

Specifies the size of the AppleTalk Phase 1 routing table. This reflects the number of networks in the internet that are running AppleTalk Phase 1. There is no maximum size limit; however, the router can run out of memory.

Example:

```
set nnets 4
```

zone ***interface#*** ***name***

Specifies the zone name to be seeded on this network. Setting the zone name to an empty string restores auto configuration.

Example:

```
set zone 1 jupiter
```

Exit **C M**

Return to the previous prompt level.

Syntax: exit

Example:

```
exit
```

3

Configuring and Monitoring AppleTalk Phase 2

This chapter describes the AppleTalk Phase 2 (AP2) configuration and console commands.

For more information about AppleTalk Phase 2, refer to the *Routing Protocols Reference Guide*.

3.1 AppleTalk Phase 1 and AppleTalk Phase 2

Your router provides separate packet forwarders to support both AppleTalk Phase 1 (APL) and its enhancement, AppleTalk Phase 2 (AP2). The difference between Phase 1 and Phase 2 is that Phase 2 removes the Phase 1 restriction of a maximum number of 254 concurrently active AppleTalk devices on one network. You can now assign more than one network number to a single AppleTalk network. The size of the range of network numbers assigned to a network determines the maximum number of concurrently active AppleTalk devices that can be supported on that network (253 devices per network number).

To allow Phase 1 hosts to transparently communicate with Phase 2 hosts, you must enter the AppleTalk Phase 2 configuration process on the router running AP2 and enable the AppleTalk Phase 1/2 translation process through that router’s AP2 **enable translation** configuration command. For more information about the **enable translation** command, see the command section of this chapter.

In addition to providing the Phase 1/Phase 2 translation process function, this router now acts as both a Phase 1 and Phase 2 router on whatever interfaces these protocols are configured. Routing information is passed between Phase 1 and Phase 2 networks through the translation process resulting in a (logically) single internet.

3.2 Accessing the AppleTalk Phase 2 Configuration Environment

For information about accessing the AppleTalk Phase 2 configuration environment, see Chapter 1.

3.3 Basic Configuration Procedures

This section outlines the initial steps required to get the AppleTalk Phase 2 protocol up and running. Information about how to make further configuration changes is covered in the command sections of this chapter. For the new configuration changes to take effect, the router must be restarted.

A complete configuration example appears near the end of this section.

3.3.1 Enabling Router Parameters

When you configure a router to forward AppleTalk Phase 2 packets, you must enable certain parameters regardless of the number or type of interfaces in the router. If you have multiple routers transferring AppleTalk Phase 2 packets, specify these parameters for each router.

- **Globally Enable AppleTalk Phase 2** – To begin, you must globally enable the AppleTalk Phase 2 software using the AppleTalk Phase 2 configuration **enable ap2** command. If the router displays an error in this step, there is no AppleTalk Phase 2 software present in your load. If this is the case, contact your Digital Customer Services representative.
- **Enable Specific Interfaces** – You must then enable the specific interfaces over which AppleTalk Phase 2 is to send the packets. Use the **enable interface** *interface-number* command to do this.
- **Enable Checksumming** – You can then determine whether the router computes DDP checksums of the packets it originates. Checksum software does not work correctly in some AppleTalk Phase 2 implementations, so you may not want to originate packets with checksums for compatibility with these implementations. Any packet forwarded with a checksum has its checksum verified.
- **Enable Phase 1/Phase 2 Translation (Optional)** – Use the **enable translation c**ommand to allow Phase 2 hosts to transparently communicate with Phase 1 hosts.

3.3.2 Setting Network Parameters

You must also specify certain parameters for each network and interface that sends and receives AppleTalk Phase 2 packets. After you specify the parameters, use the AppleTalk Phase 2 **list** configuration command to view the results of the configuration.

- **Set the Network Range for Seed Routers** – Coordinating network ranges and zone lists for all routers on a network is simplified by having specific routers designated as seed routers. Seed routers are configured with the network range and zone list while all other routers are given null values. Null values indicate that the router queries the network for values from the seed routers. There are usually several seed routers on a network in case one of them fails. Also, a router can be a seed router for some or all of its network interfaces. Use the **set net-range** command to assign the network range in seed routers.
- **Set the Starting Node Number** – Use the **set node-number** command to assign the starting node number for the router. The router AARPs for this node, but if it is already in use, a new **node-number** is chosen.
- **Add a Zone Name** – You can add one or more zone names for each network in the internetwork. You can add a zone name for a given network in any router connected to that network; however, only the seed router needs to contain the zone name information for a connected network. Attached routers dynamically acquire the zone names from adjacent routers using the ZIP protocol. Apple recommends that for a given network, you choose the same seed router for the network range and the zone names. The zone names cannot be configured for a network unless the network range is also configured. To add a zone name to an interface, use the AppleTalk Phase 2 configuration **add zone** *name* command.

3.3.3 Setting Up Zone Filters

Zone filtering lets you filter zones in each direction on each interface. To filter incoming packets, set up an input filter. To filter outgoing packets, set up an output filter. The interface does not readvertise filtered zone information in the direction that you define. To set up a zone filter:

1. Add zone filters to an interface.

   To add an input zone filter, use the **add zfilter in** command. To add an output zone filter, use the **add zfilter out** command. The software prompts you for the interface number and the name of the zone that you want to filter.

   ```
   AP2 config> add zfilter in
   Interface # [0]? 1
   Zone name []? Admin
   ```

2. Enable the zone filters that you added.

   To enable an input zone filter, enter **enable zfilter in**. To enable an output zone filter, enter **enable zfilter out**. The software prompts you for the interface number and for whether or not the filter is inclusive or exclusive. Inclusive filters forward only the zone information in a filter. Exclusive filters block only the zone information in a filter.

   ```
   AP2 config> enable zfilter in
   Interface # [0]? 1
   INCLUSIVE/EXCLUSIVE [INCLUSIVE]? exclusive
   ```

3.3.4 Setting Up Network Filters

Network filters are similar to zone filters, except they let you filter an entire network. To set up a network filter, follow these steps:

1. Add a network filter. Use the **add nfilter in** command to add an input network filter to an interface. Use the **add nfilter out** command to add an output network filter to an interface. For example:

   ```
   AP2 config> add nfilter out
   Interface # [0]? 0
   First Network range number (decimal)[0]? 11
   Last Network range number (decimal) [0]? 15
   ```

   The network range you enter here must match the range that you assigned to that network.

2. Enable the network filter that you added and make it either inclusive or exclusive. Inclusive filters forward only network information in a filter. Exclusive filters block only network information in a filter, and they allow all other network information to be forwarded.

```
AP2 config> enable nfilter in
Interface # [0]? 0
INCLUSIVE/EXCLUSIVE [INCLUSIVE]? exclusive
```

3.3.4.1 Sample Configuration Procedure for AppleTalk Phase 2

The following procedure demonstrates the configuration of AppleTalk Phase 2 on a RouteAbout Access EI bridging router.

In this procedure, each parameter that has a default value is given that value. For parameters that have no default (for example, addresses), the procedure uses an arbitrary value.

Note: You should check all values carefully and change them as necessary for your individual implementation.

The following steps summarize the procedure you can use the to configure the AppleTalk Phase 2 protocol on router interfaces:

1. Talk to the router and reach AppleTalk configuration.
2. Globally enable AppleTalk Phase 2.
3. Enable specific interfaces.
4. Enable checksum checking.
5. Enable Phase 1/Phase 2 translation (optional).
6. Set the network parameters:
   a. Set the network range for seed routers.
   b. Set the starting node number.
   c. Add a zone name.
7. Set up zone filters:
   a. Add a zone filter to an interface.
   b. Enable the added zone filters.
8. Set up network filters:
   a. Add a network filter.

3.3 Basic Configuration Procedures

b. Enable the network filter.

To configure AppleTalk Phase 2 on the router, perform the following steps. You may need to change the values or parameters given here according to your network setup.

Note: You must restart the router to put these values in effect.

1. Talk to the router and reach the AppleTalk configuration.

   ```
   * t 6
   Config> protocol AP2
   AP2 Protocol user configuration
   AP2 Config>
   ```

2. Globally enable AppleTalk Phase 2.

   ```
   AP2 Config> enable ap2
   AP2 Config>
   ```

3. Enable specific interfaces.

   ```
   AP2 Config> enable interface 1
   AP2 Config>
   ```

4. Enable checksum checking.

   ```
   AP2 Config> enable checksum
   AP2 Config>
   ```

5. Enable Phase 1/Phase 2 translation (optional).

   ```
   AP2 Config> enable translation
   AP2 Config
   ```

6. Set the network parameters.

   a. Set the network range for seed routers.

      ```
      AP2 Config> set net-range
      Interface # [0]? 1
      First network range number (1-65279, or 0 to delete)[]? 1
      Last Network range number (1-65279) []? 65279
      AP2 Config>
      ```

b. Set the starting node number.

```
AP2 Config> set node-number
Interface # [0]? 1
Node Number (1-253, or 0 to delete) []? 1
AP2 Config
```

c. Add a zone name.

```
AP2 Config> add zone
Interface # [0]? 1
zone name []? zone_name_1
AP2 Config>
```

7. Set up zone filters.

a. Add a zone filter to an interface.

```
AP2 Config> add zfilter in
Interface # [0]? 1
zone name []? Admin
```

b. Enable the added zone filters.

```
AP2 Config> enable zfilter in
Interface # [0]? 1
INCLUSIVE/EXCLUSIVE [INCLUSIVE]?  EXCLUSIVE
```

8. Set up network filters.

a. Add a network filter.

```
AP2 Config> add nfilter out
Interface # [0]? 1
First network range number (decimal) [0]? 11
Last network range number (decimal) [0]? 15
AP2 Config>
```

b. Enable the network filter.

```
AP2 Config> enable nfilter in
Interface # [0]? 1
INCLUSIVE/EXCLUSIVE [INCLUSIVE]? EXCLUSIVE
AP2 Config>
```

3.3.5 Disabling Checksumming

As the default, the router computes DDP checksums of packets it originates. Checksum software does not work correctly in some AppleTalk implementations, so you may not want to originate packets with checksums. In this case, disable checksum using the **disable checksum** command.

3.3.6 Configuration Restrictions

There are no configuration restrictions on Phase 2 networks. All network numbers must be either Phase 1 only or Phase 2 only. A physical network can contain both Phase 1 and Phase 2 hosts as long as the router is configured with different network numbers.

With Phase 1 networks, a network can belong to only one zone. The reason for this restriction is that the Phase 1 Zone Information Protocol (ZIP) maps a network number into a single zone. Without the single zone restriction, a Phase 1 ZIP query cannot be properly processed.

3.4 AppleTalk Phase 2 Configuration and Console Commands

This section explains the AppleTalk Phase 2 configuration and console commands.

The AppleTalk Phase 2 configuration commands allow you to specify network parameters for router interfaces that transmit AppleTalk Phase 2 packets. The information you specify with the configuration commands becomes active when you restart the router.

The AppleTalk Phase 2 console commands allow you to view the parameters and statistics of the interfaces and networks that transmit AppleTalk Phase 2 packets.

Enter the AppleTalk Phase 2 configuration commands at the `AP2 config>` prompt.

Enter the AppleTalk Phase 2 console commands at the `AP2>` prompt.

3.4 AppleTalk Phase 2 Configuration and Console Commands

Table 3–1 AppleTalk Phase 2 Configuration and Console Commands Summary

Command	Tasks	Function
? (Help)	Configuring and Monitoring	Lists the AppleTalk Phase 2 configuration commands or lists the options associated with specific commands.
Add	Configuring	Adds the interface default zone name to the interface zone list, an IP tunnel address, a network filter or a zone filter, or a zone name to the interface zone list.
Counters	Monitoring	Displays the input overflow count of AP2 packets for each interface.
Delete	Configuring	Deletes an interface definition, an IP tunnel address, a network filter or a zone filter, or a zone name from the interface zone list.
Disable	Configuring	Disables AppleTalk Phase 2 globally, split horizon routing, checksum generation, a specified interface, an IP tunnel, Phase 1/2 translation, or a network filter or zone filter.
Dump	Monitoring	Displays the current state of the routing table for all networks in the internet and their associated zone names.
Enable	Configuring	Enables AppleTalk Phase 2 globally, split horizon routing, checksum generation, a specified interface, an IP tunnel, Phase 1/2 translation, or a network filter or a zone filter.
Interface	Monitoring	Displays the current addresses of the interfaces.
List	Configuring	Displays the current AppleTalk Phase 2 configuration.
Set	Configuring	Sets the net range or node number for an interface.
Exit	Configuring and Monitoring	Exits the AppleTalk Phase 2 configuration process and returns to the CONFIG environment.

3.4 AppleTalk Phase 2 Configuration and Console Commands

? (Help) C M

Lists the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example:

```
?
ADD
DELETE
DISABLE
ENABLE
LIST
SET
EXIT
```

Example:

```
set ?
CACHE-SIZE
NET-RANGE
NODE-NUMBER
```

Add C

Adds a zone name to the interface zone list as the default for the interface, an IP tunnel address, a network filter or zone filter, or a zone name to the interface zone list.

Syntax: add

defaultzone . . .
ip-tunnel-address . . .
nfilter . . .
zfilter . . .
zone . . .

defaultzone ***interface# zonename***

Adds the zone name to the interface zone list as the default for the interface. If two defaults are defined, the last one overrides the first. If no default is defined, the first zone name is the default.

Example:

```
add defaultzone 2 newyork
```

3.4 AppleTalk Phase 2 Configuration and Console Commands

ip-tunnel-address *address* DEC

Adds an IP tunnel endpoint. The value *address* is a 32-bit IP address in the form *n.n.n.n*, where each *n* is a decimal integer representing one octet of the address. DEC is the encapsulation type and is currently the only one supported.

Example:

```
add ip-tunnel-address 1.2.3.4 DEC
```

nfilter <in or out> *interface# first-range# last-range#*

Adds a network filter to let you filter an entire network. The arguments **in** and **out** let you define the direction of flow on which the filter applies. The values of *first-range#* and *last-range#* represent the network numbers that defined the network range when the network was created.

Example:

```
add nfilter in
Interface # [0]? 1
First Network range number (decimal)[0]? 11
Last Network range number (decimal) [0]? 15
```

zfilter <in or out> *interface# zonename*

Adds a zone filter to let you filter zones on input or output. The arguments **in** and **out** let you define the direction of flow on which the filter applies. The values of *interface#* and *zonename* represent the interface and the name of the zone that you are filtering.

Example:

```
add zfilter in
interface # [0]? 1
Zone name []? Admin
```

zone *interface# zonename*

Adds the zone name to the interface zone list. If no default zone name is defined, the first zone name for the interface is the default. If there are no zone names for an interface, the default is “*”.

Example:

```
add zone 2 newyork
```

3.4 AppleTalk Phase 2 Configuration and Console Commands

Counters M

Displays the number of input packet overflows on each network that sends and receives AppleTalk Phase 2 packets. This command displays the number of times the AppleTalk Phase 2 forwarder input queue was full when packets were received from the specified interface.

Syntax: counters

Example:

```
counters
AP2 input packet overflows
     Net   Count
    Eth/0  4
    TKR/0  0
```

Delete C

Deletes an interface definition, an IP tunnel address, a network filter or a zone filter, or a zone name from the interface zone list.

Syntax: delete

```
interface . . .
ip-tunnel-address . . .
nfilter . . .
zfilter . . .
zone  . . .
```

interface *interface#*

Deletes all AP2 information for the specified interface. Sometimes this is the only way to delete zone names that have nonprinting characters.

Example:

```
delete interface 2
```

ip-tunnel-address *address*

Deletes the IP tunnel endpoint whose address is *address*. The *address* is a 32-bit IP address in the form *n.n.n.n*, where each *n* is a decimal integer representing one octet of the address.

Example:

```
delete ip-tunnel-address 1.2.3.4
```

nfilter <in or out> ***interface# first-network# last-network#***

Deletes a network filter from the input or output of the interface. Enter the same network range you set using the **add nfilter in** command.

Example:

```
delete nfilter in
Interface # [0]? 0
First Network range number (decimal) [0]? 1
Last Network range number (decimal) [0]? 12
```

zfilter <in or out> ***interface# zonename***

Deletes a zone name filter from the input or the output of the interface.

Example:

```
delete zfilter out
Interface # [0]? 1
Zone name []? Marketing
```

zone ***interface# zonename***

Deletes the specified zone name from the interface zone list.

Example:

```
delete zone 2 newyork
```

Disable C

Disable AppleTalk Phase 2 globally, checksumming, a specified interface, an IP tunnel, AppleTalk Phase 1/2 gateway functionality, network filters, or zone filters.

Syntax: disable

ap2
checksum
interface . . .
ip-tunnel
nfilter . . .
split-horizon-routing
translation
zfilter . . .

3.4 AppleTalk Phase 2 Configuration and Console Commands

ap2

Disables the AppleTalk Phase 2 packet forwarder for all interfaces.

Example:

```
disable ap2
```

checksum

Specifies that the router does not compute the checksum in packets that it generates. This is the default. The router verifies checksums on all packets that it forwards.

Example:

```
disable checksum
```

interface ***interface#***

Disables all AP2 functions on the specified network interface. The network continues to remain available for all other protocols.

Example:

```
disable interface
Interface #  [0]? 2
```

ip-tunnel

Globally disables the IP tunnel.

Example:

```
disable IP-tunnel
```

nfilter <in or out> ***interface#***

Disables, but does not delete, the input or output network filters on this interface.

Example:

```
disable nfilter in
```

split-horizon-routing ***interface #***

Disables split-horizon routing on the specified interface. Disabling split-horizon-routing means that ALL routing information will be propagated on this interface. This is usually only necessary on a Frame Relay interface on a hub router that has multiple DLCI connections in a partially-meshed Frame Relay network.

Split horizon routing is **enabled** by default.

Example:

```
disable split-horizon-routing
Interface # [0]? 2
```

3.4 AppleTalk Phase 2 Configuration and Console Commands

translation

Disables the translation process that allows Phase 2 hosts to transparently communicate with Phase 1 hosts.

Example:

```
disable translation
```

zfilter <in or out> *interface#*

Disables, but does not delete, the input or output zone filters on this interface.

Example:

```
disable zfilter out
Interface # [0]? 1
```

Dump M

Obtains routing table information about the interfaces on the router that forwards AppleTalk Phase 2 packets.

Syntax: dump

Example:

```
dump
Dest Net    Cost    State    Next hop          ZoneList
  10-19       0      Dir     0/0        “Ethertalk”, “Sales”
  40-49       1     Good     10/13      “Marketing”,“CustomerSer”,
  20-29       2     Sspct    10/13      “Fuchsia”, “Backbone”,
                                        “Engineering”, “MKTING”
3 entries
```

Dest Net Specifies the destination network number range, in decimal.

Cost Specifies the number of router hops to this destination network.

State Specifies the state of the entry in the routing table. It includes the following:

- **Dir** – Indicates that the router is directly connected to the destination network, the interface is enabled and up, and the network number is known.
- **Good** – Indicates that an RTMP packet containing a good tuple for this network was heard in the last 20 seconds.

- **Suspct** – Indicates that no RTMP tuple was received for this network in the last 20 seconds.
- **Bad** – Indicates that no RTMP tuple was received for this network in the last 40 seconds. RTMP packets with tuples listing this network as unreachable are sent for 20 seconds, then the network is deleted from the RTMP routing table.

Next hop Specifies the next hop for packets going to networks that are not directly connected. For directly connected networks, this field is blank.

ZoneList Specifies the zone list for that network. The zone names are enclosed in quotation marks in case there are embedded spaces or nonprinting characters. If a zone name contains characters beyond the 7-bit ASCII character set (they are 8-bit), the zone name that is displayed depends on the characteristics of your console terminal.

Note: If the Phase 1/Phase 2 translation gateway process was enabled, you are also shown another column (between next hop and zone). This column is labelled **source** and lists the originating network type for that table entry.

Enable C

Enable the AppleTalk Phase 2 protocol globally, the checksum function, a specified interface, an IP tunnel, the AppleTalk Phase 1/2 gateway functionality, a network filter, or a zone filter.

Syntax:

```
enable
          ap2
          checksum
          interface . . .
          ip-tunnel
          nfilter . . .
          split-horizon-routing
          translation
          zfilter . . .
```

3.4 AppleTalk Phase 2 Configuration and Console Commands

ap2

Enables the AppleTalk Phase 2 packet forwarder over all of the interfaces.

Example:

```
enable ap2
```

checksum

Specifies that the router computes the checksum in packets that it generates. The router checksums all AP2 packets that it forwards.

Example:

```
enable checksum
```

interface *interface#*

Enables the router to send and receive AppleTalk Phase 2 packets over the specified interface.

Example:

```
enable interface
Interface #  [0]? 1
```

ip-tunnel

Globally enables the IP tunnel.

Example:

```
enable ip-tunnel
```

nfilter <in or out> *interface#*

Enables network input or output filters and controls how the filter is applied to the interface. Inclusive forwards matches. Exclusive drops matches.

Example:

```
enable nfilter in
Interface # [0]? 1
INCLUSIVE/EXCLUSIVE [INCLUSIVE]? inclusive
```

split-horizon-routing *interface #*

Enables split-horizon routing on the specified interface. Routing information learned over this interface will not be advertised back on this interface. Split horizon routing is enabled by default.

Example:

```
enable split-horizon-routing
```

3.4 AppleTalk Phase 2 Configuration and Console Commands

translation

Enables the translation process that allows Phase 2 hosts to transparently communicate with Phase 1 hosts. Besides providing the translation function, this router now acts as both a Phase 1 and Phase 2 router on whatever interfaces these protocols are configured. Routing information passes between Phase 1 and Phase 2 networks through the translation process, resulting in a (logically) single internet.

Example:

```
enable translation
```

zfilter <in or out> *interface#*

Enables and controls how the zone input or output filter is applied to the interface. Inclusive forwards matches. Exclusive drops matches.

Example:

```
enable zfilter out
Interface # [0]? 0
INCLUSIVE/EXCLUSIVE [INCLUSIVE]? inclusive
```

Interface M

Displays the addresses of all the interfaces in the router on which AppleTalk Phase 2 is enabled. If the interface is present in the router but is disabled, this command shows that status.

Syntax: interface

Example:

```
interface
Interface    Addresses
  SL/0       0/1 on net 1000-1000  default zone “SerialLine”
  Eth/0      10/52 on net 10-19    default zone  “Sales”
  SL/1       0/0 in startup range
```

3.4 AppleTalk Phase 2 Configuration and Console Commands

List C

Displays the current AppleTalk Phase 2 configuration. In the example, the router is a seed router on networks 13 and 22. It chooses a node number dynamically on net 22, and on interface 2. It also learns the network number and zone name from a seed router on interface 2.

Syntax: list

Example:

```
list
AP2 globally        enabled
AP2 IP tunnel       enabled
Checksumming        disabled
Translation Gateway enabled

List of configured interfaces:
  Interface      netrange / node   Zones
       0       1000-1000  / 1    ”SerialLine”(Def)
 Input ZFilters disabled
 Input NFilters disabled
 Output ZFilters disabled
 Output NFilters disabled
 Split-horizon-routing enabled

       1         10-19    / 52   ”Sales”(Def),”EtherTalk”
 Input ZFilters disabled
 Input NFilters disabled
 Output ZFilters disabled
 Output NFilters disabled
 Split-horizon-routing enabled

  IP Tunnel Endpoint  Encapsulation
     1.2.3.4          DEC
```

AP2 globally Indicates whether AppleTalk Phase 2 is globally enabled or disabled.

AP2 IP tunnel Indicates whether the AppleTalk Phase 2 IP tunnel is globally enabled or disabled.

Checksumming Indicates whether checksum is enabled or disabled.

Translation Gateway Indicates whether the AppleTalk Phase 1/Phase 2 translation is globally enabled or disabled.

3.4 AppleTalk Phase 2 Configuration and Console Commands

List of configured interfaces
Lists each interface number and its associated net range, node number, zones (including the default zone), and input and output network filters and zone filters. The zone name is enclosed in double quotes in case there are embedded spaces or nonprinting characters.

Input/Output Zfilters
Indicates whether input or output Zfilters are enabled or disabled for each interface.

Input/Output Nfilters
Indicates whether input or output Nfilters are enabled or disabled for each interface.

IP Tunnel Endpoint
Indicates the IP address of the endpoint.

Encapsulation
Indicates the type of IP encapsulation. DEC is the only encapsulation currently supported.

Set C

Defines the network range or node number for an interface.

Syntax: set

net-range . . .
node . . .

net-range *interface# start# end#*

Assigns the network range in seed routers using the following:

- **interface#** – Designates the router interface to operate on.
- **start#** – Assigns the lowest number of the network range. Legal values are 1 to 65279 (FEFF hexadecimal).
- **end#** – Sets the highest number of the network range. Legal values are *start#* to 65279.

A single numbered network has the same start and end values. A start value of zero deletes the network range for the interface and turns the seeded interface into an unseeded interface. *Start#* and *end#* are inclusive in the network range.

Example:

```
set net-range 2 43 45
```

node ***interface# node#***

Assigns the starting node number for the router interface. The router will AARP for this node, but if it is already in use, a new node number is chosen. The following explains each argument that is entered after this command:

- **interface#** – Designates the router interface to operate on.
- **node#** – Designates the first attempted node number. Legal values are 1 to 253. A *node#* value of zero deletes the node number for the interface and forces the router to choose one at random.

Example:

```
set node 2 2
```

Exit C M

Returns to the previous prompt level.

Syntax: exit

Example:

```
exit
```

4

Configuring and Monitoring ARP

This chapter describes how to configure and monitor the Address Resolution Protocol (ARP) and how to use the ARP configuration and console commands.

For more information about ARP, refer to the *Routing Protocols Reference Guide*.

4.1 Accessing the ARP Configuration Environment

You can access the ARP configuration commands by typing **protocol arp** at the `Config>` prompt:

```
Config> protocol arp
```

You can access ARP console commands by entering **protocol arp** at the + prompt. For example:

```
+ protocol arp
```

For more information about accessing the ARP configuration environment, see Chapter 1.

4.2 ARP Configuration and Console Commands

This section explains the ARP configuration and console commands. Table 4–1 lists the ARP configuration and console commands.

4.2 ARP Configuration and Console Commands

Table 4–1 ARP Configuration and Console Commands Summary

Command	Tasks	Function
? (Help)	Configuring and Monitoring	Lists the ARP configuration commands or monitoring commands, or lists the options associated with specific commands.
Add Entry	Configuring	Adds a MAC address translation entry.
Clear	Monitoring	Clears the cache for a specified interface.
Change Entry	Configuring	Changes a MAC address translation entry.
Delete Entry	Configuring	Deletes a MAC address translation entry.
Disable Auto-refresh	Configuring	Disables ARP auto-refresh.
Dump	Monitoring	Displays the cache for a specified interface.
Enable Auto-refresh	Configuring	Enables ARP auto-refresh.
Hardware	Monitoring	Lists each ARP-configured network.
List	Configuring	Lists ARP configuration data in SRAM.
Protocol	Monitoring	Lists each ARP-configured protocol.
Set	Configuring	Sets the usage and refreshes timeout values.
Statistics	Monitoring	Displays ARP information.
Exit	Configuring and Monitoring	Exits the ARP configuration or monitoring process.

? (Help) C M

Lists the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example:

```
?
LIST
ADD
CHANGE
DELETE
DISABLE
ENABLE
SET
EXIT
```

Example:

```
list ?
ALL
ENTRY
CONFIG
```

Add Entry C

Adds a MAC address translation entry. All packets with the specified IP address will be sent over the specified interface to the given MAC address.

Syntax: add entry *ifc# prot-addr MAC-addr*

Example:

```
add entry
Interface Number [0]?
IP Address [0.0.0.0]?
Mac Address []?
```

Change Entry C

Changes a MAC address translation entry. The hardware address parameter (*MAC-addr*) is the address of the node being changed.

Syntax: change entry *ifc# prot-type prot-addr MAC-addr*

Example:

```
change entry
Interface Number [0]?
Protocol [IP]?
IP Address [0.0.0.0]?
Mac Address []?
```

Clear M

Flushes the ARP cache for a given network interface. The **clear** command can be used to force the deletion of bad transactions.

To clear a particular interface, enter the interface or network number as part of the command. To obtain the interface number, use the `Config>` **list devices** command.

Syntax: clear *interface#*

Example:

```
clear 1
```

4.2 ARP Configuration and Console Commands

Delete Entry C

Deletes a MAC address translation entry.

Syntax: delete entry *ifc# prot-type prot-addr*

Example:

```
delete entry
Interface Number [0]?
Protocol [IP]?
IP Address [0.0.0.0]?
```

Disable Auto-Refresh C

Disables the auto-refresh function. The auto-refresh function is the router’s capability to send another ARP request based on the entry in the translation cache before the refresh timer expires. The request is sent directly to the hardware address in the current translation instead of a broadcast. If auto-refresh is disabled, no additional ARP request is made, and the refresh timer is allowed to expire.

Syntax: disable auto-refresh

Example:

```
disable auto-refresh
```

Dump M

Displays the ARP cache for a given network/protocol combination. To display the ARP cache for a particular interface, enter the interface or network number as part of the command. To obtain the interface number, use the `Config>` **list devices** command.

If there is more than one protocol on that network, the protocol number must also be given. This causes the console to display the hardware address-to-protocol mappings stored in that database. If ARP is in use by only one protocol on the specified interface, then the protocol number is optional. To obtain the protocol number, use the `Config>` **protocol** command.

Syntax: dump *interface# protocol#*

Example:

```
dump 2
Hardware Address    IP Address      Refresh
02-07-01-00-00-01   192.9.1.2         5
```

4.2 ARP Configuration and Console Commands

Enable Auto-Refresh C

Enables the auto-refresh function. The auto-refresh function is the router’s capability to send another ARP request based on the entry in the translation cache before the refresh timer expires. The request is sent directly to the hardware address in the current translation instead of a broadcast. If auto-refresh is enabled, an additional ARP request is made in this manner before the refresh timer is allowed to expire.

Syntax: enable auto-refresh

Example:

```
enable auto-refresh
```

Hardware M

Displays the networks registered with ARP. The **hardware** command lists each ARP-registered network, and displays each network’s hardware address space (Hardware AS) and local hardware address. Hardware addresses are displayed according to hardware type (decimal for Token Ring, hexadecimal for Ethernet).

Syntax: hardware

Example:

```
hardware
Network       Hardware AS    Hardware Address

2 Eth/0       1              02-07-01-00-00-01
```

List C

Displays the contents of the router’s ARP configuration as stored in SRAM. The **list** command displays the current timeout settings for the refresh and usage timer.

Syntax: list

all
config
entry

all

Lists the ARP configuration followed by all of the ARP entries.

4.2 ARP Configuration and Console Commands

Example:

```
list all
ARP configuration:

Refresh Timeout: 5 minutes
Auto Refresh: disabled

Mac address translation configuration

IF #     Prot #  Protocol -> Mac Address
 6        0      9.9.3.2 -> 08002B1B0098
 0        0      9.9.4.1 -> 000000000045
```

config

Lists the configuration for the different ARP timers.

Example:

```
list config
ARP configuration:

Refresh Timeout: 5 minutes
Auto refresh: disabled
```

entry

Lists the ARP entries in nonvolatile memory.

Example:

```
list entry
Mac address translation configuration

IF #     Prot #  Protocol -> Mac address
 6        0      9.9.3.2 -> 08002B1B0098
 0        0      9.9.4.1 -> 000000000045
```

4.2 ARP Configuration and Console Commands

Protocol M

Displays (by network) the protocols with addresses registered with ARP. This command displays the network, protocol name, protocol number, protocol address space (in hexadecimal), and local protocol addresses.

Syntax: protocol

Example:

```
protocol
Network  Protocol (num)   AS    Protocol Address(es)
2 Eth/0  IP        (00)   800  192.9.1.1 18.124.0.11
```

Set C

Sets the ARP refresh timer.

Syntax: set refresh-timer

refresh-timer

Changes the timeout value for the refresh timer. To change the timeout value for the refresh timer, enter the timeout value in minutes. A setting of zero (0) turns off (disables) the refresh timer.

Example:

```
set refresh-timer 3
```

Statistics M

Displays a variety of statistics about the operation of the ARP module.

Syntax: statistics

Example:

```
statistics
ARP input packet overflows
   Net     Count
   Eth /0    0
   ARPA/0    0


ARP cache meters
Net Prot Max Cur Cnt Alloc Rfrsh:Tot Fail TMOs:Rfrsh TMOs:Use
  0   4    1   1   1    1         0    0          0        0
  1   0    2   2  12   12         0    0          0        0
  2   4    1   1   1    1         0    0          0        0
```

ARP input packet overflows Displays counters that represent the number of ARP packets discarded on input because the ARP layer was too busy. The counts shown are per network interface.

ARP cache meters Consists of a variety of meters on the operation of the ARP cache. The counts shown are all per protocol, per interface.

Net Displays the interface numbers.

Prot Displays the protocol numbers.

Max Displays the all-time maximum length hash chain.

Cur Displays the current maximum length hash chain.

Cnt Displays the count of entries currently active.

Alloc Displays the count of entries created.

Rfrsh:Tot Displays the number of refresh requests sent for this network interface and protocol.

Fail Displays the number of auto-refresh attempt failures due to unavailability of internal resources. This count is not related to whether or not an entry was refreshed.

TMOs:Rfrsh Displays the count of entries deleted due to a timeout of the refresh timer.

TMOs:Use Displays the count of entries deleted due to a timeout of the usage timer.

Exit C M

Returns to the previous prompt level.

Syntax: exit

Example:

```
exit
```

5

Configuring and Monitoring BGP4

This chapter describes how to configure the Border Gateway Protocol (BGP) using the BGP configuration commands and how to monitor BGP using the console commands.

5.1 Border Gateway Protocol Overview

BGP is an exterior gateway routing protocol used to exchange network reachability information among autonomous systems (ASs). An AS is essentially a collection of routers and endnodes that operate under a single administrative organization. Within each AS, routers and endnodes share routing information using an interior gateway protocol. The interior gateway protocol may be either RIP, OSPF, or Integrated IS-IS.

BGP was introduced in the Internet in the late 1980s to facilitate the loop-free exchange of routing information between autonomous systems (ASs). Based on Classless Inter-Domain Routing (CIDR*)*, BGP has since evolved to support the *aggregation* and *reduction* of routing information*.*

In essence, CIDR is a strategy designed to address the following problems:

- Exhaustion of Class B address space
- Routing table growth

CIDR eliminates the concept of address classes, and provides a method for summarizing *n* different routes into single routes. This significantly reduces the amount of routing information that BGP routers must store and exchange.

Note: Digital Equipment Corporation supports only the latest version of BGP, BGP4, which is defined in RFC 1654. All references to BGP in this chapter and on the interface of DIGITAL routers are to BGP4, and do not apply to previous versions of BGP.

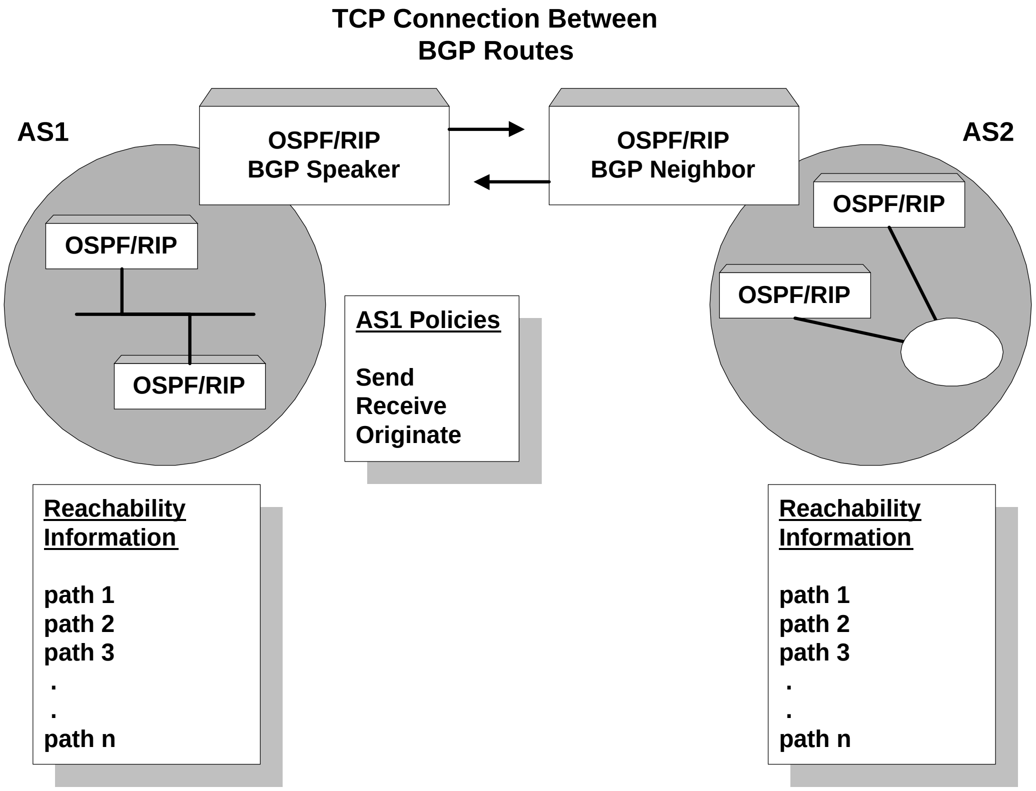
5.2 How BGP Works

BGP is not a routing protocol, but a reachability protocol. In essence, BGP routers selectively collect and advertise reachability information to and from BGP neighbors in their own and other autonomous systems (ASs). Reachability information consists of the sequences of AS numbers that form the paths to particular BGP speakers, and the list of IP addresses that can be reached via each advertised path. An *AS* is a administrative group of networks and routers that share reachability information using one or more Interior Gateway Protocols (IGPs), such as RIP or OSPF.

Routers that run BGP are called *BGP speakers.* These routers function as servers with respect to their BGP neighbors (their clients). Each BGP router opens a passive TCP connection on port 179, and listens for incoming connections from neighbors at this well-known address. The router also opens active TCP connections to enabled BGP neighbors. This TCP connection enables BGP routers to share and update reachability information with neighbors in the same or other ASs. Connections between BGP speakers in the same AS are called *internal BGP (IBGP) connections*, while connections between BGP speakers in different ASs are *external BGP (EBGP) connections.* A single AS may have one or many BGP connections to outside ASs.

Figure 5–1 shows two ASs. The BGP speaker in AS1 attempts to establish a TCP connection with its neighbor in AS2. After this connection is established, the routers can share reachability information.

Figure 5–1 BGP Connections Between Two Autonomous Systems



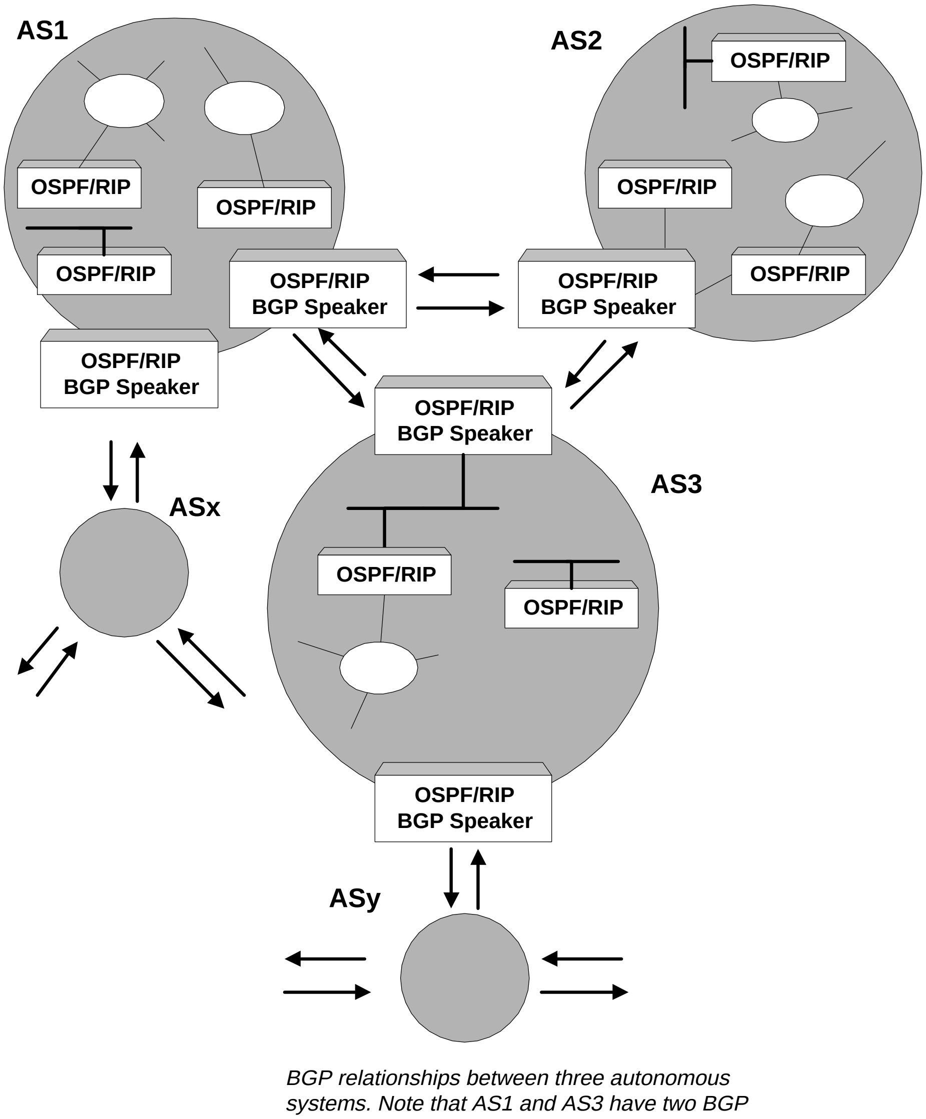
Once the BGP speaker in AS1 establishes a TCP connection with its BGP neighbor in AS2, the two routes can selectively exchange reachability information. The information each router sends or accepts is determined by policies defined for each router.

LKG-10601-97V

While the ASs shown in Figure 5–1 have only one BGP router, each may have multiple connections to other ASs. As an example of this, Figure 5–2 shows three interconnected ASs. AS1 has three BGP connections to outside ASs: one to AS2, one to AS3, and one to ASx. Similarly, AS3 has connections to AS1, AS2 and to ASy.

5.2 How BGP Works

Figure 5–2 BGP Connections Between Three Autonomous Systems



BGP relationships between three autonomous systems. Note that AS1 and AS3 have two BGP speakers.

LKG-10602-97V

5.2.1 Originate, Send, and Receive Policies

Decisions about which reachability information to advertise (send) and which to accept (receive) are made on the basis of explicitly defined policy statements. DIGITAL BGP implementation supports three types of policy statements:

- Originate Policies
- Send Policies
- Receive Policies

Once a TCP connection is established, the BGP speaker shown in Figure 5–2 can send its entire routing table to its BGP neighbor in AS2. However, for security or other reasons, it may not be desirable to send reachability information on each network to AS2. Similarly, it may not be desirable for AS2 to receive reachability information on each network in AS1.

Note: Before you can send or receive information, you must establish policies.

5.2.2 BGP Messages

BGP routers use four kinds of messages to communicate with their neighbors: OPEN, KEEP ALIVE, UPDATE, and NOTIFICATION messages.

5.2.2.1 OPEN

Open messages are the first transmitted when a link to a BGP neighbor comes up and establishes a connection.

5.2.2.2 KEEP ALIVE

Keep alive messages are used by BGP routers to inform one another that a particular connection is alive and working.

5.2.2.3 UPDATE

Update messages contain the interior routing table information. BGP speakers send update messages only when there is a change in their routing tables.

5.2.2.4 NOTIFICATION

Nofication messages are sent whenever a BGP speaker detects a condition that forces it to terminate an existing connection. These messages are advertised before the connection is transmitted.

5.3 Setting Up BGP

Setting up BGP involves four basic steps:

1. Accessing the BGP Configuration Prompt

   To access the BGP configuraton prompt, you talk to the router and specify the protocol.

2. Enabling BGP

   Enabling BGP requires you to specify the BGP router’s unique AS number. AS numbers are assigned by Stanford Research Institute Network Information Center.

3. Defining BGP Neighbors

   BGP neighbors are BGP routers with which a BGP speaker establishes a TCP connection. Once neighbors are defined, connections to them are established by default.

4. Defining Policies

   The policies you establish determine which routes are imported and exported by the BGP speaker.

   You can set up policies for different purposes. See Section 5.4 for more information.

The sections that follow explain each of these steps in detail.

5.3.1 Accessing the BGP Configuration Prompt

To access the BGP configuration prompt (`BGP Config>`), enter the **talk 6** command to the ***** prompt.

Example:

```
* t 6
Config> protocol BGP

Border Gateway Protocol User Configuration
BGP Config>
```

5.3.2 Enabling BGP

You enable BGP using the **enable BGP speaker** command as shown:

Example:

```
BGP Config> enable BGP speaker
AS [0]? 167
TCP segment size [1024]? 1024
BGP Config>
```

The AS number must be greater than zero, but less than or equal to 65535.

The TCP segment size must be greater than zero, but less than or equal to 65535. The default value is 1024. This number represents the maximum segment size BGP uses for passive TCP connections.

5.3.3 Defining BGP Neighbors

After enabling a BGP speaker, you must define its neighbors. BGP neighbors can be internal or external. Internal neighbors exist in the same AS, and do not need to have a direct connection to one another. External neighbors exist in different ASs. They must have a direct connection to one another.

To define internal or external BGP neighbors, use the **add neighbor** command. You must specify the IP address of the neighbor, and assign an AS number to the neighbor as shown below. Internal neighbors must have the same AS number as the BGP speaker.

Example:

```
BGP Config>add neighbor 192.0.190.178
AS [0]? 178
Init timer [12]? 30
Connect timer [120]? 120
Hold timer [90]? 30
TCP segment size [1024]? 512
BGP Config>
```

Adding a BGP neighbor automatically enables it, causing the BGP speaker to send out a connection request to the neighbor.

5.3.4 Adding Policies

DIGITAL BGP implementation supports three policy commands:

- **Originate Policy** – Enables you to select the interior gateway protocol (IGP) networks to export. These policies apply to routes to which the BGP speaker is directly connected; that is, routes that are local to the BGP speaker.

- **Receive Policy** – Enables you to select the route information to import from BGP peers.
- **Send Policy** – Enables you to select the route information to export to peers. Note that exportable route information can include information collected from neighboring ASs, as well as the routes that originate in the IGP.

5.4 Sample Policy Definitions

This section provides a set of examples of some specific policies you can set up for a BGP speaker.

You define all policies using the BGP **add** command.

5.4.1 Originate Policy Examples

5.4.1.1 Include All Routes for Advertisement

This example includes all routes in the BGP speaker’s IGP routing table for advertisement. In this sense, you can view this command as the default originate policy statement for BGP.

Notice that the command specifies a range of addresses, rather than a single (exact) address.

Example:

```
BGP Config> add originate-policy inclusive
Network Prefix [0.0.0.0]? 0.0.0.0
Network Mask [0.0.0.0]? 0.0.0.0
Address Match (Exact/Range) [Exact]? range
Tag [0]? 0
BGP Config>
```

5.4.1.2 Exclude a Range of Routes

This example also specifies a range, but in this case the goal is to prevent the BGP speaker from advertising addresses in this range to its neighbors.

This example excludes all routes in the range 194.10.16.0 to 194.10.31.255 from the BGP routing table, which in turn, prevents them from being advertised.

Example:

```
BGP Config>add originate-policy exclusive
Network Prefix [0.0.0.0]? 194.10.16.0
Network Mask [0.0.0.0]? 255.255.240.0
Address Match (Exact/Range) [Exact]? range
Tag [0]? 0
```

5.4.2 Receive Policy Examples

5.4.2.1 Import all Routes from All BGP Neighbors

This example ensures that the BGP speaker import all routes from all of its neighbors into its IGP routing table.

Example:

```
BGP Config> add receive-policy inclusive
Network Prefix [0.0.0.0]? 0.0.0.0
Network Mask [0.0.0.0]? 0.0.0.0
Address Match (Exact/Range) [Exact]? range
Originating AS# [0]? 0
Adjacent AS# [0]? 0
IGP-metric [0]? 0
BGP Config>
```

IGP-metric specifies the metric value with which the accepted routes are imported into the speaker's IGP routing table. You are prompted to enter a value for *IGP-metric* only when setting up a policy for route inclusion.

5.4.2.2 Block Specific Routes from a Transit AS

This example prevents the BGP speaker from importing any routes originating at AS 168 from neighboring AS 165. You might use this command if you do not want the BGP speaker to receive any routes from AS 168 for security reasons.

Example:

```
BGP Config>add receive-policy exclusive
Network Prefix [0.0.0.0]?
Network Mask [0.0.0.0]?
Address Match (Exact/Range) [Exact]? range
Originating AS# [0]? 168
Adjacent AS# [0]? 165
```

5.4.3 Send Policy Examples

5.4.3.1 Restrict Route Advertisement to a Specific AS

This example restricts the BGP speaker. The speaker cannot advertise routes in the address range 143.116.0.0 to 143.116.255.255 that originate from AS 165 to autonomous system 168.

Example:

```
BGP Config>add send-policy exclusive
Network Prefix [0.0.0.0]? 143.116.0.0
Network Mask [0.0.0.0]? 255.255.0.0
Address Match (Exact/Range) [Exact]? range
Tag [0]? 165
Adjacent AS# [0]? 168
```

5.4.3.2 Advertise All Known Routes

This example ensures that the BGP speaker advertises all routes originated from its IGP, and all routes learned from its neighboring autonomous systems.

Example:

```
BGP Config> add send-policy inclusive
Network Prefix [0.0.0.0]? 0.0.0.0
Network Mask [0.0.0.0]? 0.0.0.0
Address Match (Exact/Range) [Exact]? range
Tag [0]? 0
Adjacent AS# [0]? 0
BGP Config>
```

5.5 BGP Configuration and Console Commands

This section summarizes and explains BGP configuration and console commands. These commands allow you to modify the BGP protocol behavior to meet your specific requirements. Some amount of configuration is necessary to produce a fully functional BGP router. Enter BGP configuration commands at the `BGP Config>` prompt. Enter BGP console commands at the `BGP>` prompt. Table 5–1 summarizes the BGP commands.

Table 5–1 BGP Command Summary

Command	Tasks	Function
? (Help)	Configuring and Monitoring	Lists the configuration or console commands, or lists the actions associated with specific commands.
Add	Configuring	Adds BGP neighbors.
Change	Configuring	Modifies information that was originally entered with the **add** command.
Clear	Configuring	Erases the BGP configuration.
Delete	Configuring	Deletes BGP configuration information that was entered with the **add** command.
Destinations	Monitoring	Displays all entries in the BGP routing table.
Disable	Configuring	Disables certain BGP features that were turned on by the **enable** command.
Enable	Configuring	Enables BGP speakers or BGP neighbors.
List	Configuring	Displays BGP configuration items.
Neighbors	Monitoring	Displays currently active neighbors.
Paths	Monitoring	Displays all available paths in the database.
Sizes	Monitoring	Displays the number of entries in various databases.
Exit	Configuring and Monitoring	Exits the process.

5.5 BGP Configuration and Console Commands

? (Help) C M

Lists the commands that are available from the current prompt level. You can also enter **?** after a specific command name to list its options.

Syntax: ?

Example:

```
?
ADD
CHANGE
CLEAR
DELETE
DISABLE
ENABLE
LIST
EXIT
```

Add C

Adds BGP information to your configuration.

Syntax: add

aggregate . . .
neighbor . . .
no-receive asnum . . .
originate-policy . . .
receive-policy . . .
send-policy. . .

add aggregate ***network prefix*** ***network mask***

The **add aggregate** command causes the BGP speaker to aggregate a block of addresses and advertise a single route to its BGP neighbors. You must specify the network prefix common to all the routes being aggregated and the prefix mask.

The following example illustrates how to aggregate a block of addresses from 194.10.16.0 through 194.10.31.255.

Example:

```
add aggregate
Network Prefix [0.0.0.0]? 194.10.16.0
Network Mask [0.0.0.0]? 255.255.240.0
```

5.5 BGP Configuration and Console Commands

When you add an aggregate definition, remember to define a policy to block the aggregated routes from being exported. If you do not, the router supports both the individual routes and the aggregate you have defined.

add neighbor ***neighbor IP address as# init timer connect timer hold timer keep alive timer tcp segment size***

Use the **add neighbor** command to define a BGP neighbor. The neighbor can be internal to the BGP speaker’s AS, or external. An internal neighbor must exist on the same network as the speaker.

Example:

```
add neighbor
Neighbor address [0.0.0.0]? 192.0.251.165
AS [0]? 165
Init timer [12]?
Connect timer [120]?
Hold timer [90]?
TCP segment size [1024]?
```

Neighbor IP Address Address of the neighbor you wish to peer with. It may be within your own autonomous system or in another autonomous system. If it is an external neighbor, both BGP speakers must share the same network. There is no such restriction for internal neighbors.

AS# Your own autonomous system number for internal neighbor or neighbor’s autonomous system number.

Init Timer Specifies the amount of time the BGP speaker waits to initialize resources and reinitiate transport connection with the neighbor in case the speaker has previously transitioned to IDLE state due to an error. If the error persists, this timer increases exponentially. The default is 12 seconds.

Connect Timer The amount of time the BGP speaker waits to reinitiate transport connection to its neighbor if the TCP connection fails while in either CONNECT or ACTIVE state. In the meantime, the BGP speaker continues to listen for any connection that may be initiated by its neighbor. The default is 120 seconds.

Hold Timer The length of time the BGP speaker waits before assuming that the neighbor is unreachable. Both neighbors exchange the configured information in OPEN message and choose the smallest of the two timers as their negotiated Hold Timer value. The default is 90 seconds.

Once neighbors have established BGP connection, they exchange KeepAlive messages at frequent intervals to ensure that the connection is still alive and the neighbors are reachable. The KeepAlive timer interval is calculated to be one third of the negotiated hold timer value. Hence, the Hold Timer value must be either zero or at least three seconds.

Note that on switched lines, you may wish to have the Hold Timer value of zero to save bandwidth by not sending KeepAlives at frequent intervals.

TCP Segment Size The maximum data size that may be exchanged on the TCP connection with a neighbor. This value is used for active TCP connection with the neighbor. It defaults to 1024, but can be set up to 65535.

add no-receive *asnum*

Use the **add no-receive** command to exclude updates from a particular AS.

Example:

```
add no-receive
Enter AS: [0]? 178
```

add originate-policy *(exclusive/ inclusive)   network prefix   network mask   address match (Exact/Range)   tag*

Use the **add originate-policy** command to create a policy that determines whether a specific address, or range of addresses, can be imported to the BGP speaker's routing table from the IGP routing table.

Exclusive Exclusive policies prevent route information from being included in the BGP speaker's routing table.

Inclusive Inclusive policies ensure that specific routes are included in the BGP speaker's routing table.

Network Prefix The network prefix for the addresses being affected.

Address Match The address, or range of addresses, that is affected by the policy statement.

Tag The value that was set for a particular AS. All tag values match that of the AS from which they were learned.

The following example includes all routes in the BGP speaker's IGP routing table to be advertised:

Example:

```
add originate-policy exclusive
Network Prefix [0.0.0.0]?
Network Mask [0.0.0.0]?
Address Match (Exact/Range) [Exact]? range
Tag [0]?
```

See Section 5.4 for detailed examples of this policy command.

add receive-policy ***(exclusive/ inclusive) network prefix network mask address match originating as# adjacent as# igp-metric (inclusive only)***

Use the **add receive-policy** command to determine what routes are imported to the BGP speaker's routing table.

Example:

```
add receive-policy exclusive
Network Prefix [0.0.0.0]? 10.0.0.0
Network Mask [0.0.0.0]?  255.0.0.0
Address Match (Exact/Range) [Exact]? range
Originating AS# [0]? 168
Adjacent AS# [0]? 165
```

add send-policy ***(exclusive/ inclusive) network prefix network mask address match tag adjacent as#***

Use the **add send-policy** command to create policies that determine which of the BGP speaker's learned routes are readvertised. These routes may be internal or external to the BGP speaker's AS.

Example:

```
add send-policy exclusive
Network Prefix [0.0.0.0]? 180.220.0.0
Network Mask [0.0.0.0]? 255.255.0.0
Address Match (Exact/Range) [Exact]? range
Tag [0]?
Adjacent AS# [0]? 25
```

Change C

Changes a BGP configuration item previously installed by the **add** command.

Syntax: change

aggregate . . .
neighbor . . .
originate-policy . . .
receive-policy . . .
send-policy. . .

change aggregate ***index# network prefix network mask***

This example changes the current aggregate (aggregate 1). The change causes aggregate 1 to use a different network prefix and mask to aggregate all routes in the address range from 128.185.0.0 to 128.185.255.255.

Example:

```
change aggregate 1
Network Prefix [128.185.0.0]? 128.128.0.0
Network Mask [255.255.0.0]? 255.192.0.0
```

change neighbor ***neighbor IP address as# init timer connect timer hold timer keep alive timer tcp segment size***

The following example changes the value of the hold timer to zero for neighbor 192.0.251.165:

Example:

```
change neighbor 192.0.251.165
AS [165]?
Init timer [12]?
Connect timer [60]?
Hold timer [12]? 0
TCP segment size [1024]?
```

5.5 BGP Configuration and Console Commands

change originate-policy ***index# (exclusive/ inclusive)   network prefix   network mask address match   tag***

Use the **change originate-policy** command to alter an existing originate policy definition.

This example alters the BGP speaker’s originate policy. Rather than excluding networks with prefix 194.10.16.0 from the IGP routing table, the policy now includes all routes.

Example:

```
change originate-policy
Enter index of originate-policy to be modified [1]?
Policy Type (Inclusive/Exclusive) [Exclusive]? inclusive
Network Prefix [194.10.16.0]? 0.0.0.0
Network Mask [255.255.240.0]? 0.0.0.0
Address Match (Exact/Range) [Range]?
Tag [0]?
```

change receive-policy ***index# (exclusive/inclusive)   network prefix   network mask address match   originating as#   adjacent as#   igp-metric (inclusive only)***

Use the **change receive-policy** command to alter an existing receive policy definition.

This example adds a restriction to the BGP speaker’s receive policy. Rather than import route information from every BGP peer into its IGP routing table, it now prevents routes from AS 165 from being imported.

Example:

```
change receive-policy
Enter index of receive-policy to be modified [1]?
Policy Type (Inclusive/Exclusive) [Inclusive]? exclusive
Network Prefix [0.0.0.0]?
Network Mask [0.0.0.0]?
Address Match (Exact/Range) [Range]?
Originating AS# [0]?
Adjacent AS# [0]? 165
```

change send-policy ***index# (exclusive/ inclusive) network prefix network mask address match tag adjacent as#***

Use the **change send-policy** command to alter an existing send policy to one that is more inclusive or more exclusive.

This example adds a restriction to the BGP speaker’s send policy. The restriction ensures that all routes in the address range 194.10.16.0 to 194.10.31.255 are excluded when advertising to autonomous system 165.

Example:

```
change send-policy
Enter index of send-policy to be modified [1]?
Policy Type (Inclusive/Exclusive) [Inclusive]? exclusive
Network Prefix [0.0.0.0]? 194.10.16.0
Network Mask [0.0.0.0]? 255.255.240.0
Address Match (Exact/Range) [Range]?
Tag [0]?
Adjacent AS# [0]? 165
```

Clear C

Erases the complete BGP configuration.

Syntax: clear

Delete C

Deletes an IP configuration item previously installed by the **add** command.

Syntax: delete

aggregate . . .
neighbor . . .
no-receive . . .
originate-policy . . .
receive-policy . . .
send-policy. . .

delete aggregate ***index#***

You must specify the index number of the aggregate you want to delete. The index number is equivalent to the AS number.

Example:

```
delete aggregate 1
```

5.5 BGP Configuration and Console Commands

delete neighbor ***neighbor IP address***

Use this command to delete a BGP neighbor. You must specify the neighbor's network address.

Example:

```
delete neighbor 192.0.251.165
```

delete no-receive ***as***

Use this command to delete the no-receive policy set up for a particular AS. You must specify the AS number.

Example:

```
delete no-receive 168
```

delete originate-policy ***index#***

Use this command to delete a specific originate policy. You must specify the index number associated with the policy.

Example:

```
delete originate-policy 2
```

delete receive-policy ***index#***

Use this command to delete a specific receive policy. You must specify the index number associated with the policy.

Example:

```
delete receive-policy
Enter index of receive-policy to be deleted [1]?
```

delete send-policy ***index#***

Use this command to delete a specific send policy. You must specify the index number associated with the policy.

Example:

```
delete send-policy 4
```

Destinations M

Dumps all BGP routing table entries, or display information on routes advertised to, or received from, specified BGP neighbor addresses (destinations).

Syntax: destinations
net address/net address net mask
advertised-to *network address*
received-from *network address*

Example:

```
destinations
Network         Mask       NextHop         MED AAG AGRAS ORG AS-Path
128.185.0.0     FFFF0000   192.0.251.165   0   No  0     IGP
142.4.0.0       FFFF0000   192.0.190.178   0   No  0     IGP seq[178]
143.116.0.0     FFFF0000   128.185.252.168 0   No  0     IGP seq[168]
192.0.190.0     FFFFFF00   192.0.251.165   0   No  0     IGP
192.0.251.0     FFFFFF00   192.0.251.165   0   No  0     IGP
194.10.16.0     FFFFF000   192.0.251.167   0   No  167   IGP seq[167]
```

Network Indicates the IP addresses of the destinations in the routing table.

Mask The address mask for each entry in the table.

NextHop Indicates the address of the router to use as the forwarding address towards this destination.

MED Specifies a multi-exit discriminator value, used to discriminate among multiple entry/exit points to the same AS.

AAG Indicates whether the route is an aggregate or not. Values are `Yes` or `No`.

AGRAS The number of the AS that aggregated the route.

5.5 BGP Configuration and Console Commands

destinations *net-address*

Displays detailed information on the specified route or destination network. The command shows how a specific route was learned, the best path to a specific destination, the metric associated with the route, and other information.

Example:

```
destinations 142.4.0.0
Network         Mask     NextHop        MED AAG AGRAS ORG AS-Path
142.4.0.0       FFFF0000 192.0.251.165  0   No  0     IGP seq[165-178]
Dest:142.4.0.0, Mask:FFFF0000, Age:180, Upd#:13,
LastSent:0001:53:32 Eligible paths: 2

PathID: 8 - (Best Path)
        ASpath: seq[165-178]
        Origin: IGP, Pref: 507, LocalPref: 0
        Metric: 0, Weight: 0, MED: 0
        NextHop: 192.0.251.165  , Neighbor: 192.0.251.165
        AtomicAggr: No

PathID: 21
        ASpath: seq[168-165-178]
        Origin: IGP, Pref: 505, LocalPref: 0
        Metric: 0, Weight: 0, MED: 0
        NextHop: 128.185.250.168, Neighbor: 128.185.250.168
        AtomicAggr: No
```

Network Indicates the IP address of the specified destination.

Mask The address mask for this entry.

NextHop Indicates the address of the router to use as the forwarding address towards this destination.

MED Specifies a multi-exit discriminator value, used to discriminate among multiple entry/exit points to the same AS.

This field is blank if the *Network* destination is not being used for forwarding.

AAG Indicates whether the route is an aggregate or not. Values are `Yes` or `No`.

This field is blank if the *Network* destination is not being used for forwarding.

AGRAS The number of the AS that aggregated the route.

This field is blank if the *Network* destination is not being used for forwarding.

ORG Specifies the originator of this destination: either EGP, IGP, or Incomplete (originated by some other means not known).

This field is blank if the *Network* destination is not being used for forwarding.

AS-PATH Enumeration of ASs along the path.

- **seq**: Sequence of ASs in order in the path
- **set**: Set of ASs in the path

This field is blank if the *Network* destination is not being used for forwarding.

Dest Indicates the IP address of the specified destination.

Mask The address mask for this entry.

Age Indicates the age of this entry in seconds.

Upd# Indicates the sequence numberof the last update message for this destination.

LastSent Indicates the time that the last message was sent to this destination.

Eligible paths Indicates the number of eligible paths to this destination.

Path ID Indicates the unique identifier for each path.

ASpath Enumeration of ASs along the path.

- **seq:** Sequence of ASs in order in the path
- **set:** Set of ASs in the path

Origin Indicates the originator of the destination. This is either EGP, IGP, or Incomplete (originated by some other means not known).

LocalPref Indicates the originating router’s degree of preference for the destination.

Metric Specifies the path metric with which the route is imported.

Weight Specifies the path weight.

MED Specifies a multi-exit discriminator value, used to discriminate among multiple entry/exit points to the same AS.

NextHop Indicates the address of the router to use as the forwarding address for destinations reachable via the given path.

AtomicAggr Indicates whether the router advertising the path has included the path in an atomic-aggregate.

destinations *net-address net-mask*

Displays detailed information on the specified route or destination network. The command shows how a specific route was learned, the best path to a specific destination, the metric associated with the route, and other information.

This command is useful in cases where multiple network addresses have the same prefix and different masks. In such cases, specifying the network mask narrows the scope of the information presented.

Example:

```
destinations 194.10.16.0 255.255.240.0
Dest:194.10.16.0, Mask:FFFFF000, Age:0, Upd#:3, LastSent:0002:00:00
Eligible paths: 1
PathID: 0 - (Best Path)
        ASpath:
        Origin: IGP, Pref: 0, LocalPref: 0
        Metric: 0, Weight: 0, MED: 0
        NextHop: 194.10.16.167  , Neighbor: 194.10.16.167
        AtomicAggr: No, Aggregator AS167/194.10.16.167
```

destinations advertised-to *net-address*

Lists all routes advertised to the specified BGP neighbor.

Example:

```
destinations advertised-to
BGP neighbor address [0.0.0.0]? 192.0.251.165
Destinations advertised to BGP neighbor 192.0.251.165

Network       Mask     NextHop         MED AAG AGRAS ORG AS-Path
194.10.16.0   FFFFF000 194.10.16.167   0   No  167   IGP
192.0.190.0   FFFFFF00 192.0.251.165   0   No  0     IGP seq [165]
142.4.0.0     FFFF0000 192.0.251.165   0   No  0     IGP seq [165-178]
143.116.0.0   FFFF0000 128.185.250.168 0   No  0     IGP seq [168]
```

destinations received-from *net-address*

Lists all routes received from the specified BGP neighbor.

Example:

```
destinations received-from
BGP neighbor address [0.0.0.0]? 128.185.250.167

        Destinations obtained from BGP neighbor 128.185.250.167

Network     Mask     NextHop         MED AAG AGRAS ORG AS-Path
194.10.16.0 FFFFF000 128.185.250.167 0   No  167   IGP seq[167]
192.0.190.0 FFFFFF00 128.185.250.167 0   No  0     IGP seq[167-165]
142.4.0.0   FFFF0000 128.185.250.167 0   No  0     IGP seq[167-165-178]
```

Disable C

Disables a previously enabled BGP neighbor or speaker. Note that neighbors are implicitly enabled whenever added with the **add** command.

Syntax: disable

bgp speaker
neighbor . . .

disable bgp speaker

Example:

```
disable bgp speaker
```

disable neighbor *neighbor-IP-address*

Example:

```
disable neighbor 192.0.190.178
```

5.5 BGP Configuration and Console Commands

Enable C

Activates the BGP features, capabilities, and information added to your BGP configuration.

Syntax: enable

bgp speaker
neighbor . . .

enable bgp speaker *as# tcp segment size*

Use the **enable bgp speaker** command to enable the BGP protocol.

Example:

```
enable bgp speaker
AS [0]? 165
TCP segment size [1024]?
```

enable neighbor *neighbor IP address*

Use this command to enable a BGP neighbor.

Example:

```
enable neighbor 192.0.190.178
```

List C

Displays various pieces of the IP configuration data, depending on the particular subcommand invoked.

Syntax: list

aggregate
all
bgp speaker
neighbor
no-receive
originate-policy
receive-policy
send-policy

5.5 BGP Configuration and Console Commands

list aggregate

Use the **list aggregate** command to show all aggregated routes defined with the **add aggregate** command.

Example:

```
list aggregate
Aggregation:
Index     Prefix             Mask
1         194.10.16.0        255.255.240.0
```

list all

Use the **list all** command to list the BGP neighbors, policies, aggregated routes, and no-receive-as records in the current BGP configuration.

Example:

```
list all
                BGP Protocol:          Enabled
                AS:                    167
                TCP-Segment Size:      1024
Neighbors and their AS:
                                    Init    Conn    Hold    TCPSEG
Address            State     AS     Timer   Timer   Timer   Size
128.185.250.168    ENABLD    168    12      60      12      1024
192.0.251.165      ENABLD    165    12      60      12      1024

Receive-Policies:
Index  Type  Prefix       Mask      Match  OrgAS  AdjAS IGPmetric
1      INCL  0.0.0.0      0.0.0.0   Range  0      0      0

Send-Policies:
Index  Type  Prefix       Mask      Match  Tag    AdjAS
1      INCL  0.0.0.0      0.0.0.0   Range  0      0

Originate-Policies:
Index  Type  Prefix       Mask             Match  Tag
1      EXCL  194.10.16.0  255.255.240.0    Range  0

Aggregation:

Index  Prefix             Mask
1      194.10.16.0        255.255.240.0
No no-receive-AS records in configuration.
```

5.5 BGP Configuration and Console Commands

list bgp speaker

Use the **list bgp speaker** command to derive information on the BGP speaker. The information provided is shown below:

Example:

```
list bgp speaker
BGP Protocol:          Enabled
AS:                    165
TCP-Segment Size:      1024
```

list neighbor

Use the **list neighbor** command to derive information on BGP neighbors.

Example:

```
list neighbor
Neighbors and their AS:
                                    Init    Conn    Hold    TCPSEG
Address            State     AS     Timer   Timer   Timer   Size
128.185.252.168    ENABLD    168    12      60      12      1024
192.0.190.178      DISBLD    178    12      60      12      1024
192.0.251.167      ENABLD    167    12      60      12      1024
```

list no-receive

Use the **list no-receive** command to derive information on *no-receive-AS* definitions that were added to the BGP configuration.

Example:

```
list no-receive
AS-PATH with following ASs will be discarded:
AS  178
AS  165
```

list originate-policy ***all*** or ***index*** or ***prefix***

Use the **list originate-policy** command to derive information on the originate policies that were added to the BGP configuration.

Example:

```
list originate-policy
Originate-Policies:
Index  Type  Prefix           Mask              Match  Tag
1      EXCL  194.10.16.0      255.255.240.0     Range  0
2      INCL  0.0.0.0          0.0.0.0           Range  0
```

list receive-policy adj-as-number *all* or *index* or *prefix*

Use the **list receive-policy** command to derive information on the receive policies that were added to the BGP configuration. You can display all receive policies defined for an AS, or display policies by index or prefix number.

Example:

```
list receive-policy
Receive-Policies:
Index  Type  Prefix      Mask        Match  OrgAS  AdjAS  IGPmetric
1      EXCL  0.0.0.0     0.0.0.0     Range  178    165
2      INCL  0.0.0.0     0.0.0.0     Range  0      0      0
```

list send-policy adj-as-number *all* or *index* or *prefix*

Use the **list send-policy** command to display information on send policies defined for specified ASs. You can display all send policies defined for an AS, or display policies by index or prefix number.

Example:

```
list send-policy
Send-Policies:
Index  Type  Prefix      Mask           Match Tag    AdjAS
1      EXCL  194.10.16.0 255.255.240.0  Range   0    165
2      INCL   0.0.0.0          0.0.0.0  Range   0    0
```

Neighbors M

Displays information on all active BGP neighbors.

Syntax: neighbors *internet-address*

Example:

```
neighbors
IP-Address       State        DAY-HH:MM:SS  BGP-ID           AS     Upd#

128.185.252.168  Established  000-00:48:52  128.185.142.168  168     16
192.0.190.178    Established  000-02:01:49  142.4.140.178    178     16
192.0.251.167    Established  000-02:01:45  194.10.16.167    167     16
```

5.5 BGP Configuration and Console Commands

Paths M

Use the BGP **paths** command to display the paths stored in the path description database.

Syntax: paths

Example:

```
paths
PathId   NextHop   MED  AAG   AGRAS  RefCnt   ORG   AS-Path

0       10.2.0.3    0   No     0      2      IGP
4       192.2.0.2   0   No     0      2      IGP    seq[2]
5       192.2.0.2   0   No     2      1      IGP    seq[2]
6       192.2.0.2   0   No     0      1      IGP    seq[2-1]
7       10.2.0.168  0   No     0      4      IGP
8       192.3.0.1   0   No     0      2      IGP    seq[1]
9       192.2.0.2   0   No     2      1      IGP    seq[2]
10      10.2.0.3    0   No     0      1      IGP
```

PathId Path identifier.

NextHop Specifies the address of the router to use as the forwarding address for the destinations that can be reached via the given path.

MED Specifies the multi exit descriminator used to discriminate among multiple entry/exit points to the same AS.

AAG Indicates whether the path was atomic-aggregated; that is, the router that is advertising the given path has selected a less specific route over the more specific one when presented with overlapping routes.

AGRAS Indicates the AS number of the BGP speaker that aggregated the routes.

RefCnt Indicates the number of path entities referring to the descriptor.

ORG Specifies the originator of the advertised destinations in the given path: either EGP, IGP, or Incomplete (originated by some other means not known).

AS-Path Enumeration of ASs along the path.

- **seq**: Sequence of ASs in order in the path
- **set**: Set of ASs in the path

Sizes M

Use the BGP **sizes** command to display the number of entries stored in the various databases.

Syntax: sizes

Example:

```
sizes
# Paths:                                11
# Path descriptors:                      7
Update sequence#:                       22
# Routing tbl entries (allocated):       6
# Current tbl entries (not imported):    0
# Current tbl entries (imported to IGP): 3
```

Paths Total number of eligible paths for all the routes in the BGP routing table.

Path descriptors Total number of path descriptors in the database used to hold common path information.

Update sequence# Indicates the current update sequence number.

Routing tbl entries (allocated) Indicates the number of entries in BGP routing table.

Current tbl entries (not imported) Indicates the number of BGP routes not imported into IGP.

Current tbl entries (imported to IGP) Indicates the number of BGP routes imported into IGP.

IP-Address Specifies the IP address of the BGP neighbor.

State Specifies the state of the connection. Possible states are:

- **Connect** Waiting for the TCP connection to the neighbor to be completed.
- **Active** In the event of TCP connection failure, the state is changed to Active, and the attempt to acquire the neighbor continues.
- **OpenSent** In this state OPEN was sent, and BGP waits for an OPEN message from the neighbor.
- **OpenConfirm** In this state a KEEPALIVE was sent in response to neighbor's OPEN, and waits for a KEEPALIVE/NOTIFICATION from the neighbor.
- **Established** A BGP connection was successfully established, and can now start to ex-

BGP-ID Specifies the neighbor's BGP identification number.

AS Specifies the neighbor's AS number.

Upd# Specifies the sequence number of the last UPDATE message sent to the neighbor.

Neighbor *internet-address*

Use the **neighbor** command to display detailed data on a particular BGP neighbor.

Example:

```
neighbor 192.0.251.167
Active Conn:Sprt:1026 Dprt:179 State:Established KeepAlive/Hold Time 4/12
Passve Conn:None
TCP connection errors: 0          TCP state transitions: 0

BGP Messages:     Sent   Received                     Sent   Received
Open:             1      1          Update:           11     11
Notification:     0      0          KeepAlive:        1828   1830
Total Messages:   1840   1842

Msg Header Errs:  Sent   Received                     Sent   Received
Conn sync err:    0      0          Bad msg length:   0      0
Bad msg type:     0      0

Open Msg Errs:    Sent   Received                     Sent   Received
Unsupp versions:  0      0          Unsupp auth code: 0      0
Bad peer AS ident:0      0          Auth failure:     0      0
Bad BGP ident:    0      0          Bad hold time:    0      0

Update Msg Errs:  Sent   Received                     Sent   Received
Bad attr list:    0      0          AS routing loop:  0      0
Bad wlkn attr:    0      0          Bad NEXT_HOP atr: 0      0
Mssng wlkn attr:  0      0          Optional atr err: 0      0
Attr flags err:   0      0          Bad netwrk field: 0      0
Attr length err:  0      0          Bad AS_PATH attr: 0      0
Bad ORIGIN attr:  0      0

Total Errors:     Sent   Received                     Sent   Received
Msg Header Errs:  0      0          Hold Timer Exprd: 0      0
Open Msg Errs:    0      0          FSM Errs:         0      0
Update Msg Errs:  0      0          Cease:            0      0
```

Exit C M

Leave the BGP configuration or monitoring module and return to the previous prompt.

Syntax: exit

Example:

```
exit
```

6

Configuring and Monitoring Bandwidth Reservation

This chapter describes how to access the Bandwidth Reservation System (BRS) configuration and console prompts and how to use the bandwidth reservation commands.

6.1 Displaying the Bandwidth Reservation Configuration Prompt

To access bandwidth reservation configuration commands and to configure bandwidth reservation on your router, perform the following steps:

1. At the ***** prompt, enter **talk 6**.
2. At the `Config>` prompt, type **feature brs**.
3. At the `BRS Config>` prompt, type **interface #**, where # is the number of the interface you want to configure.
4. At the `BRS [i 0] Config>` prompt, type **enable**. (This is the interface prompt level, and the interface number is zero in this instance.)
5. For Frame Relay interfaces, select PVCs using the **circuit** command. At the `[BRS i 0] [dlci 16] Config>` prompt, type **enable.** (This is the circuit prompt, and the circuit number is 16 in this example.)
6. Restart your router.
7. Repeat steps 1 through 4 to configure bandwidth reservation for the particular interface that you have enabled.
8. At the `BRS [i 0] Config>` prompt, configure the bandwidth reservation parameters for the selected interface by using the appropriate configuration commands discussed in this chapter. If this is a Frame Relay interface, configure circuit classes at this prompt.

6.1 Displaying the Bandwidth Reservation Configuration Prompt

9. For Frame Relay interfaces, select PVCs using the **circuit** command. At the `[BRS i 0] [dlci 16] Config>` prompt, configure the bandwidth reservation parameters for the selected circuit using configuration commands discussed in this chapter. (This is the circuit prompt, and the circuit number is 16 in this example.)

10. Restart your router.

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 bandwidth reservation, you must select the interface to be configured. In step 4, the prompt indicates that the selected interface's number is zero.

You must enable bandwidth reservation for the selected interface and restart your router before configuring the particular interface.

To return to the `Config>` prompt at any time, enter the **exit** command at the `BRS Config>` prompt.

6.1.1 Configuration Procedure for Bandwidth Reservation System

The following procedure demonstrates the configuration of bandwidth reservation on a RouteAbout Access EI bridging router.

In this procedure, each parameter that has a default value is given that value. For parameters that have no default (for example, addresses), the procedure uses an arbitrary value.

Note: You should check all values carefully and change them as necessary for your individual implementation.

The following steps summarize the procedure you can use to configure the BRS feature on router interfaces:

1. Talk to the router configuration process and reach the Bandwidth Reservation System (BRS) feature.
2. Identify and enable the interface required for BRS.
3. Restart the router.

4. Talk to the router and reach the bandwidth reservation feature.
5. Configure bandwidth reservation parameters.

6.1.1.1 Example 1 — Basic Configuration Procedure

To configure the Bandwidth Reservation System (BRS) on the router, perform the following steps. You may have to change the values or parameters given here according to your network setup.

Note: You must restart the router to put these values in effect.

1. Talk to the router and access bandwidth reservation.

```
* t 6
Config> feature BRS
```

2. Point to and enable the interface required for BRS.

```
BRS Config> interface 1
BRS [i 1] Config> enable
```

3. Restart the router.

```
BRS [i 1] Config> exit
BRS Config> exit
Config> ^P
* restart
```

4. Talk to the router and access bandwidth reservation.

```
* t 6
Config> feature BRS

Bandwidth Reservation User Configuration

BRS Config>
```

5. Configure bandwidth reservation parameters.

```
BRS Config> interface 1
BRS [i 1] Config> add-class alpha 10
BRS [i 1] Config> assign
Protocol or Filter name [IP]? IP
Class name [DEFAULT]? alpha
Priority <URGENT/HIGH/NORMAL/LOW> [NORMAL] NORMAL
BRS [i 1] Config> tag 1
BRS [i 1] Config> exit
BRS Config> exit
Config>
```

6.1 Displaying the Bandwidth Reservation Configuration Prompt

6.1.1.2 Example 2 — Configuration Procedure Using a MAC Filter Tag

In this example procedure, we will create a MAC filter and use it in BRS to reserve a proportion of the bandwidth for the filter and for a selected protocol (IP).

1. Talk to the router and access the MAC filter feature. (Refer to the *Bridging Configuration Guide* for details about using this feature.)

```
* t 6
Config> feature MCF
MAC Filtering user configuration
Filter Config>
```

2. Bandwidth Reservation operates on outbound packets only, so create an OUTPUT filter for the interface we want to filter (interface 3, in this example).

```
Filter Config>create filter
Enter a direction to filter (INPUT or OUTPUT) [INPUT]? output
Enter an interface to filter [0]?3
```

3. Create a list for the MAC Filter, and display the lists and filters using the **list all** command to discover the ID number of the new filter.

```
Filter Config>create list mclist
Filter Config>list all
Filtering: disabled
Filter List                   Action
 -----------                   ------
mclist                        INCLUDE

 Filters
 -------
Id   Default   State     Ifc  Dir       Cache
 --   -------   -----     ---  ---       ------
1    INCLUDE   ENABLE    0    OUTPUT    16
```

4. Use the **attach** command to associate the filter with the list.

```
Filter Config>attach
Enter a filter-list name []? mclist
Enter a filter number [1]?1
```

5. Use the **update** command to configure the list entry we have just created. We want to **add** a destination MAC address to the list and **set** a tag that identifies the list. The tag must be in the range 1 through 5.

```
Filter Config>update mclist
Filter ’mclist’ Config>add destination
Enter MAC Address []? 00-00-03-00-12-34
Enter MAC Mask [ffffffffffff]?
Filter ’mclist’ Config>set tag
Enter a tag value [1]?
```

6. Use the **list** command to display the configuration details of this list (**canonical** for Ethernet destinations or **noncanonical** for Token Ring).

```
Filter ’mclist’ Config>lis canonical
Action: TAG( 1)
 Id  Type  MAC Address                   Mask
 --  ----  ------------                  ----
1   DST   00-00-03-00-12-34             FF-FF-FF-FF-FF-FF
```

7. Exit from the MAC filtering feature and enter the BRS feature.

```
Filter ’mclist’ Config>exit
Filter Config>exit
Config>fea brs
Bandwidth Reservation User Configuration
```

8. Enable BRS on the interface we are configuring (interface 3).

```
BRS Config>int 3
BRS [i 3] Config>enable
```

9. Restart the router and return to the BRS feature. Access the enabled interface and list the default bandwidth reservation configuration.

```
BRS Config>int 3
BRS [i 3] Config>list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
interface number 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 DN with default priority
    protocol VIN with default priority
    protocol IPX with default priority
    protocol OSI with default priority
    protocol APL with default priority
    protocol AP2 with default priority
    protocol ASRT with default priority
assigned tags:

default class is DEFAULT with priority NORMAL
```

6.1 Displaying the Bandwidth Reservation Configuration Prompt

In the previous listing, you can see that 50 percent of the bandwidth on interface 3 is already reserved in two classes. Ten percent of the bandwidth is reserved for a class called LOCAL, and 40 percent is reserved for the DEFAULT class, which contains all of the protocols.

10. Use the **assign ?** command to see what can be assigned to a class. In this example, we want to create a new class called **myclass,** which will contain all IP protocol packets, and schedule them at priority **high**.

```
BRS [i 3] Config>assign ?
IP
ARP
DN
IPX
OSI
APL
AP2
ASRT
TUNNELING-IP
SDLC-IP
RLOGIN-IP
TELNET-IP
NETBIOS
SNA
SNMP-IP
MULTICAST-IP
TAG1
TAG2
TAG3
TAG4
TAG5
Protocol or filter name [IP]?
Protocol or filter name [IP]? ip
Class name [DEFAULT]? myclass
Priority <URGENT/HIGH/NORMAL/LOW> [NORMAL]? high
```

11. Use the **assign** command to add the MAC filter tag to class **myclass**, with **normal** priority. This will mean that IP packets will take precedence over packets matching this tag.

```
BRS [i 3] Config>ass tag1 myclass normal
```

12. Using the **list** command again, we can see that the IP protocol has been removed from the **default** class and is assigned to class **myclass**, along with **tag1.**

```
BRS [i 3] Config>list
BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
interface number 3
total bandwidth allocated 80%
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 ARP with default priority
    protocol DN with default priority
    protocol VIN with default priority
    protocol IPX with default priority
    protocol OSI with default priority
    protocol APL with default priority
    protocol AP2 with default priority
    protocol ASRT with default priority

class myclass has 30% bandwidth allocated

  the following protocols and filters are assigned:
    protocol IP with priority HIGH
    filter TAG1 with priority NORMAL

assigned tags:

default class is DEFAULT with priority NORMAL

BRS [i 3] Config>
```

Bandwidth reservation operates only when the line becomes fully loaded. Each class is allocated a percentage of the bandwidth, which is reserved for any outbound packets matching its assigned protocols or MAC filters.

There are four sub-queues in each class, one for each priority level (Urgent, High, Normal, and Low). The highest priority queues are processed first, and must be empty before the next lower level sub-queue is processed. All protocols or filters of the same priority are processed on a *round-robin* basis. This means you can ensure that high-priority traffic always has bandwidth to get through, even if the line is fully loaded, and that other traffic can share the remainder of the bandwidth.

Refer to the *Routing Protocols Reference Guide* for more information about how bandwidth reservation functions.

6.2 Displaying the Bandwidth Reservation Console Prompt

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

1. At the OPCON prompt (`*`), type **t 5**.
2. At the GWCON prompt (+), type **feature brs**.
3. At the `BRS>` prompt, type **interface #**. (Enter the number [#] of the interface that you want to monitor.)
4. For Frame Relay only, type **circuit #** or issue one of the circuit class monitoring commands (for example, **counters-circuit-class).**
5. At the prompt, type the appropriate console command**.** (Refer to Section 6.3 for bandwidth reservation console commands.)

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).

The **counters #** command allows you to display statistics on BRS traffic for the selected interface.

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

Once you access the bandwidth reservation console prompt (`BRS>`), you can enter any of the specific console commands described in Table 6–1.

6.3 Bandwidth Reservation Configuration and Console Commands

Table 6–1 describes the bandwidth reservation configuration and console commands. The commands marked by an asterisk are used only with Frame Relay. (The asterisk is not part of the command.)

6.3 Bandwidth Reservation Configuration and Console Commands

Table 6–1 Bandwidth Reservation Configuration and Console Commands

Command	Tasks	Function
? (Help)	Configuring and Monitoring	Displays the bandwidth reservation configuration commands or lists options for specific commands (if available).
Add-circuit-class*	Configuring	Sets the name of a circuit class and its percentage of bandwidth.
Add-class	Configuring	Allocates a designated amount of bandwidth to a user-defined bandwidth class.
Assign	Configuring	Assigns a protocol or filter to a reserved class.
Assign-circuit*	Configuring	Assigns a specified circuit to the specified circuit class.
Change-circuit-class*	Configuring	Changes the percentage of the bandwidth to be used by the group of circuits assigned to the designated class.
Change-class	Configuring	Changes the amount of bandwidth configured for a bandwidth class.
Circuit	Monitoring	Displays all the bandwidth reservation commands or lists subcommand options for specific commands (if available).
Circuit #	Configuring	Selects the DLCI of a Frame Relay permanent virtual circuit.
Clear	Monitoring	Clears the current reservation counters and stores them as **last** command counters. Counters are listed by class usage.
Clear-block	Configuring	Clears the current reservation configuration from SRAM. **Note:** This command requires a router restart.
Clear-circuit-class	Monitoring	Clears the reservation counters for all the circuit classes of the interface.
Counters	Monitoring	Displays the current counters.
Counters-circuit-class	Monitoring	Displays the current counters for all the circuit classes of the interface.
Deassign	Configuring	Restores a specified protocol or filter to its default class and priority.
Deassign-circuit*	Configuring	Deassigns the specified circuit from the circuit class to which it was assigned.
Default-circuit-class*	Configuring	Assigns the name of the default circuit class.
Default-class	Configuring	Sets the default class and priority to a desired value.

6.3 Bandwidth Reservation Configuration and Console Commands

Table 6–1 Bandwidth Reservation Configuration and Console Commands (Continued)

Command	Tasks	Function
Del-circuit-class*	Configuring	Deletes the specified circuit class.
Del-class	Configuring	Deletes a previously configured bandwidth class from the specified interface.
Disable	Configuring	Disables bandwidth reservation on the interface or Frame Relay circuit. **Note:** This command requires a router restart.
Enable	Configuring	Enables bandwidth reservation on the interface or Frame Relay circuit. **Note:** This command requires a router restart.
Interface	Configuring	Selects the serial interface that runs bandwidth reservation. Use this command to enable BRS on an interface. **Note:** This command must be entered BEFORE using any other configuration commands.
Last	Monitoring	Displays the last saved statistics.
Last-circuit-class	Monitoring	Displays the last saved statistics for all the circuit classes of the interface.
List	Configuring and Monitoring	Displays the currently defined bandwidth classes by their guaranteed percentage rates and priority queuing values stored in the SRAM display. Also displays the assigned protocols and filters. (For Frame Relay, this command provides two levels of information.)
Show	Configuring	Displays the currently defined bandwidth classes stored in RAM. (For Frame Relay, this command provides two levels of information.)
Tag	Configuring	Assigns a class and priority to a filter that was tagged during the configuration of the MAC filtering feature.
Untag	Configuring	Removes the tag/tag name relationship and the tag name from the list of assignable filters.
Exit	Configuring and Monitoring	Exits from one BRS level to another or exits the bandwidth reservation configuration process.

Except for the commands marked with an asterisk, which are only for Frame Relay, the configuration commands in Table 6–1 are the same for configuring bandwidth reservation for the Proteon Serial Line protocol, the Point-to-Point protocol (PPP), Frame Relay, Integrated Services Digital Network (ISDN), and V.25 *bis*.

6.3 Bandwidth Reservation Configuration and Console Commands

Note: When the **clear-block**, **disable**, **enable**, **list,** and **show** commands are issued from within the BRS interface level, they affect or list the bandwidth reservation information configured for the selected interface. When these commands are issued from within the BRS circuit level, they affect only the FR bandwidth reservation information configured for the permanent virtual circuit (PVC.).

Note: Before using the bandwidth reservation commands, keep the following in mind:

- You must use the **interface** command to select a serial 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.

? (Help) C M

At the BRS prompt, use the **? (help)** command to list the available commands from the current prompt level. You can also enter **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

C At the `BRS [i #] [dlci #] Config>` and the `BRS Config>` prompts, the following commands are listed:

```
INTERFACE
LIST
EXIT
```

6.3 Bandwidth Reservation Configuration and Console Commands

C At the `BRS [i #] Config>` prompt (for non-FR), the following commands are listed:

```
ENABLE
DISABLE
ADD-CLASS
DEL-CLASS
CHANGE-CLASS
DEFAULT-CLASS
ASSIGN
DEASSIGN
INTERFACE
LIST
SHOW
CLEAR-BLOCK
TAG
UNTAG
EXIT
```

C At the `BRS [i #] Config>` prompt (for FR), the following commands are listed:

```
ENABLE
DISABLE
CIRCUIT
ADD-CIRCUIT-CLASS
DEL-CIRCUIT-CLASS
CHANGE-CIRCUIT-CLASS
DEFAULT-CIRCUIT-CLASS
ASSIGN-CIRCUIT
DEASSIGN-CIRCUIT
LIST
SHOW
CLEAR-BLOCK
EXIT
```

M At the `BRS>` prompt:

```
INTERFACE
EXIT
```

M At the non-Frame Relay `BRS [i #]>` prompt:

```
COUNTERS
CLEAR
LAST
EXIT
```

6.3 Bandwidth Reservation Configuration and Console Commands

M For Frame Relay, at the `BRS [i #]>` prompt, the following commands are listed:

```
COUNTERS-CIRCUIT-CLASS
CLEAR-CIRCUIT-CLASS
LAST-CIRCUIT-CLASS
CIRCUIT
EXIT
```

M For Frame Relay, at the `BRS [i #] [dlci #]>` prompt, the following commands are listed:

```
COUNTERS
CLEAR
LAST
EXIT
```

Add-circuit-class **C**

Use this command at the interface level to allocate a designated amount of bandwidth to be used by the group of Frame Relay circuits assigned to the circuit class.

Syntax: add-circuit-class *class-name* *%*

Example:

```
add-circuit-class alpha 10
```

Here *class-name* is the ASCII string assigned as the name of the circuit class, and *%* is a percentage of the bandwidth, between 1 and 100, of the interface.

Add-class **C**

Allocates a designated amount of bandwidth to a user-defined bandwidth class.

Syntax: add-class *class-name* *%*

Example:

```
add test 20
```

Here *class-name* is the ASCII string assigned as the name of the bandwidth class, and *%* is a percentage of the bandwidth of the interface or Frame Relay circuit.

6.3 Bandwidth Reservation Configuration and Console Commands

Assign C

Assigns specified tags, protocol packets, or filters to a given class and priority within that class. The four priority types include:

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

Syntax: assign *protocol* **or** *TAG* **or** *filter* *class-name*

Example:

```
assign AP2 test
priority <URGENT/HIGH/NORMAL/LOW>: [NORMAL]? low

protocol AP2 maps to class test with priority LOW
```

Assign-circuit C

Use this command at the interface level to assign the specified circuit (DLCI) to the specified circuit class.

Syntax: assign-circuit *#* *class-name*

Example:

```
assign-circuit 16 pubs
```

Change-circuit-class C

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

Syntax: change-circuit-class *class-name* *%*

Example:

```
change-circuit-class alpha 20
```

6.3 Bandwidth Reservation Configuration and Console Commands

Change-class C

Changes the amount of bandwidth configured for a bandwidth class.

Syntax: change-class *class-name* **or** *class# %*

Example:

```
change test 10
```

Circuit C

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

Syntax: circuit *permanent-virtual-circuit#*

Example:

```
circuit 16
```

When the FR circuit is enabled, the following commands may be used at the circuit prompt:

```
ADD-CLASS
ASSIGN
CHANGE-CLASS
CLEAR-BLOCK
DEASSIGN
DEFAULT-CLASS
DEL-CLASS
DISABLE
EXIT
LIST
SHOW
TAG
UNTAG
```

6.3 Bandwidth Reservation Configuration and Console Commands

Circuit M

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

Syntax: circuit *permanent-virtual-circuit #*

Example:

```
circuit 16
```

After the FR circuit is selected, the following commands can be used at the circuit prompt:

```
COUNTERS
CLEAR
LAST
EXIT
```

Clear M

Clears from RAM the current bandwidth reservation counters for the selected interface or Frame Relay circuit, and stores them as counters that can be made available by the **last** command.

Syntax: clear

Example:

```
clear
```

Clear-block C

Clears the current bandwidth reservation configuration from SRAM for the current interface or Frame Relay PVC. This command requires a router restart.

Syntax: clear-block

Example:

```
clear-block
You are about to clear BRS configuration information
Are you sure you want to do this (Yes or No): y
BRS [i 0] Config>
```

6.3 Bandwidth Reservation Configuration and Console Commands

Clear-circuit-class M

Enter at the `BRS [i #]>` prompt. It clears the current bandwidth reservation counters for the circuit classes of the selected Frame Relay interface. This command clears the counters from RAM and stores them as counters that you can display with **last-circuit-class.**

Syntax: clear-circuit-class

Example:

```
clear
```

Counters M

Displays statistics describing bandwidth reservation traffic for the selected interface or Frame Relay circuit according to the configured classes.

Syntax: counters

Example:

```
counters
```

Counters-circuit-class M

Enter at the `BRS [i #]>` prompt. It displays statistics describing bandwidth reservation traffic for the circuit classes of the selected Frame Relay interface.

Syntax: counters-circuit-class

Example:

```
counters-circuit-class
Bandwidth Reservation Circuit Class Counters
Interface 0
Class         Pkt Xmit     Bytes Xmit     Bytes Ovfl

DEFAULT         103          57692             0
new            2149        1730056             0
CLASS 2           0              0             0

TOTAL          2252        1787748             0
```

6.3 Bandwidth Reservation Configuration and Console Commands

Deassign C

Restores a specified protocol, tag, or filter to its default class and priority.

Syntax: deassign *protocol* **or** *tag* **or** *filter*

Example:

```
deassign IP
```

Deassign-circuit C

Use this command at the interface level to deassign the specified circuit (DLCI) from the circuit class to which it was previously assigned.

Syntax: deassign-circuit *permanent-virtual-circuit#*

Example:

```
deassign 16
```

Default-circuit-class C

Use this command at the interface level to select the name of the default circuit class.

Syntax: default-circuit-class *class-name*

Example:

```
default-circuit-class group
```

Default-class C

Sets the default class and priority to a desired value. If no value was previously assigned, system default values are used. Otherwise, the last previously assigned value is used.

Syntax: default-class *class-name* **or** *class#* *priority*

Example:

```
default-class test normal
```

6.3 Bandwidth Reservation Configuration and Console Commands

Del-circuit-class C

Use this command at the interface level to delete the specified bandwidth class.

Syntax: del-circuit-class *class-name*

Example:

```
del-circuit-class group
```

Del-class C

Deletes a previously configured bandwidth class from the specified interface or Frame Relay circuit.

Syntax: del-class *class-name* **or** *class#*

Example:

```
del-class ip
```

Disable C

Disables bandwidth reservation on the interface or Frame Relay circuit. This command requires a router restart.

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

Syntax: disable

Example:

```
disable
```

Enable C

Enables bandwidth reservation on the interface or Frame Relay circuit. This command requires a router restart.

Syntax: enable

Example:

```
enable
```

6.3 Bandwidth Reservation Configuration and Console Commands

Interface C M

Selects the serial interface to which bandwidth reservation configuration commands are to be applied. Bandwidth reservation is supported on routers running the Proteon Serial Line protocol, PPP, Frame Relay, V.25 *bis*, and ISDN interfaces.

Note: To enter bandwidth reservation commands for a new interface, you must enter this command BEFORE using any other bandwidth reservation configuration commands. If you have exited the bandwidth reservation prompt and want to return to make bandwidth reservation changes to a previously configured interface, you must again enter this command first.

Interface C

To configure 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 BRS configuration commands as described in this chapter.

Syntax: interface *interface#*

Example:

```
interface 2
```

Interface M

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

Syntax: interface *interface#*

Example:

```
interface 0
```

6.3 Bandwidth Reservation Configuration and Console Commands

Last M

Displays the last saved bandwidth reservation statistics. The statistics are displayed in the same format as they are for the **counters** command.

Syntax: last

Example:

```
last
```

Last-circuit-class M

Enter the **last-circuit-class** command at the `BRS [i #]>` prompt. It displays the last saved bandwidth reservation statistics for the circuit classes of the selected Frame Relay interface. The statistics are displayed in the same format as they are for the **counters-circuit-class** command.

Syntax: last-circuit-class

Example:

```
last-circuit-class
```

List C

Displays currently defined bandwidth classes by their guaranteed percentage rates and priority queuing values stored in SRAM. This command also displays all assigned protocols and filters.

Syntax: list

Example:

```
list
```

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

```
BRS Config>
BRS [i 1] Config>           (for PPP interface 1)
BRS [i 0] Config>           (for FR interface 0)
BRS [i 0] [dlci 16] Config> (for circuit 16 on FR interface 0)
```

For example, the following output appears when you issue the **list** command at the `BRS Config>` prompt:

6.3 Bandwidth Reservation Configuration and Console Commands

Example:

```
BRS Config> list
Bandwidth Reservation is available for 2 interfaces.
Interface      Type           State
---------      ----           -----
       0       FR             Enabled
       1       PPP            Enabled
```

6.3 Bandwidth Reservation Configuration and Console Commands

The **list** command is very similar to the **show** command. However, **show** displays current settings from the active RAM.

Note: For Frame Relay, there are two levels of this command: the interface level and the circuit level.

Show C

Displays currently defined bandwidth classes stored in RAM.

Syntax: show

Example:

```
show
```

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 1] Config>             (for PPP interface 1)
BRS [i 0] Config>             (for FR interface 0)
BRS [i 0] [dlci 16] Config>   (for circuit 16 on FR interface 0)
```

Tag C

Assigns a class and priority to a filter that was tagged during the configuration of the MAC filtering feature. The command requires a filter tag number (configured in MAC filtering) to reference the tag in bandwidth reservation. Refer to the Using MAC Filtering chapter in the *Bridging Configuration Guide.*

Up to five tagged MAC addresses can be set from 1 to 5. Tag 1 is searched for first, then tag 2, and so on up to tag 5.

Any newly added address filter can then be assigned a tag (as any other protocol or filter) with the **assign** command. See the **assign** command in this chapter for more information.

Syntax: tag *tag#*

Example:

```
tag 3
```

Untag C

Removes the tag/tag name relationship and the tag name from the list of assignable filters.

A tag can only be removed if it is not assigned to any class.

Syntax: untag *tag#*

Example:

```
untag 3
```

Exit C M

Use the **exit** command to do the following:

- Return from the circuit level to the interface level.
- Return from the interface level to the `BRS Config>` level.
- Return from the `BRS Config>` level to the `Config>` level.

Syntax: exit

Example:

```
exit
```

7

Configuring and Monitoring DNA IV

The DIGITAL Network Architecture Phase IV (DNA IV) protocol runs over Token Ring (TR), Frame Relay, Ethernet, PPP, and X.25 interfaces. This chapter describes how to configure and monitor DNA IV using the Network Control Program (NCP) and how to use the NCP configuration and console commands.

Note: When operating DNA IV networks with DNA V networks, all DNA IV monitoring must be done from the process described in this chapter. For more information about DNA IV and DNA V compatibility, refer to the *Routing Protocols Reference Guide*.

NCP is the user interface for this router's implementation of DECnet Phase IV. This NCP supports a limited subset of the DECnet VAX NCP commands.

For more information about DNA and NCP, refer to the *Routing Protocols Reference Guide*.

7.1 Accessing the NCP Environment

Both NCP configuration and console commands can be accessed from either the CONFIG (configuration) or GWCON (console) environments. For information about accessing the NCP environment, see Chapter 1.

Note: NCP is not available from the GWCON (console) environment unless DNA IV has been configured from the CONFIG environment and is running (state is **on)**.

7.1.1 NCP Command Syntax

The command syntax has three parts: a command, a component, and an argument.

A command indicates the action to perform. A component indicates the subsystem to which the command applies. An argument is either an attribute and its value, or a keyword representing a group of attributes.

Table 7–1 shows several commands, their components, and their arguments.

7.1 Accessing the NCP Environment

Table 7–1 Example NCP Commands

Command	Component	Argument
set	circuit eth/0	router priority 100
define	all circuits	state off
show	executor	characteristics

7.1.2 Configuring DNA IV for Token Ring

The procedure to run DNA IV over 802.5 Token Ring (TR) involves commands from the DNA IV and Token Ring configuration processes.

Note: If you are using an IBM 8209 MAC layer bridge and a RouteAbout or DECswitch router on the same Token Ring network, you must start the router before starting the IBM 8209.

1. From the OPCON prompt (*), enter the configuration process.

```
* talk 6
Config>
```

2. Enter **list device** to see the interface numbers for the Token Ring interfaces. The devices named DEC Token-Ring are Token Ring interfaces. Note the number of each Token Ring interface.

```
Config> list device
```

3. Use the **network** command with the interface number of the Token Ring interface you want to configure. This places you in the Token Ring configuration process.

```
Config> network 1
TKR config>
```

4. Enter **list** to verify the Token Ring configuration.

```
TKR config> list
Token-Ring configuration:
Packet size (INFO field):    2052
Speed:                       4 Mb/sec
Media:                       Shielded
RIF Aging Timer:             120
Source Routing:              Enabled
Mac Address   000000000000
```

5. Exit the Token Ring configuration process and enter the DNA NCP configuration process.

```
TKR config> exit
Config> protocol DN
NCP>
```

6. Use the **show** command to verify that each Token Ring circuit is working.

```
NCP> show circuits tkr/1 characteristics

Circuit Volatile Characteristics

Circuit=TKR/1

State                         = ON
Designated Router             = 62.412
Cost                          = 4
Router priority               = 10
Hello timer   = 15
Maximum routers               = 16
Routing type                  = Standard
Adjacent node                 = 62.590
Listen time                   = 90
```

7. Check the routing type field in the Circuit Volatile characteristics display. For bilingual or Phase IV AMA support, you need to change the routing type from the default (standard) to either AMA or bilingual. For example:

```
NCP> define circuit tkr/1 router type AMA
```

or

```
NCP> define circuit tkr/1 router type bilingual
```

8. Exit NCP.

```
NCP> exit
Config>
```

9. Use the **update version** command only when updating to a new release of software.

Note: To disable source routing or set the RIF timer to a value other than the default value, use the **source-routing** command and the **set RIF-timer** command in the Token Ring configuration process.

7.1 Accessing the NCP Environment

7.1.3 Configuring DNA IV for X.25

The procedure to run DNA IV over X.25 circuits involves commands from the X.25 and DNA IV configuration processes.

7.1.3.1 Configuring the X.25 Data Link

1. From the OPCON prompt (*), enter the configuration process.

```
 * talk 6
 Config>
```

2. Enter **list device** to see the interface numbers for the serial interfaces.

```
Config> list device
```

3. Set the data-link protocol for a serial interface to X.25.

```
Config> set data-link X.25
```

4. Use the **network** command with the interface number of the serial interface you want to configure. This places you in the X.25 configuration process.

```
 Config> network 1
 X.25 User Configuration
 X.25 Config>
```

5. If this is a first-time configuration of X.25, refer to Chapter 12, Configuring the X.25 Network Interface, in the *Network Interface Operations Guide* for configuration details.
6. Restart the router so that the configured parameters take effect.

7.1.3.2 Configuring DECnet Phase IV

1. Go to the `NCP>` prompt to set DNA IV parameters.

```
 Config> protocol dn
```

2. Configure DECnet executor parameters by doing the following:

   – Define the router’s DECnet address.

```
 NCP> define exec address area.node
```

– Configure the router as a level 1 or level 2 DEC-type router.

```
NCP> define exec type dec-routing-iv or dec-area
NCP> define exec state on
```

– Restart the router so that when you configure the X.25 circuit, all DEC-specific parameters are visible.

3. Verify the executor configuration.

```
NCP> show executor characteristics
```

4. Define DECnet Phase IV X.25 circuits.

   You must configure the X.25 circuit as an SVC. If you configure this circuit as an IN-SVC, you must configure the other end of the connection as an OUT-SVC.

```
NCP> define cir x25/0 usage in-svc
NCP> define cir x25/0 number remote-X.25-DTE
NCP> define cir x25/0 call-data
```

5. Define circuits as active.

```
NCP> define circuit x25/0 state on
```

6. Restart the router so that the DECnet parameters take effect.
7. Verify the X.25 configuration within the DECnet protocol.

```
NCP> list circuit x25/0 characteristics
```

7.1.4 Configuring DNA IV in an OSI/DNA V Environment

Configuring DNA IV on a router that will run OSI/DNA V requires that the DNA V NSAP address be compatible with the DNA IV address. You are advised to disable both protocols and restart the router before configuring the DNA IV executor address and the DNA IV-compatible OSI/DNA V NET. See Chapter 11 for details of the configuration procedure.

7.2 NCP Configuration and Console Commands

If OSI/DNA V is enabled and has a DNA IV compatible NET, the system will derive the executor address from the NET when you use the **define executor address** command at the `NCP>` prompt. For example, if OSI routing is enabled and the NET translates to DNA IV address 13.3, you could change the executor address to 13.1 as follows:

```
NCP>define executor address
Address [13.3]? 13.1
Defaulting NET to Phase IV compatible [49000D:AA0004000134]
```

The **define executor type command** determines whether the router is a level 1 or level 2 (area) router, and if it needs to be compatible with other DNA routers or not. The choice of executor type affects the OSI prefix address for routing circuits. There is an extended format for the prefix address which has additional fields needed for DEC-mode routers and a shorter format compatible with earlier versions of the software.

7.2 NCP Configuration and Console Commands

This section explains the NCP configuration and console commands. Enter the commands at the `NCP>` prompt.

Table 1-2 summarizes the NCP commands. Note that all commands are available for both configuring and monitoring environments.

Table 7–2 NCP Configuration and Console Commands Summary

Command	Function
? (Help)	Lists all the NCP commands or lists the options associated with a specific command.
Define	Sets or modifies configuration information in the permanent database.
Purge	Removes configuration information from the permanent database.
List	Displays configuration information in the permanent database.
Set	Sets or modifies information in the volatile database.
Show	Displays information in the volatile database.
Zero	Clears counters in the volatile database.
Exit	Exits NCP.

7.2 NCP Configuration and Console Commands

The **define**, **purge**, and **list** commands act on the configuration information stored in the router. This configuration information is referred to as the permanent database, and survives restarts, software loads, and power cycles. The router uses it each time it starts.

The **set**, **show**, and **zero** commands act on the information currently used by the running router. This information is referred to as the volatile database. It is initialized at startup from the permanent database but may change due to console commands or normal operation of the DNA IV protocol. Changes to the volatile database do not remain in effect when the router is restarted.

? (Help) C M

Lists the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example:

```
?
DEFINE
LIST
PURGE
SET
SHOW
ZERO
EXIT
```

Define C M

Defines access control lists and routing filters. The *access control* and *routing filter* modules are defined in the permanent database and cannot be amended in the volatile database, so there is no equivalent **set** command for the **define module** commands. See the **set/define** command for details of these two options.

Syntax: define

module access-control . .
module routing-filter . . .

7.2 NCP Configuration and Console Commands

Set/Define C M

Use the **set** command to set or modify information in the volatile database. Use the **define** command to set or modify configuration information in the permanent database. The command syntax for **set** and **define** is identical, except as noted below:

Syntax: set

circuit-specifier . . .
executor . . .
node . . .

Syntax: define

circuit-specifier . . .
executor . . .
module . . .
node . . .

circuit-specifier *argument*

Sets or changes circuit arguments in the volatile database of DNA when the **set c**ommand is used. The **define** command sets or changes circuit arguments in the permanent database. The circuits must be in the off state to modify numeric arguments in the volatile database.

The *circuit-specifier* options include the following:

active circuits Specifies all circuits who are up and whose state is on (**set** only).

all circuits Specifies all circuits on the router.

circuit [*name*] Specifies the named (for example, Eth/0, TKR/0) circuit.

known circuits Specifies all circuits on the router.

The arguments include the following:

call-data Used during initialization of X.25 circuits. When a circuit is defined as an outgoing SVC, the initial and all subsequent call requests contain the defined call-data when the circuit is enabled. When a circuit is defined as an incoming SVC, one of the criteria for accepting an incoming call request is a match of the defined call data.

Call-data accepts an even number of hexadecimal characters (octets) up to a maximum of 14 characters. (A hexadecimal octet is equivalent to two ASCII characters. For example, the ASCII sequence FF is equivalent to 1 octet that is interpreted as 1111 1111.)

cost [*number*] Sets the cost to receive a packet on this circuit. This is used by the routing algorithm to determine the cost of a circuit in choosing routes (cost is not the same as an IP metric). Range: 1 to 25. Default: 4.

The following values are suggested starting points:

Circuit Type	**Cost**
Ethernet	4
Token Ring	4
Sync 56 Kb	6
Sync T1	5
X.25	25
FDDI	1

number Specifies a DTE address for both incoming and outgoing SVCs on an X.25 circuit. This is always the address of the remote system. Enter a decimal number of up to 15 digits.

hello timer [*range*] Specifies how often (in seconds) router hellos are sent on this circuit. Range: 1 to 8191 seconds. Default: 15 seconds (recommended).

maximum recalls Specifies how many attempts the router makes to reestablish an outgoing static SVC call after an initial call failure. After *maximum recalls*, the router makes no further attempts to establish the SVC without your intervention. The range is 0 to 255. The default is 10. See also the *recall timer* argument.

maximum routers [*range*]

Specifies how many other routers there are on this circuit. Range*:* 1 to 33. Default: 16. Valid only with **define**.

Note: You cannot configure this parameter on an X.25 circuit when the executor *type* is set to DEC-routing-IV or DEC-area**.** In this case, the maximum number of routers is 1.

If this is a level 1 router, only routers on this circuit in the same area count. If this is a level 2 router, also count all level 2 routers on this circuit. The local router does not count against the limit.

Do not set this argument to less than the actual number of routers on the circuit. This can result in anomalies in routing.

Note: For a point-to-point (synchronous line) circuit, set this argument to 1. The result is *significant* memory savings on a router with multiple point-to-point lines.

recall timer

Determines the delay in seconds between call request attempts to establish an X.25 outgoing static SVC. Specify a value in the range 0 to 65595. The default is 60 seconds. See also the argument *maximum recalls*.

router priority [*number*]

Specifies the router’s priority in bidding to become the designated router for the end nodes on this circuit. Range: 1 to 127, where 127 is the highest priority. Default: 64.

If two routers have the same priority, the one with the higher node address wins. The router priority has no effect on area routing decisions, or in reaching the closest “attached” level 2 router.

Note: You must *not* use this command to configure a circuit router priority level on a Phase IV circuit which also has OSI configured or you will get this warning message:

Example:

```
NCP>define cir eth/0 router priority
Modify circuit router priority from OSI
NCP>
```

If the circuit is configured for Phase V/OSI and Phase IV, use the `OSI config>` **set subnet** command to set the router priority. (See Chapter 11 for more details about this command.)

router type standard
Specifies that the router is using conventional Phase IV addressing where the MAC address is built from the area and node number. The router defaults to this type.

router type ama
Specifies that the router can route packets that use phase IV addressing where the MAC address is arbitrary and learned from the data link layer. Valid only on Token Ring circuits.

router type bilingual
Specifies that the router can route packets that use both conventional and Phase IV with AMA addressing. Valid only on Token Ring circuits.

state on
Specifies that the circuit is enabled for use by DNA.

state off
Specifies that the circuit is disabled for use by DNA. This is the default.

usage
Specifies whether an X.25 circuit is:

IN-SVC – An incoming switched virtual circuit.

OUT-SVC – An outgoing switched virtual circuit.

This parameter applies when the executor type is set to *DEC-routing-IV* or *DEC-area.*

(See **set/define executor type**.)

Example:

```
set circuit eth/0 cost 4
```

Example:

```
define circuit eth/0 cost 4
```

Note: If the router is configured as a DEC-mode executor type and OSI is enabled, you must configure X.25 circuits from the OSI environment. If you attempt to configure and enable an X.25 circuit when OSI is enabled you will get an error message.

Example:

```
define circuit x25/0 state on
Can’t enable DNAIV X25 circuit while OSI is enabled
```

7.2 NCP Configuration and Console Commands

executor *argument*

Sets or modifies arguments global to DNA in the volatile database when the **set c**ommand is used. The **define** command does the same in the permanent database.

The executor must be in the off state to modify numeric arguments or the type in the volatile database.

Example:

```
set executor state on
set executor maximum broadcast routers 10
```

Example:

```
define executor state on
define executor maximum broadcast routers 10
```

address [*area.node*]

The DNA IV address of this router. Area range: 1 to 63. Node range: 1 to 1023. Area must not be greater than executor maximum area, and node must not be greater than executor maximum address.

If OSI is enabled and has a DNA IV compatible NET, then the equivalent DNA IV address will override any value entered.

The default 0.0 is illegal.

Note: DNA will not go to its **on** state if the executor address is not set to a legal value.

area maximum cost [*number*]

Maximum cost that is allowed between this level 2 router and any other area. If the best route to an area is more expensive than this, that area is considered unreachable. Maximum: 1022. Default: 1022. This argument does not apply to level 1 routers. Make this value greater than the maximum legal cost to the most distant area. A suggested value is 25 times **area maximum hops.**

area maximum hops [*number*]

Maximum number of hops allowed between this level 2 router and any other area. If the best route to an area requires more than the maximum number of hops, that area is considered unreachable. Maximum: 30. Default: 30. This argument does not apply to level 1 routers. Make this value about twice the longest path length (in hops) that is expected.

The hop count is used by routing only to speed the decay of routes to unreachable areas. This argument may be reduced to cause unreachable areas to become unreachable more quickly.

broadcast routing timer [*number*]

Specifies how often level 1 (and 2 in a level 2 router) routing messages are sent (in seconds). This is how often they are sent in the absence of any cost or adjacency changes. This protects the routing database from corruption. Routing updates are sent immediately if any cost or adjacency changes. Range: 1 to 65535. Default: 180. Lower values increase the overhead for this and all adjacent routers. Higher values increase the time required to correct the routing database if a routing update message is lost.

maximum address [*number*]

The highest node address within the area to which routes are kept. The routing database does not include routes to any nodes in the area with a higher node address. Range: 1 to 1023. Default: 1023. Valid only with **define**.

Make this value greater than or equal to the highest node address in the router's area.

maximum area [*number*]

The highest area to which routes are kept, if this is a level 2 router. The routing database does not include routes to any higher numbered areas. Range: 1 to 63. Default: 63. Valid only with **define**.

Make this value greater than or equal to the highest area number in the network.

maximum broadcast nonrouters [*number*]

Maximum number of end nodes that can be adjacent to (one hop away from) this router on broadcast circuits. Range: 1 to 1023. Default: 64. Valid only with **define**.

Make this value greater than or equal to the total number of end nodes on all broadcast circuits. If this value is too small, some end nodes will not be reachable by this router, causing unpredictable routing problems.

maximum broadcast routers [*number*]

Maximum number of routers that can be adjacent to (one hop away from) this router on broadcast circuits. Range: 1 to 33 times the number of broadcast circuits. Default: 32. Valid only with **define**.

Make this value greater than or equal to the total number of routers on all broadcast circuits. If this value is too small, routes will not be accepted from some routers, causing unpredictable routing problems.

maximum cost [*number*]
Maximum cost that is allowed between this router and any other node in the area. If the best route to a node is more expensive than this, that node is considered unreachable. Maximum: 1022. Default: 1022. A suggested value is 25 times *maximum hops*.

maximum hops [*number*]
Maximum number of hops that are allowed between this router and any node in the area. If the best route to a node requires more than the maximum number of hops, that node is considered unreachable. Maximum: 30. Default: 30. It is about twice the longest path length (in hops) that is expected. The hop count is used by routing only to speed the decay of routes to unreachable nodes. This argument may be reduced to cause unreachable nodes to become unreachable more quickly.

maximum visits [*number*]
Maximum number of times a packet can be forwarded. The router will drop any packet it receives that has already been forwarded maximum visits times. Range: 1 to 63. Default: 63.

This is used to detect packets that are in routing loops, which can occur temporarily as routes change. It should be set to at least twice the value of *maximum hops* or *area maximum hops*, whichever is larger.

state on
Enables DNA IV. Not valid with **set** if the router was started in *state off* or without a valid address, or if DNA IV initialization failed due to insufficient memory.

state off
Disables DNA IV. The default state is *off*.

type	Defines whether the router is a level 1 router or level 2 (area level) router. On X.25 circuits, this parameter defines whether the router acts as a DEC-mode router, or not. In DEC-mode, the router is compatible with other DNA routers such as DECNIS and DEMSA Enter one of the following: **DEC-routing-iv** – For a DIGITAL-compatible level 1 router. This is the default. **DEC-area** – For a DIGITAL-compatible level 2 area router. **Routing-iv** – For a level 1 router without DIGITAL compatibility on X.25 circuits. **Area –** For a level 2 (area) router without DIGITAL compatibility on X.25 circuits. A level 2 router accepts adjacencies with routers in other areas, and maintains routes to all areas. If it can reach other areas, it also advertises itself to level 1 routers as a route to other areas. For level 1 routers, adjacencies are accepted only to routers in the same area. **Note:** When a router is configured as a DEC-mode router and OSI is enabled, all X.25 circuits **must** be configured from within the OSI environment.

module access-control *circuit-specifier argument*

Defines access control lists, which are used to restrict the forwarding of packets between certain origins and destinations. Each access list is associated with one circuit, and applies to DECnet Long Format Data Packets received on that circuit. Access control does not apply to any routing or hello packets. Valid only with **define**.

The arguments for the circuit-specifiers include the following:

all circuits	Specifies all circuits on the router.
circuit name	Specifies the named circuit.
known circuits	Specifies all known circuits on the router.

7.2 NCP Configuration and Console Commands

The following items are the arguments you select from after you enter the **define module access-control** command and the circuit-specifier:

state on — Enables the access control list on this circuit.

state off — Disables the access control list on this circuit.

type exclusive — Specifies that any packets matching one or more of the filters in the access control list for this interface are dropped.

type inclusive — Specifies that only packets matching one or more of the filters in the access control list for this interface are forwarded.

filter
[*source-result*
source-mask
dest-result
dest-mask] — Adds a filter to the list for the specified circuit. The filter is added to the end of the existing list.
The source address is masked with the source-mask, and compared to the source result. The same is done with the *dest-mask* and *dest-result*. The action depends on what type of access control is in use on the circuit.

The following are the options you may select after you enter the **define module access-control** command and the *filter* circuit-specifier:

source-result — Address that the source address is compared to after masking.

source-mask — Mask used for the source address.

dest-result — Address that the destination address is compared to after masking.

dest-mask — Mask used for the destination address.

Example:

```
define module access-control circuit eth/0 state on
```

module routing-filter ***circuit-specifier argument***

Defines routing filters, which are used to restrict the sending of Area routes by level 2 (Executor Type Area) routers. Valid only with **define**.

all circuits Specifies all circuits on the router.

circuit name Specifies the named circuit.

known circuits Specifies all circuits on the router.

The following are the direction options you may select after you enter the **define module routing-filter** command and the circuit-specifier:

incoming Affects the filter on routing information received on this circuit.

outgoing Affects the filter on routing information sent on this circuit.

The following are the arguments you may select after you enter the **define module routing-filter** command and the circuit-specifier:

area [*area-list*] Specifies that the filter allows routing information to pass for the set of areas in the *area-list*. The *area-list* is a comma-separated list of areas or ranges of areas. A range is specified by two area numbers separated by a dash. The value for *area-list* can also be *none*, specifying that information is passed for no areas. The following are *area-list* examples:

1,4,9,60 Areas 1, 4, 9, and 60
1-7,9-13,23 Areas 1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 12, 13, and 23

state on Specifies that the filter is active.

state off Specifies that the filter is disabled, but continues to be stored in the permanent database. The only way to remove the filter is by using the **purge** command.

Example:

```
define module routing-filter circuit eth/0 state on
```

7.2 NCP Configuration and Console Commands

node *address argument*

Identical to executor when used with the router’s address. No other addresses are valid. See the **set/define executor** command description for more information.

Show/List C M

Use the **show** command to display information in the volatile database. Use the **list** command to display configuration information in the permanent database. The command syntax for **show** and **list** is identical, except as noted below.

Syntax: show

area-specifier . . .
circuit-specifier . . .
executor . . .
module . . .
node-specifier . . .

Syntax: list

circuit-specifier . . .
executor . . .
module . . .
node-specifier . . .

area-specifier *argument*

Examines the status of the volatile area routing database. This lets you find out what areas are reachable, and what the routes are to various areas. Valid only with **show**.

The options for the area-specifiers include the following:

active areas Provides information about those areas that are currently reachable.

all areas Provides information about all areas (up to the executor maximum area).

area [*area*] Provides information about the specified area. If the area is not provided, you are prompted for it.

known areas Provides information about those areas that are currently reachable.

The arguments are the following:

characteristics Shows the current state of the specified area. (The same as **summary**.)

status Provides detailed information about the specified areas, including cost and hops.

summary Shows the current state of the specified areas. This is the default.

The following area items are displayed by these commands:

area Indicates the area for this line of the display.

circuit Indicates which circuit the next hop to this node goes over. No circuit is given for the router's own area.

cost Indicates the cost to this area.

hops Indicates the hops to this area.

next node Indicates the router that is the next hop (intermediate destination) to the specified area.

If the circuit is a static OSI routing circuit with a DNA IV compatible prefix address, then the Node name is not available, and the value `(Static)` is displayed.

state Indicates that this is reachable or unreachable.

Example:

```
show active areas
Active Area Volatile Summary

Area State       Circuit Next
                         Node
1    reachable    Eth/0  1.22
2    reachable           2.26
56   reachable    Eth/1  (Static)
63   reachable    X25/0  2.30
```

Example:

```
show active areas status
Active Area Volatile Status
Area State       Cost Hops Circuit Next
                                   Node
1    reachable   3    1     Eth/0  1.22
2    reachable   0    0            2.26
56   reachable   20   1     Eth/1  (Static)
63   reachable   11   1     X25/0  2.30
```

circuit-specifier *argument*

Use the **show circuit-specifier** command to retrieve information about the current state of the specified circuits from the volatile database. The **list circuit** command retrieves the data that is stored in the permanent database for circuits.

The circuit-specifier options are the following:

active circuits Specifies all circuits that are currently on. Valid only with **show**.

all circuits Specifies all circuits on the router.

circuit [*name*] Specifies the named circuit.

known circuits Specifies all circuits on the router. Valid only with **show**.

The arguments are the following:

characteristics Provides detailed information about all of the argument settings for the circuit.

counters Shows counters for the circuit. Valid only with **show**.

s*tatus* Shows detailed information about the circuit from the volatile database.

s*ummary* Shows summary information about the circuit from the volatile database. This is the default if no argument is supplied.

The following circuit items are displayed by these commands:

adjacent node Node ID of a node that has an adjacency with this node on the circuit being displayed. While adjacencies with end nodes automatically make that node reachable, a router adjacency does not automatically make that node reachable. A router is not considered reachable unless a routing message was received over an active adjacency from that router. Nodes may be shown as adjacent in the circuit database, but are not in the reachable nodes database (**show active nodes**).

block size Maximum data block size that the associated adjacent node is willing to receive. This is typically 1498 bytes, which is the standard 1500 bytes of an Ethernet packet, less the 2-byte length field used with DECnet.

circuit Circuits to which this data applies.

designated router Displays what this node believes to be the designated router for this area on this circuit. (There may be some temporary disagreements when a new router starts up.) This normally is the same for all routers on the circuit. Endnodes send all packets for destinations not on the local circuit to their designated router.

hello timer Hello timer for this circuit. Router hello messages are sent this often on the circuit.

listen timer Amount of time designating how often router or endnode hellos must be received from this adjacency on this circuit. It is three times the hello timer set for this circuit on the adjacent machine.

router priority Router priority for this circuit, used in vying for designated router status.

router type Router type for this circuit: Standard, Phase IV with AMA, or Bilingual.

maximum routers Maximum number of routers allowed on this circuit.

state

Either ON or OFF. In the volatile database, the state is ON if the circuit is enabled and is passing self-test. If the circuit has failed self-test or the device is not present, the state is OFF.

In the permanent database, this tells whether DNA tries to enable the circuit.

Example:

```
show all circuits
Circuit Volatile Summary
Circuit State        Adjacent
                       Node
 X25/0    on            5.25
 Eth/0    on            1.22
 Eth/0                  2.14
 Eth/0                  1.13
 Eth/1    off
```

Example:

```
list circuit eth/0 characteristics
Circuit Permanent Characteristics
Circuit            = Eth /0
State              = On
Cost               = 4
Router priority    = 64
Hello timer        = 15
Maximum routers    = 16
Router type    = Standard
```

Example:

```
show active circuits status
Active Circuit Volatile Status
Circuit State        Adjacent   Block
                       Node     Size
 Eth/0   on            1.22     1498
 Eth/0                 2.14     1498
 Eth/0                 1.13     1498
 X25/0   on            5.25     1498
```

The next example shows the current characteristics of the circuits on this machine. This includes all of the configuration arguments, as well as the current adjacencies, and the listen timer (three times the adjacency’s hello timer).

Example:

show all circuits characteristics

```
Circuit Volatile Characteristics
Circuit            = Eth/0
State              = on
Designated router  = 2.26
Cost               = 4
Router priority    = 64
Hello timer        = 15
Maximum routers    = 16
Adjacent node      = 1.22
  Listen timer     = 45
Adjacent node      = 2.14
  Listen timer     = 45
Adjacent node      = 2.39
  Listen timer     = 90
```

The next example shows the counters that are kept for the circuits. Note that some counters kept by DECnet VAX are not kept here, but are instead read through the **network** command of GWCON.

Example:

show circuit eth/0 counters

```
Circuit Volatile Counters
Circuit = Eth/0
    525249  Seconds since last zeroed
         0  Terminating packets received
         0  Originating packets sent
      3693  Transit packets received
      4723  Transit packets sent
         0  Transit congestion loss
         0  Circuit down
         0  Initialization failure
         0  Packet corruption loss
```

executor *argument*

Retrieves information about the current state of the volatile database for DNA with the **show executor** command. The **list executor** command retrieves the data that is stored in the permanent database for DNA.

7.2 NCP Configuration and Console Commands

The arguments are the following:

characteristics Gives the detailed information about the settings of all of the adjustable arguments of the routing database.

counters Gives the global event and error counters for DNA. Valid only with **show**.

status Gives key information about the state of DNA.

summary Gives a brief summary on the state of DNA. This is the default.

The following executor items are displayed by these commands:

area maximum cost Specifies the maximum allowed cost to an area.

area maximum hops Specifies the maximum allowed hops to an area.

broadcast routing timer Specifies the frequency of sending routing messages in the absence of any changes.

buffer size Specifies the buffer size for the router.

executor node Specifies the node address and node name. The node name is the name set by the CONFIG **set hostname** command.

identification Specifies the the identification of the router software, as sent in MOP System ID messages.

maximum address Specifies the highest node number in the router's area to which routes are kept.

maximum area Specifies the highest area to which routes are kept.

maximum broadcast nonrouters Specifies the maximum number of end nodes adjacent to this router.

maximum broadcast routers Specifies the maximum number of routers adjacent to this router.

maximum buffers Specifies the number of packet buffers in the router.

maximum cost Specifies the maximum allowed cost to a node in the router's area.

maximum hops — Specifies the maximum allowed hops to a node in the router’s area.

maximum visits — Specifies the maximum number of routers through which a packet may be routed between source and destination.

physical address — Specifies the physical Ethernet address set on all Ethernet circuits when DNA starts. Derived from the node address.

routing version — Version is always Version 2.0.0.

state — The state of DNA, on or off.

type — Specifies the routing level. The value is one of the following:

DEC-routing-iv – For a DIGITAL-compatible level 1 router. This is the default.

DEC-area – For a DIGITAL-compatible level 2 (area) router.

Routing-iv – For a level 1 router without DIGITAL compatibility on X.25 circuits.

Area – For a level 2 (area) router without DIGITAL compatibility on X.25 circuits.

Example:

```
show executor
Node Volatile Summary
Executor node           = 2.26 (gato)
State                   = on
Identification          = DECnet-MC68020 V9.0
```

7.2 NCP Configuration and Console Commands

Example:

```
show executor characteristics
Node Volatile Characteristics
Executor node            = 2.26 (gato)
State                    = on
Identification           = DECnet-MC68020 V9.0
Physical address         = AA-00-04-00-1A-08
Type                     = DEC-area
Routing version          = V2.0.0
Broadcast routing timer  = 180
Maximum address          = 64
Maximum cost             = 1022
Maximum hops             = 30
Maximum visits           = 63
Maximum area             = 63
Max broadcast nonrouters = 64
Max broadcast routers    = 32
Area maximum cost        = 1022
Area maximum hops        = 30
Maximum buffers          = 103
Buffer size              = 2038
```

Example:

```
list executor status
Node Permanent Status
Executor node            = 2.26 (gato)
State                    = on
Type                     = DEC-area
```

Example:

```
show executor counters
Node Volatile Counters
Executor node            = 2.26 (gato)
525948  Seconds since last zeroed
     0  Aged packet loss
     0  Node unreachable packet loss
     0  Node out-of-range packet loss
     0  Oversized packet loss
     0  Packet format error
     0  Partial routing update loss
     0  Verification reject
```

module access-control circuit-specifier *argument*

The **show module access-control** command displays access control list information from the volatile database. The **list module access-control** command displays access control list configuration information from the permanent database.

The options for the circuit-specifiers include the following:

all circuits Specifies all circuits on the router.

circuit [*name*] Specifies the named circuit.

known circuits Specifies all known circuits on the router.

The arguments are the following:

counters Gives counters on the use of the access control lists. Valid only with **show**.

status Shows detailed information about the access control lists, including the filters in the access control list.

summary Shows summary information about the state of the access control lists. This is the default.

Example:

```
show module access-control circuit eth/0 counters
Module Access-Control Volatile Counters

Circuit = Eth /0

6337Seconds since last zeroed
   0Packets processed
   0Packets rejected
   0Access control loop iterations
```

module routing-filter circuit-specifier *argument*

The **show module routing-filter** command displays area routing filter information from the volatile database. The **list module routing-filter** command displays area routing filter configuration information from the permanent database.

7.2 NCP Configuration and Console Commands

The options for the circuit-specifiers include the following:

all circuits — Specifies all circuits on the router.

circuit [*name*] — Specifies the named circuit.

known circuits — Specifies all known circuits on the router.

The arguments are the following:

characteristics — Shows detailed information about the routing filters, including the area list.

status — Shows detailed information about the routing filters, including the area list.

summary — Shows summary information about the state of the routing filters. This is the default.

Example:

```
show module routing-filter circuit eth/0 status
```

Example:

```
list module routing-filter circuit eth/0 status
```

node-specifier argument

The **show node** command displays the contents of the volatile level 1 routing database. This indicates which nodes in the area are reachable and the next hop to each.

The permanent database only contains node information for the router itself. The **list node [executor-address]** command is equivalent to the **list executor** command.

The node-specifiers can be any of the following:

active nodes — Provides information about all nodes that are currently reachable. Valid only with **show**.

adjacent nodes — Provides information about nodes that are adjacent to (one hop away from) the router.

all nodes — Provides information about all nodes in the area (up to the *executor maximum address*). Valid only with **show**.

node [*node*] Provides information about the specified node. If the node is not provided, you are prompted. Nodes other than the router itself are valid only with **show**.

known nodes Provides information about those nodes that are currently reachable. Valid only with **show**.

The arguments include the following:

characteristics Shows the current state of the specified nodes, including type and specified nodes.

status Provides detailed information about the specified nodes, including type, cost, and hops.

summary Shows the current state of the specified nodes. This is the default.

This example shows the reachable nodes.

Example:

```
show active nodes
Active Node Volatile Summary
Executor node            = 2.26 (gato)
State                    = on
Identification           = DECnet-MC68020 V9.0

Node    State       Circuit Next
Address                     Node
 2.14   reachable    Eth /0  2.14
 1.22   reachable    Eth /0  1.22
```

The next example shows the detailed routing information about all adjacent nodes. Only nodes with one hop are shown. Note that the node type is only known and displayed for adjacent nodes, as this information is only contained in hello messages.

Example:

```
show adjacent nodes status
Adjacent Node Volatile Status

Executor node            = 2.26 (gato)
State                    = on
Physical address         = AA-00-04-00-1A-08
Type                     = DEC-area
Node    State       Type       Cost  Hops Circuit  Next
Addr                                               Node
 2.14  reachable  routing IV     3    1    Eth /0   2.14
 1.22  reachable  area           3    1    Eth /0   1.22
```

Purge C M

Removes access control lists and routing filters from the permanent database.

Syntax: purge

module access-control . . .
module routing-filter . . .

module access-control *circuit-specifier*

Removes access control lists from the permanent database. You can delete an entire access control list; you cannot delete one filter.

all circuits	Specifies all circuits on the router.
circuit name	Specifies the named circuit.

Example:

```
purge module access-control all circuits
```

module routing-filter *circuit-specifier*

Removes routing filters from the permanent database. You can purge a specified filter or you can purge all filters.

The options for the circuit-specifiers include the following:

all	Specifies all routing filters in the configuration memory.
circuit name	Specifies the routing filter for the named circuit.

Example:

```
purge module routing-filter all
```

Zero **C M**

Clears circuit counters in the volatile database, global counters in the volatile database, and counters in the access control list module.

Syntax: zero

c̲ircuit-specifier
e̲xecutor
m̲odule a̲ccess-control *circuit-specifier*

circuit-specifier

all circuits	Specifies all circuits on the router.
circuit [*name*]	Specifies the named circuit.
known circuits	Specifies all known circuits on the router.

Example:

```
zero all circuits
```

e̲xecutor

Sets all global counters in the volatile database to a zero value. There are no options.

Example:

```
zero executor
```

module access-control *circuit-specifier*

all circuits	Specifies all circuits on the router.
circuit [*name*]	Specifies the named circuit.

Example:

```
zero module access-control all circuits
```

Exit C M

Returns to the previous prompt level.

Syntax: exit

Example:

```
exit
```

8

Configuring and Monitoring DVMRP

This chapter describes how to configure DVMRP (Distance Vector Multicast Routing Protocol) using the DVMRP configuration commands and how to monitor DVMRP protocol activity using the DVMRP console commands.

For additional information about DVMRP, refer to the *Routing Protocols Reference Guide*.

8.1 Accessing the DVMRP Configuration Environment

For information about accessing the DVMRP configuration and console environments, see Chapter 1.

8.2 DVMRP Configuration and Console Commands

This section explains the DVMRP configuration and console commands. You enter the configuration commands at the `DVMRP Config>` prompt and enter the DVMRP console commands at the `DVMRP>` prompt. To enable a new or changed configuration, you must restart the router.

Table 8–1 summarizes the DVMRP configuration and console commands.

Table 8–1 DVMRP Configuration and Console Commands Summary

Command	Tasks	Function
? (Help)	Configuring and Monitoring	Lists all of the DVMRP configuration commands or lists the options associated with specific commands.
Dump routing tables	Monitoring	Displays the OSPF routes contained in the routing table.
DVMRP	Configuring	Enables or disables DVMRP.
Interface summary	Monitoring	Displays OSPF interface statistics and parameters.

8.2 DVMRP Configuration and Console Commands

Table 8–1 DVMRP Configuration and Console Commands Summary (Continued)

Command	Tasks	Function
Join	Monitoring	Configures the router to belong to one or more multicast groups.
Leave	Monitoring	Removes the router from membership in multicast groups.
List	Configuring	Displays the current DVMRP configuration.
Mcache	Monitoring	Displays a list of currently active multicast forwarding cache entries.
Mgroups	Monitoring	Displays the group membership of the router's attached interfaces.
MOSPF	Configuring	Sets the metric and threshold for the DVMRP interface running over MOSPF. This command also disables the MOSPF VIF.
Mstats	Monitoring	Displays various multicast routing statistics.
Phyint	Configuring	Sets the metric and threshold for LAN interfaces associated with DVMRP. This command also deletes LAN interfaces associated with DVMRP.
Tunnel	Configuring	Adds or deletes tunnels in a MOSPF/DVMRP configuration.
Exit	Configuring and Monitoring	Exits the DVMRP configuration process and returns to the CONFIG environment.

? (Help) C M

Lists the available commands from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example:

```
?
DVMRP
MOSPF
PHYINT
TUNNEL
EXIT
LIST
```

Example:

```
dvmrp ?
ON
OFF
```

Dump Routing Tables M

Displays the set of known DVMRP multicast sources. Each source is listed together with the DVMRP router it was learned from, an associated cost, and the number of seconds since the routing table entry was refreshed.

Syntax: dump

Example:

```
dump
                   Multicast Routing Table
    Type   Origin-Subnet   From-Gateway    Metric  Age  In  Out-Vifs
    DVMRP  18.26.0.0       192.35.82.97      10     30   1    0  2*
    DVMRP  18.58.0.0       192.35.82.97       4     30   1    0  2*
    DVMRP  18.85.0.0       192.35.82.97       4     30   1    0  2*
    DVMRP  18.180.0.0      192.35.82.97       3     30   1    0  2*
    DVMRP  36.8.0.0        192.35.82.97       9     30   1    0  2*
    DVMRP  36.56.0.0       192.35.82.97       7     30   1    0  2*
    DVMRP  36.103.0.0      192.35.82.97       9     30   1    0  2*
    DVMRP  128.61.0.0      192.35.82.97       8     30   1    0  2*
    DVMRP  128.89.0.0      192.35.82.97      10     30   1    0  2*
    DVMRP  128.109.0.0     192.35.82.97       4     30   1    0  2*
    DVMRP  128.119.0.0     192.35.82.97       4     30   1    0  2*
    DVMRP  128.150.0.0     192.35.82.97       6     30   1    0  2*
```

Type Displays the type of multicast sources (DVMRP).

Origin-Subnet Displays the IP address of the originating subnet.

From-Gateway Displays the IP address of the gateway from which the entry came.

Metric Displays the associated cost of that route.

Age Displays the age of routing table entry as the number of seconds since the routing table entry was refreshed.

In Displays the DVMRP VIF that multicast datagram from the source must be received on.

Out-Vifs Displays those VIFs that send the multicast datagrams. VIFs marked with an asterisk indicate that a datagram is only forwarded if there are group members on the attached network.

DVMRP C

Enables or disables DVMRP on the bridging router.

Syntax: dvmrp

on

off

on

Enables DVMRP on the bridging router. When enabled, DVMRP interfaces are automatically assigned to all LAN interfaces that are NOT running MOSPF.

Example:

```
dvmrp on
```

off

Disables DVMRP on the router.

Example:

```
dvmrp on
```

Interface Summary M

Displays the current list of DVMRP interfaces (or VIFs).

Syntax: interface *interface-ip-address*

Example:

```
interface
Virtual Interface Table
 Vif  Local-Address                            Metric  Thresh  Flags
  0   10.1.153.22      subnet: 10.1.153.0         1       1    querier
  1   10.1.154.22      subnet: 10.1.154.0         1       1    down
```

Vif Displays the number assigned to DVMRP interfaces (or VIFs) command. Each VIF is assigned a number, which is used to identify the VIF in other commands.

Local Address Displays the local IP address of the DVMRP interface.

Flags Displays whether the VIF is down or that the router is the querier (sender of IGMP Host Membership Queries) on the interface.

Join M

Establishes the router as a member of a multicast group.

This command is similar to the **join** command in the OSPF configuration console with two differences:

- The effect on group membership is immediate when the commands are given from the OSPF monitor (a restart/reload is not required).
- The command keeps track of the number of times a particular group is “joined.”

When the router is the member of a multicast group, it responds to pings and SNMP queries sent to the group address.

Syntax: join *multicast-group-address*

Example:

```
join  128.185.00.00
```

Leave M

Removes a router’s membership in a multicast group. This prevents the router from responding to pings and SNMP queries sent to the group address.

This command is similar to the **leave** command in the OSPF configuration console with two differences:

- The effect on group membership is immediate when the commands are given from the OSPF monitor (a restart/reload is not required).
- The command does not delete group membership until the “leaves” executed equals the number of “joins” previously executed.

Syntax: leave *multicast-group-address*

Example:

```
leave 128.185.00.00
```

8.2 DVMRP Configuration and Console Commands

List C

Displays the current DVMRP configuration. The output displays the current DVMRP state (*on* or *off*), tunnel configuration information, and MOSPF configuration information.

Syntax: list

Example:

```
list
DVMRP on
tunnel 0.0.0.0 0.0.0.0 1 1
MOSPF 1 1
```

Mcache M

Displays a list of currently active multicast cache entries. Multicast cache entries are built on demand, whenever the first matching multicast datagram is received. There is a separate cache entry (and therefore a separate route) for each datagram source network and destination group combination.

Cache entries are cleared on topology changes (for example, a point-to-point line in the MOSPF system going up or down), and on group membership changes.

Note: The numbers displayed in the legend at the top of the output do NOT refer directly to VIFs, but instead refer to physical interfaces (which may be running either DVMRP or MOSPF) and tunnels.

Syntax: mcache

Example:

```
mcache
            0: Eth /0           1: Internal
            2: 128.185.246.17

      Source          Destination     Count   Upst    Downstream
      128.185.146.0   239.0.0.1       1       0       2,4
      128.119.0.0     224.2.199.198   9       4       3
      128.9.160.0     224.2.127.255   1       4       3
      13.2.116.0      224.2.0.1       27      4       3
      140.173.8.0     224.2.0.1       31      4       3
      128.165.114.0   224.2.0.1       25      4       3
      132.160.3.0     224.2.158.99    11      4       3
      132.160.3.0     224.2.170.143   56      4       3
      128.167.254.0   224.2.199.198   27      4       3
      129.240.200.0   224.2.0.1       21      4       3
      131.188.34.0    224.2.0.1       28      4       3
```

Source Displays the source network/subnet of matching datagrams.

Destination Displays the destination group of matching datagrams.

Count Displays the number of entries processed for that multicast group.

Upstream Displays the neighboring network/router from which the datagram must be received in order to be forwarded. When this reads as *none*, the datagram is never forwarded.

Downstream Displays the total number of downstream interfaces/neighbors to which the datagram is forwarded. When this is 0, the datagram is not forwarded.

There is more information in a multicast forwarding cache entry. A cache entry can be displayed in detail by providing the source and destination of a matching datagram on the command line. If a matching cache entry is not found, one is built. A sample of this command is shown below:

Example:

```
mcache 128.185.182.9  224.0.1.2
source Net:     128.185.182.0
Destination:    224.0.1.2
Use Count:      472
Upstream Type:  Transit Net
Upstream ID:    128.185.184.114
Downstream:     128.185.177.11 (TTL = 2)
```

In addition to the information shown in the short form of the **mcache** command, the following fields are displayed:

Upstream Type Indicates the type of node from which the datagram must be received to be forwarded. Possible values for this field are *none* (indicating that the datagram is not forwarded), *router* (indicating that the datagram must be received over a point-to-point connection), *transit network*, *stub network*, and *external* (indicating that the datagram is expected to be received from another autonomous system).

Downstream Prints a separate line for each interface or neighbor to which the datagram is sent. A TTL value is also given, indicating that datagrams forwarded out of or to this interface must have at least the specified TTL value in their IP header. When the router is itself a member of the multicast group, a line specifying *internal Applicatio*n appears as one of the downstream interfaces/neighbors.

Mgroups M

Displays the group membership of the router's attached interfaces. This command displays only the group membership for those interfaces on which the router is either the designated router or the backup designated router.

Syntax: mgroups

Example:

```
mgroups
                 Local Group Database

  Group              Interface                  Lifetime (secs)

 224.0.1.1         128.185.184.11 (Eth /1)         176
 224.0.1.2         128.185.184.11 (Eth /1)         170
 224.1.1.1         Internal                        1
```

Group Displays the group address as it was reported (through IGMP) on a particular interface.

Interface Displays the interface address to which the group address was reported (through IGMP).

The router's internal group membership is indicated by a value of *internal*. For these entries, the lifetime field (see below) indicates the number of applications that have requested membership in the particular group.

Lifetime Displays the number of seconds that the entry persists if Membership Reports cease to be heard on the interface for the given group.

MOSPF C

Sets the metric and threshold for the DVMRP interface running over MOSPF. This command also disables the MOSPF VIF.

Syntax: mospf

metric threshold

delete

metric threshold

Sets the metric and threshold for the MOSPF VIF. Default values for the metric and threshold parameters are 1.

When using a MOSPF domain to join DVMRP tunnels, DVMRP is actually run over MOSPF. When this occurs, a DVMRP interface named **MOSPF VIF** (VIF or virtual interface) is automatically created. DVMRP tries to run over MOSPF automatically using the MOSPF VIF.

Example:

```
mospf 1 1
```

delete

Disables the MOSPF VIF. When MOSPF is enabled, DVMRP tries to run over MOSPF automatically using the MOSPF VIF.

Example:

```
mospf delete
```

8.2 DVMRP Configuration and Console Commands

Mstats M

Displays various multicast routing statistics. The command indicates whether multicast routing is enabled and whether the router is an inter-area and/or inter-AS multicast forwarder.

Syntax: mstats

Example:

```
mstats
                 MOSPF forwarding:       Enabled
                 Inter-area forwarding:  Enabled
                 DVMRP forwarding:       Enabled

  Datagrams received:      164612  Datagrams (ext source):        0
  Datagrams fwd (multicast):98807  Datagrams fwd (unicast):       0
  Locally delivered:            0  No matching rcv interface:     0
  Unreachable source:           0  Unallocated cache entries:     0
  Off multicast tree:       77230  Unexpected DL multicast:       0
  Buffer alloc failure:         0  TTL scoping:                   0

  # DVMRP routing entries:      1  # DVMRP entries freed:         1
  # fwd cache alloc:          649  # fwd cache freed:           648
  # fwd cache GC:               0  # local group DB alloc:        2
  # local group DB free:        2
```

MOSPF forwarding Displays whether the MOSPF is running (*Enabled*) or not (*Disabled*).

Inter-area forwading Displays whether the router forwards IP multicast datagrams between areas.

DVMRP forwarding Displays whether the DVMRP is running (*Enabled*) or not (*Disabled*).

Datagrams received Displays the number of multicast datagrams received by the router (datagrams whose destination group lies in the range 224.0.0.1 to 224.0.0.255 are not included in this total).

Datagrams (ext source) Displays the number of datagrams that were received whose source is outside the AS.

Datagrams fwd (multicast) Displays the number of datagrams that were forwarded as data-link multicasts. (This includes packet replications, so this count can be greater than the number received.)

Datagrams fwd (unicast) Displays the number of datagrams that were forwarded as data-link unicasts.

Locally delivered Displays the number of datagrams that were forwarded to internal applications.

No matching rcv interface Displays the count of those datagrams that were received by a non-inter-AS multicast forwarder on a non-MOSPF interface.

Unreachable source Displays a count of those datagrams whose source address was unreachable.

Unallocated cache entries Displays a count of those datagrams whose cache entries were not created due to resource shortages.

Off multicast tree Displays a count of those datagrams that were not forwarded, either because there was no upstream neighbor or no downstream interfaces/neighbors in the matching cache entry.

Unexpected DL multicast Displays a count of those datagrams that were received as data-link multicasts on those interfaces that were configured for data-link unicast.

Buffer alloc failure Displays a count of those datagrams that were not replicated because of buffer shortages.

TTL scoping Indicates those datagrams that were not forwarded because their TTL indicated that they were unable to reach a group member.

#DVMRP routing entries Indicates the number of DVMRP routing table entries currently in use.

#DVMRP entries freed Indicates the number of DVMRP routing table entries that have been freed.

# fwd cache alloc Indicates the number of cache entries allocated. The current forwarding cache size is the number of entries allocated (*# fwd cache alloc*) minus the number of cache entries freed (*# fwd cache freed*).

# fwd cache freed Indicates the number of cache entries freed. The current forwarding cache size is the number of entries allocated (*# fwd cache alloc*) minus the number of cache entries freed (*# fwd cache freed*).

# fwd cache GC Indicates the number of cache entries cleared because they were not recently used and the cache overflowed.

# local group DB alloc Indicates the number of local group database entries allocated. The number allocated (*# local group DB alloc*) minus the number freed (*# local group DB free*) equals the current size of the local group database.

# local group DB free Indicates the number of local group database entries freed. The number allocated (*# local group DB alloc*) minus the number freed (*# local group DB free*) equals the current size of the local group database.

The number of cache hits can be calculated as the number of datagrams received (*Datagrams received*) minus the total of datagrams discarded due to *No matching rcv interface*, *Unreachable source*, and *Unallocated cache entries*, and minus *# local group DB alloc*. The number of cache misses is simply *# local group DB alloc*.

Phyint C

Sets the metric and threshold for LAN interfaces associated with DVMRP. This command also deletes LAN interfaces associated with DVMRP.

Syntax: phyint
intrfc_address metric threshold
intrfc_address delete

intrfc_address metric threshold

Sets the metric and threshold for LAN interfaces (specified by the *intrfc_address* parameter) associated with DVMRP. Default values for the *metric* and *threshold* parameters are 1.

Example:

```
phyint XXXXX 1 1
```

intrfc_address delete

Deletes LAN interfaces associated with DVMRP.

Example:

```
phyint XXXXX delete
```

8.2 DVMRP Configuration and Console Commands

Tunnel C

Add tunnels or delete tunnels in a MOSPF/DVMRP configuration.

Syntax: tunnel

source-addr destination-addr metric threshold
source-addr destination-addr delete

source-adr destination-addr metric threshold

Adds a tunnel to a MOSPF/DVMRP configuration.

Example:

```
tunnel XXX XXX 1 1
```

***source-addr destination-addr* delete**

Deletes a tunnel from a MOSPF/DVMRP configuration.

Example:

```
tunnel XXX XXX delete
```

Exit C M

Returns to the previous prompt level.

Syntax: exit

Example:

```
exit
```

9

Configuring and Monitoring IP

This chapter describes how to configure the IP protocol and how to use the IP configuration and console commands.

For more information about IP, refer to the *Routing Protocols Reference Guide.*

9.1 Accessing the IP Configuration and Console Environments

For information about accessing the IP configuration and console (monitoring) environments, see Chapter 1.

9.2 Basic Configuration Procedures

This section outlines the initial steps required to set up and run the IP protocol. Details about making further configuration changes are covered in the command sections of this chapter.

The following steps outline the initial configuration tasks you must perform to configure IP on the router. After completing these steps, you must restart the router for the new configuration to take effect.

1. Access the IP configuration environment. (See Chapter 1.)
2. Assign IP addresses to hardware interfaces.
3. Enable dynamic routing.
4. Add static routing information (if necessary).
5. Enable ARP subnet routing (if necessary).
6. Set up IP access control.
7. Exit the IP configuration process.
8. Restart the router to activate the configuration changes.

9.2 Basic Configuration Procedures

The following sections discuss each configuration task in steps 2 through 6 in more detail.

9.2.1 Assigning IP Addresses to Network Interfaces

Use the IP configuration **add address** command to assign IP addresses to the network hardware interfaces. The arguments for this command include the hardware interface number (obtained from the `Config>` **list devices** command), and the IP address and its associated address mask.

In the following example, network interface 2 is assigned the address 128.185.123.22 with the associated address mask 255.255.255.0 (using the third byte for subnetting).

Example:

```
IP Config> add address 2 128.185.123.22 255.255.255.0
```

IP is automatically enabled whenever you assign at least one IP address to any of the router’s hardware interfaces. A hardware interface does not accept or send IP packets unless it has at least one IP address.

IP allows you to use a serial line interface for IP traffic without assigning an IP address to the line. However, you must still assign each serial line a label. Use the **add address** command to assign the serial line an address of the form 0.0.0.*n*, where *n* is the hardware interface number (again obtained from the `Config>` **list devices** command). This address format tells the router that the interface in question is an *unnumbered serial line*. Refer to Chapter 2 of the *Routing Protocols Reference Guide* for information about the limitations on unnumbered serial lines.

To enable IP on serial line interface number 2 to the router without assigning the interface an IP address, use the following command:

Example:

```
IP Config> add address 2 0.0.0.2
```

9.2.2 Enabling Dynamic Routing

Use the following procedures to enable dynamic routing on the router. The routers support OSPF and RIP for Interior Gateway Protocols as well as EGP (Exterior Gateway Protocol).

All three routing protocols can run simultaneously. However, most routers run only a single routing protocol (one of the IGPs). The OSPF protocol is recommended because it is robust and supports additional IP features (such as equal-cost multipath and variable-length subnets).

9.2.2.1 Enabling the OSPF Protocol

The OSPF routing protocol is enabled on an interface-by-interface basis. Each OSPF interface is assigned a cost. Also, an estimate of the OSPF database's size must be given, and the interaction between OSPF and the other two routing protocols (RIP and EGP) must be defined. Use the following procedures to initially configure OSPF.

OSPF configuration is done through its own configuration console (entered through the `Config>` **protocol ospf** command). To enable OSPF, use the following command:

Example:

```
OSPF Config> enable OSPF
```

After enabling the OSPF protocol, you are prompted for size estimates for the OSPF link state database. This tells the router software approximately how much memory to reserve for OSPF. You must supply the following two values that are used to estimate the size of the OSPF link state database:

- Total number of AS external routes imported into the OSPF routing domain. A single destination may lead to multiple AS external routes when imported by separate AS boundary routers. For example, if the OSPF routing domain has two AS boundary routers, both importing routes to the same 100 destinations, the number of AS external routes is set to 200.
- Total number of OSPF routers in the routing domain.

Enter these values at the following prompts (sample values are provided):

Example:

```
OSPF Config> enable ospf
Estimated # external routes[0]? 200
Estimated # OSPF routers [0]? 60
```

Next, configure each IP-interface that is to participate in OSPF routing. To configure an IP interface for OSPF, use the following command:

Example:

```
OSPF Config> set interface
```

You are prompted to enter a series of operating parameters. Each interface is assigned a cost as well as a list of OSPF operating parameters.

9.2 Basic Configuration Procedures

When running other IP routing protocols besides OSPF, you may want to enable the exchange of routes between OSPF and the other protocols. To do this, use the following command:

Example:

```
OSPF Config> enable AS-boundary-routing
```

For more information about the OSPF configuration process, see Chapter 12.

9.2.2.2 Enabling the RIP Protocol

This section describes how you initially configure the RIP protocol. When configuring the RIP protocol, you can specify which set of routes the router advertises or accepts on each IP interface, or both. You can also specify how RIP information affects static routing and the interaction between RIP and EGP. Since RIP uses broadcast messages for its routing updates, the format of the IP broadcast address must also be specified when using the RIP protocol.

First, enable the RIP protocol with the following command:

Example:

```
IP Config> enable RIP
```

By default, RIP advertises all network and subnet routes on all interfaces of the router. Once RIP is enabled, you can configure what it listens to and what it advertises by setting the various RIP flags. For detailed information about the RIP flags, consult the RIP description in the IP section of the *Routing Protocols Reference Guide*. These flags are configured on a per-IP-interface basis. The following commands can enable or disable the various flags:

Example:

```
IP Config> enable/disable sending net-routes
IP Config> enable/disable sending subnet-routes
IP Config> enable/disable sending static-routes
IP Config> enable/disable sending default-routes
IP Config> enable/disable receiving rip
IP Config> enable/disable receiving dynamic nets
IP Config> enable/disable receiving dynamic subnets
IP Config> enable/disable override default
IP Config> enable/disable override static-routes
```

The RIP protocol uses IP broadcast when sending its routing updates. Since there are different formats of IP broadcast in use, you must specify which broadcast format to use. The IP broadcast format is specified on a per-interface basis by using the following command:

Example:

```
IP Config> set broadcast-address IP-interface-address
Use a NET or LOCAL-WIRE style address [NETWORK]?
Fill pattern for wildcard part of address (0 or 1) [0]?
```

From the prompts, choose either the `LOCAL-WIRE` or `NETWORK` broadcast format and then select whether you want the rest of the broadcast address filled with either ones or zeroes. For more information about the RIP protocol, see the RIP description in the IP section of the *Routing Protocols Reference Guide*.

9.2.2.3 Enabling the Triggered RIP Protocol

This section describes how to configure the Triggered RIP protocol. Triggered RIP is an extension of the RIP protocol, so you must first configure RIP as described in the previous section. You may use Triggered RIP at both ends of a point-to-point link in order to reduce the bandwidth overhead of broadcast routing updates.

When configuring the Triggered RIP protocol, you can specify whether to enable or disable the call sensitivity feature. This is an extension to the Triggered RIP RFC for use with dialup connections (V.25 *bis* or ISDN). If the dialup circuit is not connected, then Triggered RIP may elect not to send the update until the connection is made for data transfer. This feature has no effect on permanent (leased line) interfaces or dial circuits with the idle timer set to zero (0).

First, enable the RIP protocol and set an IP address on the interface. For example:

Example:

```
IP Config> enable RIP
IP Config> add address 2 128.185.123.22
```

Configure the Triggered RIP protocol on the interfaces at both ends of a point-to-point link using the following command:

Example:

```
IP Config> set triggered-RIP
Set for which interface [0]? 2
Enable/Disable Call Sensitivity?(E/D) [Enable]?
Response Retransmission Timer (seconds)  [5]?
Maximum Response Retransmissions (0 for no limit)  [36]?
Initial Request Poll Timer (seconds)  [5]?
Maximum Initial Poll Retransmissions (0 for no limit)  [36]?
Failed-State Request Poll Timer (minutes)  [20]?
Maximum Failed-State Poll Retransmissions (0 for no limit)  [0]?
```

From the prompts, enable or disable the call sensitivity feature and set the timer values. Restart the router for this configuration to take effect. For more information about the Triggered RIP protocol, see the Triggered RIP description in the IP section of the *Routing Protocols Reference Guide*.

9.2.2.4 Enabling the EGP Protocol

This section describes how to initially configure the EGP protocol. Your router may need to run the EGP protocol if it is exchanging reachability information with routers belonging to other autonomous systems. For example, this may be the case if you have a MILnet/NSF backbone network connection.

Only routers that lie on the boundary of the autonomous system can run the EGP protocol. To enable EGP on your router, configure the following:

- An autonomous system number for the router running EGP
- A list of initial EGP neighbors
- The exchange of routing information between EGP and the IGPs

To enable the EGP protocol and configure the autonomous system number for your router, use the command shown in the following example:

Example:

```
IP config> enable EGP
EGP autonomous system number [0]? 47
```

You must assign the same AS number to all routers belonging to the same autonomous system.

9.2 Basic Configuration Procedures

Next, configure the list of initial routers with which you want to exchange EGP information. These routers are called EGP neighbors and belong to different autonomous systems. Use the **add EGP-neighbor** command to configure each EGP neighbor. With this command, you specify the IP address of the neighbor as well as the autonomous system to which it belongs. For example:

Example:

```
IP Config> add EGP-neighbor 192.9.1.1
AS id [1]? 32
```

In this example, the EGP neighbor’s IP address is 192.9.1.1 and the neighbor belongs to autonomous system number 32.

After configuring your EGP neighbors, configure the set of routes that you want to exchange with these neighbors. The route exchange is defined in two directions. In the **out** direction, you specify which routes you want to advertise through EGP. In the **in** direction, you specify which received EGP routes you want to readvertise through your IGPs (OSPF and/or RIP).

The EGP routing exchange can be defined on a per-neighboring-AS basis. If two of your EGP neighbors belong to separate ASs, then you can exchange separate sets of routes with each neighbor. To describe the set of routes to exchange with a neighboring autonomous system, use the **add-EGP-AS-info** command (followed by the neighbor’s AS number). After entering the command, a prompt asks you to select the direction you want the exchange of routes to follow.

In the following example, the interchange direction (flag) is **in**. This means that all routes received from the neighboring AS (here shown as AS number 32) are readvertised by OSPF and RIP. In this case, a prompt asks you for the metric (shown as 10) that is used (by OSPF and RIP) when readvertising the routes:

Example:

```
IP config> add EGP-AS-info 32
Interchange Flag (IN/OUT/OFF)- [OFF]? in
Default metric for IGP (-1 use EGP) [-1]? 10
```

If routes are not to be exchanged freely in one (or both) directions, those directions are table driven. When the interchange flag is not set to **out**, the Output Exchange Table lists all those routes that are advertised through EGP. Similarly, when the interchange flag is not set to **in**, the Input Exchange Table lists the received routes that you readvertise through OSPF or RIP, or both.

9.2 Basic Configuration Procedures

You can configure both the Output and Input Exchange Tables on a per-neighboring-AS basis. Both tables consist of lists of IP networks. To add an IP network to or delete an IP network from one of the tables, use one of the following commands (followed by the desired AS number):

Example:

```
IP config> add/delete input-interchange
IP config> add/delete output-interchange
```

Each network is added to an Input/Output Exchange Table together with the route cost that is advertised. In the following example, the command specifies that if network 18.0.0.0 is received by EGP from autonomous system 32, it readvertises the IGPs with a cost of 2:

Example:

```
IP config> add input-interchange 32
Destination network [0.0.0.0]? 18.0.0.0
Metric to advertise (-1 use EGP) [-1]? 2
```

Entries in the Output Exchange Table can specify that a route is to be advertised through EGP only if it was originally received from a particular autonomous system. In the next example, the command specifies that a route to network 10.0.0.0 is advertised (through EGP) to AS 32, but only if the route was originally received from AS number 50. In this case, the route is advertised by EGP with a cost of 3.

Example:

```
IP config> add output-interchange 32
Source AS id (0 for don’t care) [0]? 50
Destination network (0.0.0.0 for all) [0.0.0.0]? 10.0.0.0
Metric to advertise (-1 use IGP) [-1]? 3
```

You can find more detailed information about the EGP route exchange in the commands section of this chapter.

9.2.2.5 Using EGP Routers As Defaults

In EGP environments, the EGP router is usually the authoritative router since it has the knowledge of the routers and networks in other autonomous systems.

You can configure a router running EGP to advertise itself as the default router through its IGPs (OSPF and RIP). This is called originating default. When this feature is enabled, the router advertises itself as the default only if it has EGP-derived routers in its IP routing table.

Use the following command to enable this feature within OSPF:

Example:

```
OSPF Config> enable/disable AS-boundary-routing
```

For more details, see the configuration commands in Chapter 12.

To enable this feature within RIP, use the following commands:

Example:

```
IP Config> enable/disable originate-default
IP Config> set advertised default-metric
IP Config> enable/disable sending default
```

For more details, see the configuration command section of this chapter.

9.2.3 Using the IS-IS Protocol in a Combined DECnet and IP Network

Figure 9–1 shows an example configuration with IP routing. In this case, there is no need to accommodate either the RIP protocol or static IP routes. Therefore, use the IS-IS protocol. Note that you cannot use the IS-IS protocol on any level 1 only routing circuit if the ROUTING MANUAL L1 ALGORITHM is ROUTING VECTOR.

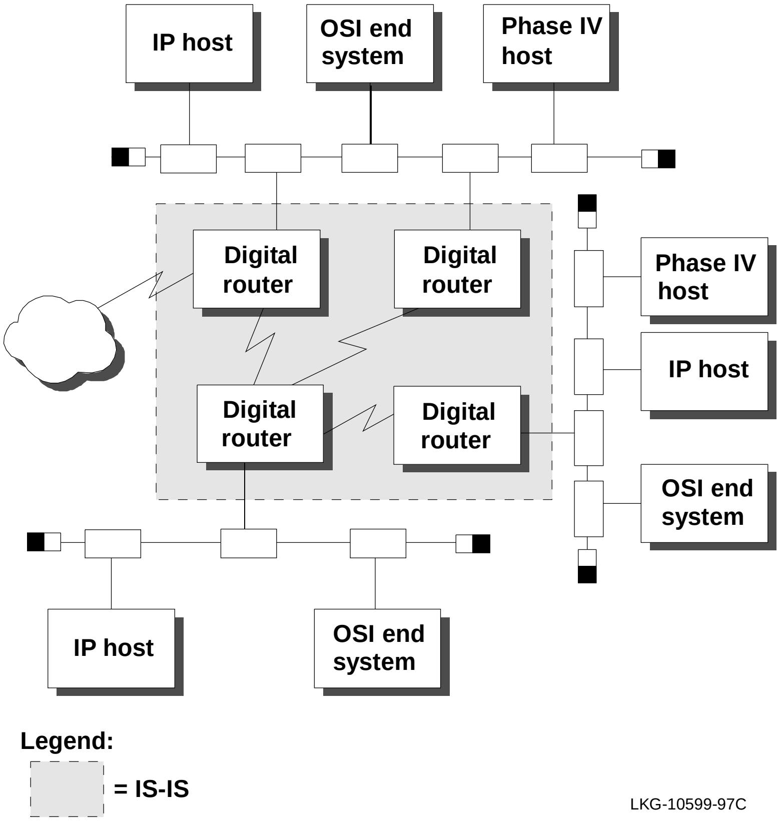
To add the IS-IS protocol to exchange IP routing information as shown in Figure 9–1:

1. Run link state routing (DECnet Phase V) at level 2 and level 1. A router that runs link state routing at level 1 and/or level 2 uses the IS-IS protocol at that level.
2. Assign IP addresses and masks to the interfaces indicated in the diagram.

   For a point-to-point link, do not assign addresses. However, assign IP addresses over all the circuits that you expect to carry IP packets because this helps you diagnose problems.
3. At the `OSI config>` prompt, enable the integrated IS-IS protocol by entering the command **enable integrated-isis**, and answer the questions about route configuration.

9.2 Basic Configuration Procedures

Figure 9–1 Using IS-IS in an IP Configuration



9.2.4 Adding Static Routing Information

This procedure is necessary only if you cannot gain routing information from any of the above dynamic routing protocols. Static routing persists over power failures and is used for routes that never change or are not dynamic. Static routing information consists of any of the following items:

- **Default Gateway** – Packets are routed to default (authoritative) gateways when the packet destination cannot be found in the routing table.
- **Default Subnet Gateways** – If you are using subnetted networks, you can define a separate default gateway for each subnetted network.
- **Static Network/Subnet Routes** – For each destination with a fixed route, configure the next hop and distance to the destination.

9.2.4.1 Default Gateway

Routers send packets having unknown destinations (destinations not present in the routing table) toward the default gateway. A default gateway is configured in the router by specifying the next hop to use to get to the default gateway and the cost of sending packets to the default gateway.

In the following example, the next hop toward the default gateway is 192.9.1.4 and the cost of sending a packet to the default gateway is 5:

Example:

```
IP Config> set default network-gateway
Default gateway [0.0.0.0]? 192.9.1.4
gateway’s cost [0]? 5
```

Default gateways can be learned and advertised by both the OSPF and RIP protocol. For the OSPF protocol, a router can be configured to advertise itself as the default gateway with the following OSPF command:

Example:

```
OSPF Config> enable/disable AS-boundary-routing
```

9.2 Basic Configuration Procedures

The RIP protocol can be configured so that it advertises knowledge of the default gateway (if it has any) to its neighbors. RIP can also be configured so that a learned default gateway overrides (or does not override) a statically configured default gateway. These configuration tasks are accomplished with the following two commands:

Example:

```
IP Config> enable/disable sending default-routes
IP Config> enable/disable override default
```

Finally, a router that runs EGP or BGP can be configured to advertise itself (through the OSPF and RIP protocols) as the default gateway. It can do this always, or it can be set to do this only when it has routes learned through EGP or BGP. For OSPF, this is accomplished through the OSPF **enable/disable AS-boundary-routing** configuration command. For RIP, the IP **set originate-rip-default** configuration command is used.

9.2.4.2 Default Subnet Gateways

There can be a default subnet gateway configured for each subnetted network that the router knows about. When the router attempts to forward a packet to a destination belonging to the subnetted network, but that destination cannot be found in the routing table, the packet is forwarded instead to the default subnet gateway.

Configuring default subnet gateways is the same as configuring the preceding default network gateway. The only difference is that you must specify the subnetted network on the command line. For example, to create a default subnet gateway for the subnetted network 128.0.0.0, you can use the following command:

Example:

```
IP Config> set default subnet-gateway 128.0.0.0
Default gateway [0.0.0.0]? 128.185.123.22
gateway’s cost [0]? 2
```

The above example specifies that the next hop to the subnet default gateway is 128.185.123.22, and that the cost of routing a packet to the default subnet gateway is 2.

9.2.4.3 Static Network/Subnet Routes

Configure static routes for those destinations that cannot be discovered by the dynamic routing protocols. The destination is described by an IP network/subnet number (**dest-addr**) and the destination's address mask (**mask**). The route to the destination is described by the IP address of the first hop router to use (**1st-hop**) and the cost of routing a packet to the destination (**cost**). To create/modify/delete a static route, use the commands:

Example:

```
IP Config> add route dest-addr mask 1st-hop cost
IP Config> change route dest-addr mask 1st-hop cost
IP Config> delete route dest-addr mask
```

Routes dynamically learned through the OSPF and RIP protocols can override static routes. For the RIP protocol, you can disable this override behavior. See Section 9.2.2.2 for information about the **enable/disable override static-routes** commands.

9.2.5 Enabling ARP Subnet Routing

If attached subnetted networks have hosts that do not support IP subnetting, you must use Address Resolution Protocol (ARP) subnetting routing (described in RFC 1027). When the router is configured for ARP subnet routing, it replies by proxy to ARP requests for destination (off the LAN if the router is the best route to the destination). For proper operation, all routers attached to a LAN containing subnetting-ignorant hosts are configured for ARP subnet routing.

To enable ARP subnet routing, use the following command:

Example:

```
IP Config> enable ARP-subnet-routing
```

9.2.6 Enabling RFC 925 ARP Subnet Routing

Some IP hosts use ARP for all destinations, whether or not they are attached to the local network segment. For these hosts, ARP subnet routing is not enough and you must see the proxy ARP functionality specified in RFC 925 instead. RFC 925 ARP routing is a subset of ARP subnet routing.

To enable RFC 925 ARP routing, use the following command:

Example:

```
IP Config> enable RFC-925
```

9.2.7 Setting Up IP Access Control

The IP access control system allows the IP forwarder to control packet forwarding based on source and destination IP addresses, IP protocol number, and port number for the TCP and UDP protocols. This can control access to particular classes of IP addresses and services.

The IP access control system is based on one global ordered list of inclusive and exclusive access control entries. If access control is enabled, each IP packet being originated, forwarded, or received is subject to the access control list. Each entry in the list may be inclusive or exclusive, permitting or denying forwarding. Each entry has fields for source and destination IP address, optional IP protocol number, and optional port number for UDP and TCP.

For each received packet, the headers are compared to all specified fields in each entry in the list in turn. If the entry matches the packet and the entry is inclusive, the packet is forwarded. If the entry is exclusive, the packet is dropped. If no entry matches after going through the entry list the packet is dropped.

Each entry has an IP address mask and result pair for both the source and destination IP addresses. An address is logically “AND-ed” with the mask, and compared to the result. For example, a mask of 255.0.0.0 with a result of 26.0.0.0 matches any address with 26 in the first byte. A mask of 255.255.255.255 with a result 192.67.67.20 matches only the IP host 192.67.67.20. A mask of 0.0.0.0 with a result of 0.0.0.0 is a wildcard, and matches any IP address.

Each entry may also have an optional IP protocol number range. This applies to the protocol byte in the IP header. Any IP packet with a protocol value within the specified range matches. A range of 0 to 255 matches all IP packets. The commonly used protocol numbers are 1 for ICMP, 6 for TCP, 8 for EGP, 17 for UDP, and 89 for OSPF.

Each entry may also have an optional port number range. This applies only to TCP and UDP packets because the port number is part of the TCP and UDP headers. Any TCP or UDP packet with a destination port number within the specified range matches. (TCP and UDP use the same port numbers.) A range of 0 to 65535 disables port filtering. Some commonly used port numbers are 21 for FTP, 23 for Telnet, 25 for SMTP, 513 for rlogin, 520 for RIP, and 6000 for X.

The following example allows any host to send packets to the SMTP TCP socket on 192.67.67.20:

Example:

```
   IP Config> add access-control inclusive 0.0.0.0 0.0.0.0
              192.67.67.20 255.255.255.255 6 6 25 25
```

The next example prevents any host on subnet 1 of Class B network 150.150.0.0 from sending packets to hosts on subnet 2 of Class B network 150.150.0.0 (assuming a 1-byte subnet mask):

Example:

```
   IP Config> add access-control exclusive 150.150.1.0
              255.255.255.0 150.150.2.0 255.255.255.0 0 255 0 65535
```

This command allows the router to send and receive all RIP packets:

Example:

```
   IP Config> add access-control inclusive 0.0.0.0 0.0.0.0 0.0.0.0
              0.0.0.0 17 17 520 520
```

This command allows the router to send and receive all OSPF packets:

Example:

```
   IP Config> add access-control inclusive 0.0.0.0 0.0.0.0
              0.0.0.0 0.0.0.0 89 89
```

If IP access control is enabled, you must be careful with packets that the router originates and receives. Be sure not to filter out the RIP or OSPF packets being sent or received by the router. The easiest way to do this is to add a wildcard inclusive entry as the last in the access control list. Alternately, you can add specific entries for RIP or OSPF, or both, perhaps with restrictive addresses and masks. Note that some OSPF packets are sent to the Class D multicast addresses 224.0.0.5 and 224.0.0.6, which is important if address checking is being done for routing protocols. See the **add** command section in this chapter for more information about access control.

9.2.8 Setting Up IP Network Filters

If you have certain IP networks/subnets that you do not want to forward packets to, nor distribute routing information about, specify those networks as filters (this is more efficient than the access control mechanism). To add a network filter, use the following command:

Example:

```
   IP Config> add filter IP-address IP-mask
```

To avoid propagating packets destined as loopback packets, filter to the local loopback network 127.0.0.0. Use the following command:

Example:

```
   IP Config> add filter 127.0.0.0 255.0.0.0
```

9.2.9 Configuring Packet Filtering on Specific Interfaces

You can configure packet filters that enable your router to block or forward packets on a per-interface basis. The filters can affect incoming or outgoing traffic based on packet source and destination addresses, protocol number, or transport port number.

To implement per-interface filtering, you define packet filters and update them with appropriate inclusive/exclusive access control entries. Unlike IP access-controls, which apply to the router as a whole, each packet filter applies to only one router interface. This means that a filter designed for interface *n* is only used for incoming or outgoing packets on that interface. The filter has no effect on packets seen by the router’s other interfaces.

9.2.10 How Packet Filters Work

Packet filters work in the same way as access controls. When the IP forwarder receives a packet, it checks to see whether any packet filters are defined for it on the interface on which the packet arrives. If a filter exists, and the packet is not excluded by it, the packet passes through the access control list defined for the entire router. If the packet is being forwarded, it passes through any access control list specified for outgoing traffic on the outbound interface. In either case, the router discards all packets that are not explicitly included by a filter.

9.2.10.1 Defining Packet Filters

To define packet filters, use the **add packet-filter** command at the `IP config>` prompt. For example:

Example:

```
   IP config> add packet-filter
```

The router prompts you for the filter’s name, its direction (either IN or OUT), and the number of the interface to which it applies.

```
   Packet-filter name [ ]? test
   Filter incoming or outgoing traffic? [ IN]?
   Which interface is this filter for [0]? 1
```

You can use the **list packet-filter** command to list all packet filters configured in the router. If you specify a name, as shown in the following example, the router also lists the access control information configured for the specified packet filter.

```
IP config>list packet-filter test

Name               Direction     Interface
test               In            0

Access Control is: enabled
List of access control records:

                                                  Beg  End Beg End
    Ty Source      Mask      Destination  Mask     Pro  Pro Prt Prt
1   E  128.185.0.0 FFFF0000  0.0.0.0      00000000 0   255  0  65535
2   I  0.0.0.0     00000000  0.0.0.0      00000000 0   255  0  65535
```

The next section explains how to configure access control information for packet filters.

9.2.10.2 Setting Up Access Control Entries For Packet Filters

You must assign access control entries to each defined filter. Otherwise, defined filters have no affect.

Use the **update packet-filter** command at the `IP config>` prompt to assign access control entries. The router prompts you for the name of the filter you want to update. The `IP config>` prompt changes to incorporate the packet filter name you provide.

```
IP Config> update packet-filter
Packet-filter name [ ]? test
Packet-filter ’test’ Config>
```

You can access a list of subcommands by entering **?** at the `Packet-filter ’name’ Config>` prompt.

```
Packet-filter ’test’ Config> ?
LIST
DELETE
ADD
MOVE
EXIT
```

9.2.10.3 Using the Add Access-Control Command

Use the **add access-control** command to add access controls to the packet filter. The router prompts you for the access control type (Exclusive or Inclusive), and the source and destination addresses and masks of packets to which the filter applies.

This example shows how to exclude all incoming packets originating from network 128.185.0.0 and received on interface 0:

```
Packet-filter ’pf-in-0’ Config>add access-control
Enter type [E]?
Internet source [0.0.0.0]? 128.185.0.0
Source mask [255.255.255.255]? 255.255.0.0
Internet destination [0.0.0.0]?
Destination mask [255.255.255.255]? 0.0.0.0
Enter starting protocol number ([CR] for all) [-1]?
```

9.2.10.4 Using the List, Move, and Delete Commands

Commands for packet filtering are similar to those used to modify IP access controls. The only difference is that you issue these commands at the `Packet-filter ’name’ Config>` prompt, where `’name’` represents the specific packet filter you are configuring.

9.2.10.5 Listing Access Controls for a Filter

To list the access controls configured for a filter, use the **list access-control** command as shown below:

Example:

```
Packet-filter ’test’ Config> list access-control
Access Control is: enabled
List of access control records:
                                                  Beg  End Beg End
    Ty Source      Mask      Destination  Mask    Pro  Pro Prt Prt
1   E  12.15.0.0   FFFF0000  0.0.0.0      00000000 0   255  0  65535
2   I  1.2.9.9     FFFFFFFF  0.0.0.0      00000000 0   255  0  65535
3   E  128.185.0.0 FFFF0000  0.0.0.0      00000000 0   255  0  65535
4   I  0.0.0.0     00000000  0.0.0.0      00000000 0   255  0  65535
```

9.2.10.6 Completing the IP Configuration Process

To complete the IP configuration process, you must perform two more steps. You must exit `IP config>` (enter the **exit** command) and you must reboot the router.

You can now operate your router as an internet router between two or more IP networks.

9.3 The BOOTP Forwarding Process

BOOTP (documented in RFC 951) is a bootstrap protocol used by a diskless workstation to learn its IP address and the location of its boot file and boot server. BOOTP requests/replies are forwarded at the application level (UDP). They are not forwarded at the network level. This means that the IP header changes as the packet is forwarded. The workstation broadcasts the request in a UDP packet to the routers, and in turn, the routers forward the packets to BOOTP servers.

The following terms are useful when discussing the BOOTP forwarding process:

- BOOTP client – the diskless workstation
- BOOTP servers – the boot host (with UNIX daemon bootpd or DOS version available from FTP software)
- BOOTP relay agent or BOOTP forwarder – your router

The following steps outline an example of the BOOTP forwarding process:

1. The BOOTP client copies its Ethernet address (or appropriate MAC address) into a BOOTP packet and broadcasts it onto the local LAN. BOOTP is running on top of UDP.
2. The local BOOTP relay agent receives the packet and checks to see if the packet is formatted correctly and that the maximum number of application hops did not expire. It also checks to see if the client tried long enough.

   Note: If multiple hops are required before reaching the BOOTP agent, the packet is routed normally through IP. All other routers do not examine the packet to determine whether it is a BOOTP packet.

If this is the case:

3. The local BOOTP agent forwards a separate BOOTP request to each of its configured BOOTP servers. The BOOTP request is the same as the one that was initially sent by the client except that it has a new IP header with the relay agent’s IP address copied into the body of the BOOTP request.
4. The BOOTP server receives the request and looks up the client’s Ethernet address in its database. If found, it formats a BOOTP reply containing the client’s IP address and boot file name. The reply is then sent to the BOOTP relay agent.

5. The BOOTP relay agent receives the reply and makes an entry in its ARP table for the client and then forwards the reply to the station.
6. The station then continues to boot using TFTP, using the information in the BOOTP reply packet.

9.3.1 Enabling/Disabling BOOTP Forwarding

To enable or disable BOOTP forwarding on the router, enter the following command line at the IP configuration prompt:

Example:

```
IP Config> enable/disable bootp
```

When enabling BOOTP, you are prompted for the following values:

- Maximum number of application hops you want the BOOTP request to go. This is the maximum number of BOOTP relay agents that can forward the packet. This is NOT the maximum number of IP hops to the BOOTP server. A typical value for this parameter is 1.
- Number of seconds you want the client to retry before the BOOTP request is forwarded. A typical value for this parameter is 0.

  Note: This parameter is not commonly used.

After accepting a BOOTP request, the router forwards the BOOTP request to each BOOTP server. If there are multiple servers configured for BOOTP, the transmitting server replicates the packet.

9.3.2 Configuring a BOOTP Server

To add a BOOTP server to the router’s configuration, enter the following command at the IP configuration prompt:

Example:

```
IP Config> add BOOTP-SERVER [IP address of server]
```

Multiple servers can be configured. In addition, if only the network number of the server is known or if multiple servers reside on the same network segment, a broadcast address can be configured for the server.

9.4 IP Configuration and Console Commands

This section summarizes and explains the IP configuration and console commands. The configuration commands allow you to modify the IP protocol behavior to meet your specific requirements. Some amount of configuration is necessary to produce a fully functional IP router. Enter IP configuration commands at the `IP Config>` prompt.

The console commands allow you to monitor the router's IP forwarding process. The monitoring capabilities include the following: configured parameters such as interface address and static routes can be viewed, the current state of the IP routing table can be displayed, and a count of IP routing errors can be listed. Enter console commands at the `IP>` prompt.

Table 9–1 summarizes the IP configuration and console commands.

Table 9–1 IP Configuration and Console Commands Summary

Command	Tasks	Function
? (Help)	Configuring and Monitoring	Lists the configuration commands or lists the actions associated with specific commands.
Access controls	Monitoring	Lists the current IP access control mode together with the configured access control records.
Add	Configuring	Adds to the IP configuration information. Interface addresses can be added, along with access controls, route filters, packet filters, EGP exchange information, and EGP neighbors.
Cache	Monitoring	Displays a table of all recent routed destinations.
Change	Configuring	Modifies information that was originally entered with the **add** command.
Counters	Monitoring	Lists various IP statistics, including counts of routing errors and packets dropped.
Delete	Configuring	Deletes IP configuration information that was entered with the **add** command.
Disable	Configuring	Disables certain IP features that were turned on by the **enable** command.
Dump	Monitoring	Lists the contents of the IP routing table.
EGP-neighbors	Monitoring	Displays the current state of all EGP neighbors.
EGP-routes	Monitoring	Displays the routing information that either can be or was exchanged with a particular EGP neighbor.

9.4 IP Configuration and Console Commands

Table 9–1 IP Configuration and Console Commands Summary (Continued)

Command	Tasks	Function
Enable	Configuring	Enables IP features such as ARP subnet routing, EGP, originate default, directed broadcasts, BOOTP, and the various RIP flags controlling the sending and receiving of RIP information.
Interface	Monitoring	Lists the router's IP interface addresses.
List	Configuring	Displays IP configuration items.
Move	Configuring	Changes the order of access control records.
Packet-Filter	Monitoring	Displays information for a specific packet filter or for all packet filters.
Ping	Monitoring	Sends ICMP Echo Requests to another host once a second and watches for a response. This command can be used to isolate trouble in an internetwork environment.
Route	Monitoring	Lists whether a route exists for a specific IP destination, and if so, the routing table entry that corresponds to the route.
Set	Configuring	Establishes IP configuration modes such as the type of access control and the format of broadcast addresses. Configures Triggered RIP. Also sets IP parameters such as default routers and the size of the IP routing table.
Sizes	Monitoring	Displays the configured sizes of specific IP parameters.
Static	Monitoring	Displays the static routes that were configured. This includes the default gateway.
Traceroute	Monitoring	Displays the complete path (hop-by-hop) to a particular destination.
Triggered-rip	Monitoring	Displays the current state and counters for Triggered RIP on an interface.
Update	Configuring	Accesses the commands to modify packet filters.
Exit	Configuring and Monitoring	Exits the IP configuration process.

9.4 IP Configuration and Console Commands

? (Help) C M

Lists the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example:

```
?
LIST
CHANGE
DELETE
DISABLE
ENABLE
ADD
SET
MOVE
UPDATE
EXIT
```

Example:

```
add ?
ACCEPT-RIP-ROUTE
ACCESS-CONTROL
ADDRESS
BOOTP-SERVER
BROADCAST-FORWARDER
EGP-AS-INFO
EGP-NEIGHBOR
FILTER
INPUT-INTERCHANGE
OUTPUT-INTERCHANGE
PACKET-FILTER
ROUTE
```

9.4 IP Configuration and Console Commands

Access Controls M

Prints the access control mode in use together with a list of the configured access control records.

The access control mode is one of the following: *disabled* (meaning that no access control is being done and the access control records are being ignored) or *enabled* (meaning that access control is being done and the access control records are being recognized). When access control is enabled, access control records are scanned in order looking for the first match.

Exclusive (*E*) means that packets matching the access control record are being discarded. Inclusive (*I*) means that packets matching the access control record are being forwarded. When access control is enabled, packets failing to match any access control record are discarded. *Pro* (protocol) indicates the IP protocol number and *Prt* (port) indicates the UDP or TCP port number.

Syntax: access

Example:

access

```
Access control currently ENABLED
List of access control records:

                                                         Beg End  Beg End
      Source       Mask       Destination   Mask         Pro Pro  Prt Prt
0 I   0.0.0.0      00000000   192.67.67.20  FFFFFFFF     6   6    25  25
1 E   150.150.1.0  FFFFFF00   150.150.2.0   FFFFFFFF     0   255  0   65535
2 I   0.0.0.0      00000000   0.0.0.0       00000000     89  89   0   65535
```

9.4 IP Configuration and Console Commands

Add C

Adds IP information to your configuration. This command lets you add interface addresses, access controls, filters, EGP exchange information, and EGP neighbors.

Syntax: add

accept-rip-route . . .
access-control . . .
address . . .
bootp-server
broadcast-forwarder
egp-as-info . . .
egp-neighbor . . .
filter . . .
input-interchange . . .
output-interchange . . .
packet-filter . . .
route . . .

accept-rip-route ***IP-network/subnet***

Allows an interface to accept a RIP route when input filtering is enabled for an interface. You can prompt the list of networks/subnets that are already entered using the **list rip-routes-accept** command. You can enable the input filtering of RIP routes on a per-IP-interface basis. This is done separately for network-level routes (for example, a route to 10.0.0.0) and for subnet-level routes (for example, a route to 128.185.0.0). To enable input filtering of network-level routes on an IP interface, use the **disable dynamic nets** command. To enable input filtering of subnet-level routes, use the **disable dynamic subnets** command.

Example:

```
add accept-rip-route 10.0.0.0
```

access-control ***type IP-source source-mask IP-dest dest-mask [first-protocol last-protocol] [first-port last- port]***

Adds an access control entry to the end of the access control list. This allows you to describe a class of packets to forward or drop, depending on the type of the entry.

The length and order of the IP access control list can affect the performance of the IP forwarder.

9.4 IP Configuration and Console Commands

This command adds an IP access control entry to the end of the list. Each entry must be assigned the following: type, IP source, source-mask, IP destination, and destination-mask fields. The *type* must either be inclusive or exclusive. The *IP-source* and *IP-dest* fields are in the form of IP addresses in dotted decimal notation. Optionally, you may specify an IP protocol number range with the *first-protocol* and *last-protocol* fields, which are an inclusive range of IP protocols that match this entry. If a range of protocols was specified, you may specify a TCP and UDP port number range with the *first-port* and *last-port* fields, which are an inclusive range of TCP and UDP ports that matches this entry.

Note: Before access control can become effective, you must enable it with the **set access-control on** command.

Example:

```
add access-control inclusive 0.0.0.0 0.0.0.0
         192.67.67.20 255.255.255.255 6 6 25 25
```

address ***interface-number IP-address address-mask***

Assigns an IP address to one of the router's hardware network interfaces. A hardware network interface does not receive or transmit IP packets until it has at least one IP address.

You must specify an IP address together with its subnet mask. For example, if the address is on a class B network, using the third byte for subnetting, the mask is 255.255.255.0. Use the **list devices** command to obtain the appropriate command *interface-number*. Serial lines do not need addresses. Such lines are called *unnumbered*. However, you must still enable them for IP traffic using the **add address** command. The address then used is 0.0.0.*n*, where *n* is the *interface-number.*

Example:

```
add address 0 128.185.123.22 255.255.255.0
```

bootp-server ***server-IP-address***

Adds a BOOTP server to a network configuration. Acting as a boot relay agent, your router accepts and forwards BOOTP requests to the BOOTP server. BOOTP is a bootstrap protocol used by a diskless workstation to learn its IP address and the location of its boot file and boot server.

Note: Before the **list all** command can display the BOOTP server address, you must enable BOOTP forwarding with the **enable bootp-forwarding** command.

Example:

```
add bootp-server 128.185.123.22
```

broadcast-forwarder udp ***port Interface address***

Adds a broadcast forwarding server for a specified UDP port on a specified network interface. The default interface is **all**, and by default the entry is enabled.

Example:

```
add broadcast-forwarder udp
Specify Port Number [1-65534]? 200
Specify Interface Number [All]?1
Enter address to be Added [0.0.0.0]? 16.24.10.255
```

egp-as-info ***neighboring-AS interchange-flag [default-metric]***

Defines the type of EGP route exchange that takes place when communicating with a neighboring autonomous system.

The EGP route exchange is defined in a bidirectional fashion. One direction, called **in**, defines the set of routes that, when received from the neighboring AS, are readvertised through OSPF and RIP. The other direction, called **out**, defines the set of routes that is advertised through EGP to the neighboring AS.

The configured *interchange-flag* indicates if EGP route exchange is free (all routes exchanged) in a particular direction. The *interchange-flag*'s permissible values are **in**, **out**, and **off**.

If the exchange is free in one direction, you can configure a default metric (*default-metric*) for the (re)advertisements. In the **in** direction, this means that OSPF and RIP readvertises all routes with the given cost. In the **out** direction, this means that all routes in the routing table are advertised with the given cost. Setting the default cost to -1 means that the readvertisement or advertisement does not change the metrics; A cost of -2 means that the specified network is not added to the routing table; and a cost -3 places the network in the routing table, but does advertise it to other routers.

When EGP route exchange is not free in a particular direction, it is table driven. The table for the **in** direction is called the Input Exchange Table. The table for the **out** direction is called the Output Exchange Table. See the commands **add input-interchange** and **add output-interchange** for information about updating these tables.

Even when the *interchange-flag* is set to **out**, EGP routes that were originally received from a particular AS are not be readvertised back to that AS.

When the EGP route exchange is set to be free in one direction, the entries in the appropriate Input/Output Exchange Table override the *default-metric* on a per-route basis.

When the *neighboring-AS* is specified as 0, the default EGP exchange is defined. This controls EGP route exchange with all neighboring ASs that were not specified in the **add egp-as-info** commands. The following example reads: readvertise through OSPF and RIP all EGP routes received from autonomous system 47. The default cost to use in the readvertisements is 10.

Example:

```
add egp-as-info 47 IN 10
```

egp-neighbor ***neighbor-IP-address neighbor-AS***

Adds an initial EGP neighbor to the router's IP configuration. If the EGP protocol is enabled, it attempts to contact each router configured as an EGP neighbor.

Each EGP neighbor is configured together with the neighbor's autonomous system number. The neighbor's AS number is then verified before an EGP connection to the neighbor is fully established. The set of routes that are exchanged with the EGP neighbor is configured on a neighboring-AS basis. If two of the router's EGP neighbors belong to the same neighboring AS, the same set of routes is exchanged with each neighbor.

For more information about the route exchange, see the **add egp-as-info** command. The router can form EGP neighbor relationships with neighbors that have not been configured with the **add egp-neighbor** command. In this case, the neighbor's AS membership cannot be verified, and the set of routes exchanged with such a neighbor is governed by the ‘‘default” EGP route exchange (the exchange configured for the special AS number 0).

Example:

```
add egp-neighbor 10.0.0.7 1
```

filter ***dest-IP-address address-mask***

Designates an IP network/subnet to be filtered. IP packets are not forwarded to filtered networks/subnets, nor is routing information disseminated concerning such destinations. Packets destined for filtered network/subnets are simply discarded.

You must specify a filtered network/subnet together with its subnet mask. For example, to filter a subnet of a class B network, using the third byte for subnetting, the mask is 255.255.255.0.

Using the filter mechanism is more efficient than IP access controls, although not as flexible. Filters also affect the operation of the IP routing protocols, unlike access controls.

Example:

```
add filter 127.0.0.0 255.0.0.0
```

input-interchange ***neighboring-AS IP-network metric***

Adds an IP network to the list of routes that, when received from a neighboring autonomous system, is readvertised through OSPF and RIP. You can configure a separate list for each neighboring AS.

The list of routes specified by this command is called the neighboring AS's Input Exchange Table. Each route is specified by its IP network number and the cost that OSPF and RIP readvertised.

If the metric is set to -1, the readvertised cost is identical to the cost received by the EGP protocol. If the metric is set to -2, the route is not installed in the routing table. If the metric is -3, the route is added to the routing table, but is not advertised to other routers. Before creating the Input Exchange Table, use the **add egp-as-info** command to define the neighboring AS's *interchange-flag.* When the *interchange-flag* is set to **in**, all routes received from the neighboring AS (not just those specified in the Input Exchange Table) are readvertised by OSPF and RIP. In this case, adding routes to the Input Exchange Table can specify the metric used for the readvertisement on a route-to-route basis (overriding the default value specified in the **add egp-as-info** command).

When the *neighboring-AS* is specified as 0, the route is added to the default Input Exchange Table. This table controls EGP route readvertisement for all neighboring ASs that have not been specified in **add egp-as-info** commands.

The example below reads as follows: when receiving an EGP route to 10.0.0.0 from autonomous system 47, readvertise it through OSPF and RIP with a cost of 5.

Example:

```
add input-interchange 47 10.0.0.0 5
```

output-interchange ***neighboring-AS source-AS IP-network metric***

Adds an IP network to the list of routes that EGP advertises to the neighboring autonomous system. You can configure a separate list for each neighboring AS.

The list of routes specified by this command is called the neighboring AS's Output Exchange Table. Each route is specified by its IP network number and the cost that EGP advertises.

If the metric is set to -1, the cost advertised by EGP is identical to the route’s routing table cost.

In any case, routes originally received from a particular autonomous system are not readvertised back to that same AS.

When the *neighboring-AS* is specified as 0, the route is added to the default Output Exchange Table. This table lists routes that EGP sends to all neighboring ASs that were not specified in the **add egp-as-info** commands.

Entering a value other than 0 as the *source-AS* indicates that the route is advertised only if the route was received from the specified AS. Entering 0 indicates that the route is advertised regardless of the *source-AS.*

The example below reads as follows: if a route to network 8.0.0.0 exists and was learned from AS 7, advertise it to autonomous system 47 with a cost of 3.

Example:

```
add output-interchange 47 7 8.0.0.0 3
```

packet-filter ***name IN/OUT interface***

Adds a packet filter to block or forward packets on a per interface basis. Each packet filter must then be updated with a list of access controls.

Note: Before packet filters are effective, you must enable access control with the **set access-control on** command.

Example:

```
add packet-filter
Packet-filter name [ ]? filt-1-0
Filter incoming or outgoing traffic? [ IN]?
Which interface is this filter for [0]?1
```

route ***IP-network/subnet IP-mask next-hop cost***

Adds static network/subnet routes to the router’s IP configuration. When dynamic routing information is not available for a particular destination, static routes are used.

The destination is specified by an IP address (*IP-network/subnet*) together with an address mask (*IP-mask*). For example, if the destination is a subnet of a class B network, and the third byte of the IP address is used as the subnet portion, the address mask is set to 255.255.255.0.

The route to the destination is specified by the IP address of the next hop (*next-hop*) and the cost (*cost*) of routing the packet to the destination. The next hop must be on the same (sub)net as one of the router’s directly connected interfaces.

Example:

`add route 17.0.0.0 255.0.0.0 128.185.123.22 6`

Cache M

Displays the IP routing cache that contains recently routed destinations. If a destination is not in the cache, the router looks up the destination in the routing information table in order to make a forwarding decision.

Syntax: cache

Example:

cache

```
Destination       Usage                 Next hop

128.185.128.225     1                   128.185.138.180     (Eth /0)
192.26.100.42       1                   128.185.138.180     (Eth /0)
128.185.124.121     4                   128.185.124.121     (Eth /0)
```

Destination IP destination host.

Usage Number of packets recently sent to the destination host.

Next hop IP address of the next router on the path toward the destination host. Also displayed is the network name of the interface used by the sending router to forward the packet.

Change C

Changes an IP configuration item previously installed by the **add** command. In general, you must specify the item you want to change, just as you specified the item with the **add** command.

Syntax: change

address . . .
egp-as-info . . .
egp-neighbor . . .
filter . . .
input-interchange . . .
output-interchange . . .
route . . .

9.4 IP Configuration and Console Commands

address *old-address new-address new-mask*

Modifies one of the router's IP interface addresses. You must specify each new address together with the new address' subnet mask. This command can also be used to change an existing address' subnet mask.

Example:

```
change address 192.9.1.1 128.185.123.22 255.255.255.0
```

egp-as-info *as-id new-interchange [default-metric]*

Modifies the interchange flag associated with a neighboring AS (*as-id*). If the interchange flag is set to either in or out, you must specify a default metric for route advertisement in that direction.

Example:

```
change egp-as-info 32 in 3
```

egp-neighbor *nbr-ip-address new-as*

Modifies the configured EGP neighbor's AS membership.

Example:

```
change egp-neighbor 192.9.1.3 47
```

filter *destination new-mask*

Modifies the subnet mask associated with a filtered network/subnet. Networks that are filtered become black holes. No packets are forwarded to them nor is routing information distributed about them.

Example:

```
change filter 127.0.0.0 255.0.0.0
```

input-interchange *as-id net-number new-cost*

Modifies the cost associated with a network appearing in an neighboring AS's (*as-id*) Input Exchange Table.

Example:

```
change input-interchange 32 8.0.0.0 6
```

output-interchange *as-id source-as net-number new-cost*

Modifies the cost associated with a *source-as/network* pair appearing in a neighboring AS's (*as-id*) Output Exchange Table.

Example:

```
change output-interchange 32 47 128.185.0.0 1
```

route *destination new-mask new-1st-hop new-cost*

Modifies either the subnet mask, next hop, or cost associated with a configured static network/subnet route.

Example:

```
change route 10.0.0.0 255.0.0.0 128.185.123.18 6
```

Counters M

Displays the statistics related to the IP forwarding process. This includes a count of routing errors, along with the number of packets that were dropped due to congestion.

Syntax: counters

Example:

```
counters
Routing errors
Count   Type
    0   Routing table overflow
 2539   Net unreachable
    0   Bad subnet number
    0   Bad net number
    0   Unhandled broadcast
58186   Unhandled multicast
    0   Unhandled directed broadcast
 4048   Attempted forward of LL broadcast

Packets discarded through filter  0
IP multicasts accepted:           60592

IP input packet overflows
   Net    Count
   Eth /0  0
   Eth /1  0
```

Routing table overflow Lists the number of routes that were discarded due to the routing table being full.

Net unreachable Indicates the number of packets that were not forwarded due to unknown destinations. This does not count the number of packets that were forwarded to the authoritative router (default gateway).

Bad subnet number Counts the number of packets or routes that were received for illegal subnets (all ones or all zeroes).

Bad net number Counts the number of packets or routes that were received for illegal IP destinations (for example, class E addresses).

Unhandled broadcast Counts the number of (nonlocal) IP broadcasts received (these are not forwarded).

Unhandled multicast Counts the number of IP multicasts that were received but whose address was not recognized by the router (these are discarded).

Unhandled directed broadcast Counts the number of directed (nonlocal) IP broadcasts received when forwarding of these packets is disabled.

Attempted forward of LL broadcast Counts the number of packets that are received having nonlocal IP addresses but were sent to a link level broadcast address. These are discarded.

Packets discarded through filter Counts the number of received packets that were addressed to filtered networks/subnets. These are discarded silently.

IP multicasts accepted Counts the number of IP multicasts that were received and successfully processed by the router.

IP input packet overflows Counts the number of packets that were discarded due to congestion at the forwarder’s input queue. These counts are sorted by the receiving interface.

Delete C

Deletes an IP configuration item previously installed by the **add** command. In general, you must specify the item you want to delete, just as you specified the item with the **add** command. The exception is Triggered RIP, which is installed with the **set** command.

Syntax: delete

accept-rip-route . . .
access-control . . .
address . . .
bootp-server
broadcast-forwarder udp
default network/subnet-gateway . . .
egp-as-info . . .
egp-neighbor . . .
filter . . .
input-interchange . . .
ip-host-only-default . . .
output-interchange . . .
packet-filter . . .
route . . .
triggered-rip . . .

accept-rip-route *net-number*

Removes a route from the list of networks that the RIP protocol always accepts.

Example:

```
delete accept-rip-route 10.0.0.0
```

access-control *record-number*

Deletes one of the access control records.

Example:

```
delete access-control  2
```

address *ip-interface-address*

Deletes one of the router's IP interface addresses.

Example:

```
delete address 128.185.123.22
```

9.4 IP Configuration and Console Commands

bootp-server ***server-IP-address***

Removes a BOOTP server from an IP configuration.

Example:

```
delete bootp-server 128.185.123.22
```

broadcast-forwarder udp ***port Interface address***

Removes a broadcast forwarding server on a specified UDP port and network interface.

Example:

```
add broadcast-forwarder udp
Specify Port Number [1-65534]? 200
Specify Interface Number [All]?1
Enter address to be Deleted [All]? 16.24.10.255
```

default network/subnet-gateway ***[subnetted network]***

Deletes either the default gateway or the default subnet gateway for the specified subnetted network.

Example:

```
delete default subnet-gateway 128.185.0.0
```

egp-as-info ***as-id***

Deletes the route exchange description for the specified neighboring AS (*as-id*). Without this record, you cannot configure Input or Output Exchange Tables for the neighboring AS. The default tables are used instead (specified by setting *as-id* equal to 0).

Example:

```
delete egp-as-info 32
```

egp-neighbor ***nbr-ip-address***

Deletes an initial EGP neighbor.

Example:

```
delete egp-neighbor 10.0.0.7
```

filter ***destination***

Deletes one of the router’s filtered networks.

Example:

```
delete filter 127.0.0.0
```

input-interchange *as-id net-number*

Deletes a network from a neighboring AS’s (*as-id*) Input Exchange Table.

Example:

```
delete input-interchange 32 8.0.0.0
```

ip-host-only-default

Deletes the default gateway used by the router when in host-only mode.

Example:

```
delete ip-host-only-default
```

output-interchange *as-id source-as net-number*

Deletes a source AS or network pair from a neighboring AS’s (*as-id*) Output Exchange Table.

Example:

```
delete output-interchange 32 47 128.185.0.0
```

packet-filter *name* or ***all***

Removes the specified packet filter and its associated access controls. If *all* is specified, all packet filters and their associated access controls are deleted.

Example:

```
delete packet-filter
Enter packet-filter name to be deleted []? filt-1-0
All access controls defined for ’filt-1-0’ will also be deleted.
Are you sure you want to delete(Yes or [No]): y
Deleted
```

route *destination*

Deletes one of the router’s configured static routes.

Example:

```
delete route 10.0.0.0
```

triggered-rip *interface*

Deletes Triggered RIP from the specified interface.

Example:

```
delete triggered-rip
Enter interface number [0]? 2
```

9.4 IP Configuration and Console Commands

Disable C

Disables IP features previously enabled by the **add** or **enable** command.

Syntax: disable

arp-subnet-routing
bootp-forwarding
broadcast-forwarding udp
directed-broadcast
egp
egp-readvertise
override default . . .
override static-routes . . .
per-packet-multipath . . .
receiving rip . . .
receiving dynamic nets . . .
receiving dynamic subnets . . .
rfc925-routing
rip
sending default-routes . . .
sending net-routes . . .
sending poisoned-reverse-routes . . .
sending subnet-routes . . .
sending static-routes . . .

arp-subnet-routing

Turns off the IP feature called ARP subnet routing or proxy ARP. When enabled, it deals with hosts that have no IP subnetting support. This is the default and the generally recommended setting.

Example:

```
disable arp subnet routing
```

bootp-forwarding

Turns off the BOOTP relay function.

Example:

```
disable bootp-forwarding
```

9.4 IP Configuration and Console Commands

broadcast-forwarding udp *port Interface address*

Disables broadcast forwarding for a specified UDP port and network interface.

Example:

```
disable broadcast-forwarding udp
Specify Port Number [1-65534]? 200
Specify Interface Number [All]?1
```

directed-broadcast

Disables the forwarding of IP packets whose destination is a nonlocal (for example, remote LAN) broadcast address. The source host originates the packet as a unicast where it is then forwarded as a unicast to a destination subnet and “exploded” into a broadcast. You can use these packets to locate network servers.

Note: Forwarding and exploding cannot be disabled separately.

Example:

```
disable directed-broadcast
```

egp

Turns off the EGP protocol.

Example:

```
disable egp
```

egp-readvertise

Prevents EGP from readvertising routes that were originally learned from EGP.

Example:

```
disable egp-readvertise
```

override static-routes *ip-interface-address*
override default *ip-interface-address*

Prevents an RIP default route received on interface *ip-interface-address* from being installed as the router’s default route. The **disable override static-routes** command prevents RIP routes received on interface *ip-interface-address* from overriding any of the router’s static routes.

Example:

```
disable override default 128.185.123.22
```

9.4 IP Configuration and Console Commands

per-packet-multipath

Turns off the per-packet-multipath feature, which is enabled by default. With the feature enabled, IP can use as many as four equal-cost paths to a destination subnet, which are selected in round robin fashion. Disabling the feature causes IP to use a single path.

Example:

```
disable per-packet-multipath
```

receiving rip ***ip-interface-address***

Prevents any RIP packets form being received on interface *ip-interface-address*.

Example:

```
disable receiving rip 128.185.123.22
```

receiving dynamic nets ***ip-interface-address***
receiving dynamic subnets ***ip-interface-address***

Ensures that RIP updates receiving on the interface *ip-interface-address* accept only those network level routes entered by the **add accept-rip-route** command. The **disable receiving dynamic subnets** command produces the analogous behavior for subnet routes.

Example:

```
disable receiving dynamic nets 128.185.123.22
```

rfc925-routing

Turns off RFC 925 routing. When this is enabled, the router replies by proxy to all ARP requests for remote destinations that are best reached through the router.

Example:

```
disable rfc925-routing
```

rip

Turns off the RIP protocol.

Example:

```
disable rip
```

sending default-routes ***ip-interface-address***

sending net-routes ***ip-interface-address***

sending poisoned-reverse-routes ***ip-interface-address***

sending subnet-routes ***ip-interface-address***

sending static-routes ***ip-interface-address***

Prevents the router from advertising a default route in RIP updates sent out the interface *ip-interface-address*. The other flags controlling the RIP routes sent out an interface are **net-routes**, **poisoned-reverse-routes**, **subnet-routes**, and **static-routes**. You can turn these off individually. A route is advertised if it is specified by any of the enabled flags.

Example:

```
disable sending net-routes 128.185.123.22
```

Dump M

Displays the IP routing table. A separate entry is printed for each reachable IP network/subnet. The IP default gateway in use (if any) is listed at the end of the display.

Syntax: dump

Example:

```
dump
Type   Dest net         Mask     Cost Age   Next hop(s)
SPE1   0.0.0.0          00000000  4    3    128.185.138.39    (2)
SPF*   128.185.138.0    FFFFFF00  1    1    Eth /0
Sbnt   128.185.0.0      FFFF0000  1    0    None
SPF    128.185.123.0    FFFFFF00  3    3    128.185.138.39    (2)
SPF    128.185.124.0    FFFFFF00  3    3    128.185.138.39    (2)
SPF    192.26.100.0     FFFFFF00  3    3    128.185.131.10    (2)
RIP    197.3.2.0        FFFFFF00  10   30   128.185.131.10
RIP    192.9.3.0        FFFFFF00  4    30   128.185.138.21
Del    128.185.195.0    FFFFFF00  16   270  None


Default gateway in use.

Type Cost Age  Next hop
SPE1 4    3    128.185.138.39

Routing table size: 768 nets (36864 bytes), 36 nets known
```

Type (route type)

Indicates how the route was derived:

- **Sbnt** – Indicates that the network is subnetted; such an entry is a placeholder only.
- **Dir** – Indicates a directly connected network or subnet.
- **RIP** – Indicates the route was learned through the RIP protocol.
- **Del** – Indicates the route was deleted.
- **Stat** – Indicates a statically configured route.
- **EGP** – Indicates routes learned through the EGP protocol.
- **EGPR** – Indicates routes learned through the EGP protocol that are readvertised by OSPF and RIP.
- **Fltr** – Indicates a routing filter.
- **SPF** – Indicates that the route is an OSPF intra-area route.
- **SPIA** – Indicates that it is an OSPF inter-area route.
- **SPE1**, **SPE2** – Indicates OSPF external routes (types 1 and 2, respectively).
- **Rnge** – Indicates a route type that is an active OSPF area address range and is not used in forwarding packets.
- **RDR** – This is only applicable when running bridging (host-only mode). Indicates a route learned by ICMP Redirect from the local router.
- **GWD** – This is only applicable when running bridging (host-only mode). Indicates a route learned by ICMP discovery packet.
- **I1** – Indicates that the route is an Integrated ISIS Level 1 route.
- **I2I** – Indicates that the route is an Integrated ISIS Level 2 Internal route.

- **I2S** – Indicates that the route is an Integrated ISIS Level 2 Summary route.
- **I2EI** – Indicates that the route is an Integrated ISIS Level 2 External route with Internal metric.
- **I2EE** – Indicates that the route is an Integrated ISIS Level 2 External route with External metric.
- **IID** – Indicates that the route is an Integrated ISIS Level 2 implicit default route.
- **BGP** – Indicates routes learned through the BGP protocol.
- **BGPR** – Indicates routes learned through the BGP protocol that are readvertised by IGPs.

Dest net IP destination network/subnet.

Mask IP address mask.

Cost Route cost.

Age For RIP and EGP routes, the time that has elapsed since the routing table entry was last refreshed.

Next hop(s) IP address of the next router on the path toward the destination host. Also displayed is the interface type used by the sending router to forward the packet.

An asterisk (*) after the route type indicates that the route has a static or directly connected backup. A percent sign (%) after the route type indicates that RIP updates are always accepted for this network/subnet.

A number in parentheses at the end of the column indicates the number of equal-cost routes to the destination. The first hops belonging to these routes can be displayed with the IP **route** command.

9.4 IP Configuration and Console Commands

EGP-Neighbors M

Displays the current EGP state and the configured EGP interchange flag for each of the router's EGP neighbors. Each EGP neighbor is identified by its IP address. The EGP neighbor states are explained in the EGP specification (RFC 904).

Syntax: egp-neighbor

Example:

```
egp-neighbor
EGP neighbor 128.185.138.11   State:NEIGHBOR    Flag:in
EGP neighbor 128.185.138.19   State:NEIGHBOR    Flag:out
```

In Indicates that all routes received from the neighbor are readvertised through OSPF and RIP.

Out Indicates that all routes found in the routing table are sent to the neighbor.

Off Indicates that the EGP route exchange is determined completely by the Input and Output Exchange Tables. For more information, see the `IP config>` **add egp-as-info** command.

EGP-Routes M

Displays the routes that are being sent to and received from an EGP neighbor. The EGP neighbor's current state and configured interchange flag are also listed as in the IP **egp-neighbors** command.

Syntax: egp-routes *egp-neighbor-address*

Example:

```
egp-routes 128.185.138.11
EGP neighbor 128.185.138.11    State:NEIGHBOR    Flag:in

Routes advertised to EGP neighbor
Destination         Cost         Source AS
128.185.0.0           13              6
192.9.12.0            13              6

Routes obtained from EGP neighbor
Destination         Cost
129.9.0.0             5       Route Advertised through IGP
192.26.101.0          4       Route Advertised through IGP
192.26.100.0          4       Route Advertised through IGP
128.52.0.0            6       Route Advertised through IGP
18.0.0.0              6       Route Advertised through IGP
```

Routes advertised to EGP neighbor
Lists the IP networks that are currently being sent (in EGP poll responses) to the EGP neighbor. The advertised cost is also listed, together with the autonomous system that originally supplied the route.

Routes obtained from EGP neighbor
Lists the current portion of the IP routing table that was received from the EGP neighbor. Those routes that are readvertised by OSPF and RIP are suitably marked.

9.4 IP Configuration and Console Commands

Enable C

Activates IP features, capabilities, and information added to your IP configuration.

Syntax:

```
enable
                arp-subnet-routing
                bootp-forwarding
                broadcast-forwarding udp
                directed-broadcast
                egp . . .
                egp-readvertise
                override default . . .
                override static-routes . . .
                per-packet-multipath
                receiving rip . . .
                receiving dynamic nets . . .
                receiving dynamic subnets . . .
                rfc925-routing
                rip
                sending default-routes . . .
                sending net-routes . . .
                sending poisoned-reverse-routes . . .
                sending subnet-routes . . .
                sending static-routes . . .
```

arp-subnet-routing

Turns on the router’s ARP subnet routing (sometimes also called Proxy ARP) functionality. This functionality is used when there are subnet-incapable hosts attached to directly connected IP subnets. The directly connected subnet having subnet-incapable hosts must use ARP for this feature to be useful.

The way ARP subnet routing works is as follows. When a subnet-incapable host wants to send an IP packet to a destination on a remote subnet, it does not realize that it must send the packet to a router. The subnet-incapable host therefore simply broadcasts an ARP request. This ARP request is received by the router. The router responds as the destination (hence the name proxy) if both arp-subnet-routing is enabled and if the next hop to the destination is over a different interface than the interface receiving the ARP request.

If there are no hosts on your LAN that are "subnet-incapable," do not enable ARP-subnet routing. If ARP-subnet-routing is needed on a LAN, it is enabled on all routers on that LAN.

Example:

```
enable arp-subnet-routing
```

bootp-forwarding

Turns on BOOTP packet forwarding. In order to use the BOOTP forwarding, you must also add one or more BOOTP servers with the **add bootp-server** command.

Example:**enable bootp-forwarding**

```
Maximum number of forwarding hops [4]?
Minimum seconds before forwarding [0]?
```

Maximum number of forwarding hops	Maximum number of allowable BOOTP agents that can forward a BOOTP request from the client to the server (this is **not** the maximum number of IP hops to the server). Default: 4.
Minimum seconds before forwarding	This parameter is generally not used. Use this parameter when there is a redundant path between the client and the server, and you want to use the secondary paths as a standby.

broadcast-forwarding udp *port Interface address*

Enables broadcast forwarding for a specified UDP port and network interface.

Example:

```
enable broadcast-forwarding udp
Specify Port Number [1-65534]? 200
Specify Interface Number [All]?1
```

directed-broadcast

Enables the forwarding of IP packets whose destination is a nonlocal (for example, remote LAN) broadcast address. The packet is originated by the source host as a unicast where it is then forwarded as a unicast to a destination subnet and "exploded" into a broadcast. These packets can be used to locate network servers. This command enables both the forwarding and exploding of directed broadcasts. The IP packet forwarder never forwards link-level broadcasts/multicasts, unless they correspond to Class D IP addresses. (See the OSPF **enable multicast-routing** command.) The default setting for this feature is enabled.

Note: Forwarding and exploding cannot be implemented separately. Also, the router does not forward subnet wide IP broadcasts.

Example:

```
enable directed-broadcast
```

egp *own-as-id*

Enables the router's EGP protocol capabilities. When enabling the router's EGP capability, the router's own autonomous system number (*own-as-id*) must be specified. These are assigned by Stanford Research Institute's Network Information Center. All routers belonging to the same autonomous system are configured with the same AS number.

Example:

```
enable egp 21
```

egp-readvertise

Turns on EGP readvertising. This allows the router to readvertise through EGP routes that were originally learned from EGP.

Note: A route can always be advertised through EGP regardless of its origin if it is specified in an Output Exchange Table. (See the IP **add output interchange** command.)

Example:

```
enable egp-readvertise
```

originate-default

Originates a default RIP route whenever the router has EGP-derived routes in its routing table.

RIP advertises such a default route with a metric of 1 unless otherwise specified in the **set default-metric** command. In order for RIP to actually send a default route out a particular interface, you must also invoke the **enable sending default** command for the interface.

Example:

```
enable originate-default
```

override default *ip-interface-address*

Enables received RIP information to override the router's default gateway. This command is invoked on a per-IP-interface basis. When the **enable override default** command is invoked, default RIP routes received on interface *ip-interface-address* overwrite the router's current default gateway, providing the cost of the new default is cheaper.

Example:

```
enable override default 128.185.123.22
```

override static-routes *ip-interface-address*

Enables received RIP information to override some of the router's statically configured routing information. This command is invoked on a per-IP-interface basis. When the **enable override static-routes** command is invoked, RIP routing information received on interface *ip-interface-address* overwrite statically configured network/subnet routes providing the cost of the RIP information is cheaper.

Example:

```
enable override static-routes 128.185.123.22
```

per-packet-multipath

Turns on the per-packet-multipath feature, which is enabled by default. With the feature enabled, IP can use as many as four equal-cost paths to a destination subnet, which are selected in round robin fashion. This technicque is known as path-splitting. Disabling the feature causes IP to use a single path.

Example:

```
enable per-packet-multipath
```

receiving rip *ip-interface-address*

Modifies the processing of RIP updates that are received on a particular interface. This command has an analogous **disable** command. (See the **disable receiving** command.) This command is enabled by default. The opposite command is easier to describe, and that is done in the following paragraph.

If you invoke the **disable receiving rip** command, no RIP updates are accepted on interface *ip-interface-address* address.

Example:

```
enable receiving rip 128.185.123.22
```

9.4 IP Configuration and Console Commands

receiving dynamic nets *ip-interface-address*

Modifies the processing of RIP updates that are received on a particular interface. This command has an analogous **disable** command. (See the **disable receiving** command.) This command is enabled by default. The opposite command is easier to describe, and that is done in the following paragraph.

If you invoke the **disable receiving dynamic nets** command, RIP updates received on interface *ip-interface-address* cannot accept any network-level routes unless they were previously specified in an **add accept-rip-route** command.

Example:

```
enable receiving dynamic nets 128.185.123.22
```

receiving dynamic subnets *ip-interface-address*

Modifies the processing of RIP updates that are received on a particular interface. This command has an analogous **disable** command. (See the **disable receiving** command.) This command is enabled by default. The opposite command is easier to describe, and that is done in the following paragraph.

If you invoke the **disable receiving dynamic subnets** command, RIP updates received on interface *ip-interface-address* cannot accept any subnet-level routes unless they were previously specified in an **add accept-rip-route** command.

Example:

```
enable receiving dynamic subnets 128.185.123.22
```

rfc925-routing

Turns on RFC 925 routing. When enabled, the router replies by proxy to all ARP requests for remote destinations that are best reached through the router. Use this command when there are hosts on the LAN that ARP for all destinations, instead of (as is proper) only local destinations.

Example:

```
enable rfc925-routing
```

rip

Enables the router’s RIP protocol processing. When the RIP protocol is enabled, use the **enable/disable sending** commands to configure its routing update sending behavior. Its routing update receiving behavior is defined by the **enable/disable receiving** and **enable/disable override** commands.

Example:

```
enable rip
```

sending default-routes ***ip-interface-address***

Determines the contents of RIP updates that are sent out a particular interface. This command has an analogous **disable** command. (See the **disable sending** command.)

The effect of the **enable sending** command is cumulative. Each separate **enable sending** command specifies that a certain set of routes is advertised from a particular interface. A route is included in a RIP update only if it was included by at least one of the **enable sending** commands. The **enable sending default-routes** command specifies that the default route (if one exists) is included in RIP updates sent out interface *ip-interface-address.*

Example:

```
enable sending default-routes 128.185.123.22
```

Note: Some settings of the **enable sending** commands are redundant. For example, if you invoke **enable sending net-routes** and **enable sending subnet-routes** for a particular interface, there is no need to also specify **enable sending static-routes** (because each static route is either a network-level or subnet route). By default, when you first enable RIP, **sending net-routes** and **sending subnet-routes** are enabled for each interface, while **sending static-routes** and **sending default** are disabled.

sending net-routes ***ip-interface-address***

Determines the contents of RIP updates that are sent out a particular interface. This command has an analogous **disable** command. (See the **disable sending** command.)

The effect of the **enable sending** command is cumulative. Each separate **enable sending** command specifies that a certain set of routes is advertised from a particular interface. A route is included in an RIP update only if it was included by at least one of the **enable sending** commands. The **enable sending network-routes** command specifies that all network-level routes are included in RIP updates sent out interface *ip-interface-address*. A network-level route is a route to a single class A, B, or C IP network.

Example:

```
enable sending net-routes 128.185.123.22
```

sending poisoned-reverse-routes ***ip-interface-address***

Determines the contents of RIP updates that are sent out a particular interface. This command has an analogous **disable** command. (See the **disable sending** command.)

The effect of the **enable sending** command is cumulative. Each separate **enable sending** command specifies that a certain set of routes is advertised from a particular interface. A route is included in an RIP update only if it was included by at least one of the **enable sending** commands. The **enable sending poisoned-reverse-routes** command specifies that all network-level routes are included in RIP updates sent out interface *ip-interface-address*. A network-level route is a route to a single class A, B, or C IP network.

Example:

```
enable sending poisoned-reverse-routes 128.185.123.22
```

sending subnet-routes ***ip-interface-address***

Determines the contents of RIP updates that are sent out a particular interface. This command has an analogous **disable** command. (See the **disable sending** command.)

The effect of the **enable sending** command is cumulative. Each separate **enable sending** command specifies that a certain set of routes is advertised through a particular interface. A route is included in a RIP update only if it was included by at least one of the **enable sending** commands. The **enable sending subnet-routes** command specifies that all subnet routes are included in RIP updates sent out interface *ip-interface-address*. However, a subnet route is included only if *ip-interface-address* connects directly to a subnet of the same IP subnetted network.

Example:

```
enable sending subnet-routes 128.185.123.22
```

sending static-routes ***ip-interface-address***

Determines the contents of RIP updates that are sent out a particular interface. This command has an analogous **disable** command. (See the **disable sending** command.)

The effect of the **enable sending** command is cumulative. Each separate **enable sending** command specifies that a certain set of routes is advertised through a particular interface. A route is included in a RIP update only if it was included by at least one of the **enable sending** commands. The **enable sending static-routes** command specifies that all statically configured and directly connected routes are included in RIP updates sent out interface *ip-interface-address.*

Example:

```
enable sending static-routes 128.185.123.22
```

9.4 IP Configuration and Console Commands

Interface M

Displays the router's IP interface addresses. Each address is listed together with its corresponding hardware interface and IP address mask.

Hardware interfaces having no configured IP interface addresses are not used by the IP forwarding process; they are listed as **Not an IN net.** There is one exception. Serial lines need not be assigned IP interface addresses in order to forward IP traffic. Such serial lines are called *unnumbered*. They show up as having address 0.0.0.0.

Syntax: interface

Example:

```
interface
Interface  IP Address(es)   Mask(s)
    SL /0   Not an IN net
    SL /1   0.0.0.0          0.0.0.0
    SL /2   Not an IN net
    Eth /0  128.185.138.19   255.255.255.0
```

Interface Indicates the hardware type of the interface.

IP address(es) Indicates the IP address of the interface.

Mask(s) Indicates the subnet mask of the interface.

List C

Displays various pieces of the IP configuration data, depending on the particular subcommand invoked.

Syntax: list

all
access-controls
addresses
bootp
broadcast-forwarding
egp-as-info
egp-neighbors
filters
input-interchange . . .

output-interchange . . .
packet-filter . . .
protocols
rip-routes-accept
routes
sizes
tags
triggered-rip

all

Prints the entire IP configuration.

Example:

```
list all
```

access-controls

Prints the configured access control mode (inclusive, exclusive, or disabled) and the list of configured access control records. Each record is listed with its record number. This record number can be used to reorder the list with the IP **move access-control** command.

Example:

```
list access control
```

addresses

Prints the IP interface addresses that were assigned to the router, along with their configured broadcast formats.

Example:

```
list addresses
```

bootp

Indicates whether BOOTP forwarding is enabled or disabled, as well as the configured list of BOOTP servers.

Example:

```
list bootp
```

broadcast-forwarding

Displays all the protocols, ports, states and destinations (servers) configured for broadcast forwarding.

Example:

```
list broadcast-forwarding
Broadcast-Forwarder
Protocol  Port   Interface  State   Destinations
--------  ----   ---------  -----   ------------
UDP       200     1         Enable  16.24.10.255
                                    16.24.11.255

UDP       201    All        Enable  16.24.10.255
UDP       202     1         Disable 16.24.10.255
------------------------------------------------
```

egp-as-info

Prints the configured interchange flags for each of the neighboring autonomous systems.

Example:

```
list egp-as-info
```

egp-neighbors

Prints the list of initial EGP neighbors that were configured.

Example:

```
list egp neighbors
```

filters

Displays a list of all the IP network/subnets that are being filtered.

Example:

```
list filters
```

input-interchange *neighbor-as-id*

Prints the set of routes that, when learned from AS **neighbor-as-id**, are readvertised by the IGPs.

Example:

```
list input-exchange 32
```

output-interchange *neighbor-as-id*

Prints the set of routes that is advertised to autonomous system *neighbor-as-id* by the EGP protocol.

Example:

```
list output-exchange 32
```

9.4 IP Configuration and Console Commands

packet-filter *name*

Lists all configured packet filters if you do not specify a name. If you specify a name, the router lists the access controls associated with the packet filters that you specify.

Example:

```
list packet-filter pf-in-0
Name              Direction     Interface
pf-in-0           In            0

Access Control is: enabled
List of access control records:
                                                  Beg  End Beg End
    Ty Source      Mask      Destination  Mask    Pro  Pro Prt Prt
1   E  128.185.0.0 FFFF0000  0.0.0.0      00000000 0   255  0  65535
2   I  0.0.0.0     00000000  0.0.0.0      00000000 0   255  0  65535
```

protocols

Prints the configured state of the IP routing protocols (OSPF, RIP, and EGP) along with whether ARP subnet routing is enabled or disabled.

Example:

```
list protocols
```

rip-routes-accept

Prints the set of routes that the RIP routing protocol always accepts. See the IP configuration commands **enable/disable receiving dynamic nets/subnets** for more information.

Example:

```
list rip-routes-accept
```

routes

Prints the list of static network/subnet routes that were configured. Also lists any configured default gateways.

Example:

```
list routes
```

sizes

Displays the routing table size, reassembly buffer size, and the route cache size.

Example:

```
list sizes
```

tags

Displays the per-interface tags that are associated with received RIP information. These tags can be used to group routes together for later readvertisement through EGP where a tag is treated as if it were a route's source AS. (See the IP **add output-interchange** command.) Tags are also propagated by the OSPF routing protocol.

Example:

```
list tags
```

triggered-rip

Displays the Triggered RIP timers and maximum retransmission limits for responses and polling messages for all Triggered RIP interfaces. It also indicates whether call sensitivity is enabled or disabled .

Example:

```
list triggered-rip
```

Move C

Changes the order of the access control list. This command places record number *from#* immediately after record number *to#*. After you move the records, they are immediately renumbered to reflect the new order.

Syntax: move access-control *from# to#*

Example:

```
move access-control 5 2
```

9.4 IP Configuration and Console Commands

Packet-filter M

Use the **packet-filter** command to display information for a specific packet filter. If no name is specified, it displays all packet filters that are configured.

Syntax: packet-filter *name*

Example:

```
packet-filter pf-in-0
Name              Direction     Interface   #Access-Controls
pf-in-0           In            0           2

Access Control currently enabled
Access Control run 8 times, 7 cache hits

List of access control records:
                                                   Beg End  Beg End
  Ty  Source       Mask     Destination  Mask      Pro Pro  Prt Prt  Invoc
0 I   0.0.0.0      00000000 192.67.67.20 00000000  6   6    25  25   0
1 E   150.150.1.0  FFFFFF00 150.150.2.0  00000000  0   255  0   655  0
2 I   0.0.0.0      00000000 0.0.0.0      00000000  89  89   0   655  27
```

Ping M

Has the router send ICMP Echo Requests to a given destination once a second and watch for a response. This command can be used to isolate trouble in an internetwork environment.

This process is done continuously, incrementing the ICMP sequence number with each additional packet. Matching received ICMP Echo responses are reported with their sequence number and the round-trip time. The granularity (time resolution) of the round-trip time calculation is usually (depending on platform) on the order of 20 milliseconds. The **ping** command completes when a character is typed at the console. At that time, a summary of packet loss, round-trip time, and number of ICMP destination unreachables received is displayed.

When a multicast address is given as destination, there may be multiple responses printed for each packet sent, one for each group member. Each returned response is displayed with the source address of the responder.

Note: The size of the ping (number of data bytes in the ICMP message, excluding the ICMP header) is 56 bytes, and the TTL used is 60.

Syntax: ping *interface-address*

Example:

```
ping 128.185.142.11
PING 128.185.142.11: 56 data bytes
64 bytes from 128.185.142.11: icmp_seq=0. time=0. ms
64 bytes from 128.185.142.11: icmp_seq=1. time=0. ms
64 bytes from 128.185.142.11: icmp_seq=2. time=0. ms
64 bytes from 128.185.142.11: icmp_seq=3. time=0. ms
64 bytes from 128.185.142.11: icmp_seq=4. time=0. ms
64 bytes from 128.185.142.11: icmp_seq=5. time=0. ms

----128.185.142.11 PING Statistics----
6 packets transmitted, 6 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 0/0/0
```

Route M

Displays the route (if one exists) to a given IP destination. If a route exists, the IP addresses of the next hops are displayed, along with detailed information concerning the matching routing table entry. (See the IP **dump** command.)

Syntax: route *ip-destination*

Example:

```
route 18.10.0.5
Destination:     18.10.0.5
Mask:            255.0.0.0
Route type:      SPE1
Distance:        3
Age:             1
Next hop(s):     128.185.123.18
```

Example:

```
route 128.185.230.0
Destination:     128.185.230.0
Mask:            255.255.255.0
Route type:      SPF
Distance:        1
Age:             1
Next hop(s):     128.185.230.0
```

Example:

```
route 128.185.232.0
Destination:     128.185.232.0
Mask:            255.255.255.0
Route type:      RIP
Distance:        3
Age:             0
Next hop(s):     128.185.146.4
```

Set C

Sets certain values, routes, and formats within your IP configuration.

Syntax: set

access-control . . .

broadcast-address . . .

cache-size

default network-gateway . . .

default subnet-gateway . . .

egp-system-number . . .

internal-ip-address

ip-host-only-default . . .

originate-rip-default

reassembly-size

router-id . . .

routing table-size . . .

tag. . .

triggered-rip . . .

access-control *on* or *off*

Allows you to configure the router to enable or disable IP access control.

Example:

```
set access-control on
```

broadcast-address *ip-interface-address style fill-pattern*

Specifies the IP broadcast format that the router uses when broadcasting packets out a particular interface. IP broadcasts are most commonly used by the router when sending RIP update packets.

The *style* parameter can take either the value *local-wire* or the value *network*. Local-wire broadcast addresses are either all ones (255.255.255.255) or all zeros (0.0.0.0). Network style broadcasts begin with the network and subnet portion of the *ip-interface-address*.

You can set the *fill-pattern* parameter to either 1 or 0. This indicates whether the rest of the broadcast address (other than the network and subnet portions, if any) is set to all ones or all zeros.

When receiving, the router recognizes all forms of the IP broadcast address.

The example below configures a broadcast address of 255.255.255.255. The second example produces a broadcast address of 192.9.1.0, assuming that the network 192.9.1.0 is not subnetted.

Example:

```
set broadcast-address 192.9.1.11 local-wire 1
set broadcast-address 192.9.1.11 network 0
```

Note: To display the broadcast address setting, issue the **list all** command.

cache-size *entries*

Configures the maximum number of entries for the IP routing cache. Default: 64. Maximum: None.

Example:

```
set cache-size 64
```

Note: To display the cache size setting, issue the **list sizes** command.

default network-gateway *next-hop cost*

Configures a route to the authoritative router (default gateway). Assume that the router's default gateway has more complete routing information than the router itself.

The route is specified by the IP address of the next hop (*next-hop*) and the distance (*cost*) to the default gateway.

All packets having unknown destinations are forwarded to the authoritative router (default gateway).

Example:

```
set default network-gateway 192.9.1.10 10
```

9.4 IP Configuration and Console Commands

default subnet-gateway *subnetted-network next-hop cost*

Configures a route to a subnetted network’s authoritative router (default subnet gateway). You can configure a separate default subnet gateway for each subnetted network.

The IP address of the subnetted network (*subnetted-network*) specifies the network to which the gateway points.

The IP address of the next hop (*next-hop*) and the distance (*cost*) to the default subnet gateway specify the route.

All packets destined for unknown subnets of a known subnetted network are forwarded to the subnetted network’s authoritative router (default subnet gateway).

Example:

```
set default subnet-gateway 128.0.0.0 128.185.123.22 6
```

egp-system-number

Configures the router’s autonomous system number that is used when running the EGP protocol.

Example:

```
set egp-system-number
```

internal-ip-address

Sets the internal IP address that belongs to the router as a whole, and not any particular interface. This address is always reachable regardless of the state of the interface. When the internal IP address and the router ID are set in the same router, the internal IP address has precedence over the router ID. To delete the internal IP address, set the address to 0.0.0.0.

Example:

```
set internal-ip-address 142.82.10.1
```

ip-host-only-default network-gateway-ip-host-only *address cost*

Configures a route to the gateway router which will be used if this router is in host-only mode. All packets destined to any address in the network will be forwarded to this gateway by default, unless there is a suitable subnet gateway defined.

Example:

```
set ip-host-only-default network-gateway-ip-host-only
Default gateway [0.0.0.0]? 2.2.3.4
gateway’s cost [0]? 4
```

ip-host-only-default subnet-gateway-ip-host-only *subnet address cost*

Configures a route to the subnet gateway router which will be used if this router starts up in host-only mode. All packets destined to an address in the specified subnetwork will be forwarded to this gateway.

Example:

```
set ip-host-only-default network-gateway-ip-host-only
For which subnetted network [0.0.0.0]? 12.0.0.0
Default gateway [0.0.0.0]? 12.4.5.6
gateway’s cost [0]? 4
```

originate-rip-default

Configures the conditions under which the router originates a RIP default route, and the cost that will be used when originating the default.

Example:

```
set originate-rip-default
Always originate default route? [No]:
Originate default if EGP/BGP routes available? [No]: yes
   From AS number [0]?
   To network number [0.0.0.0]?
Originate default if OSPF routes available? [No]: yes
Originate default of cost [1]?
```

Always originate default route
Sets the router to always advertise a RIP default route, or not.

Originate default if EGP/BGP routes available
Sets the router to advertise a RIP default route when EGP or BGP routes are available.

This prompt is not displayed if *Always originate default route* is set to Yes.

From AS number
If this is non-zero, it specifies the AS which is the source of a route. If the route from this AS to the given *To network number* is in the routing table, then the router will advertise a RIP default route. If set to zero and the destination number is 0.0.0.0, then the router will advertise the default when any EGP/BGP route is in the routing table.

This prompt is not displayed if *Originate default if EGP/BGP routes available* is set to No.

Range: 0 to 65,535
Default: 0

(A 0 indicates that routes may be learned from any AS.)

To network number — If this is non-zero it specifies the destination address for the route from the *From AS number*. If the route specified by these two fields is in the routing table, then the router will advertise a RIP default route.

Originate default if OSPF routes available — Sets the router to advertise a RIP default route when OSPF routes are available.

This prompt is not displayed if *Always originate default route* is set to Yes.

Originate default of cost — Selects the metric for the RIP default route.

Range: 0 to 16
Default: 1

reassembly-size

Configures the size of the buffers that are used for the reassembly of fragmented IP packets. Default: 12000.

Note: This parameter is relevant to the EGP routing protocol.

Example:

```
set reassembly-size 12000
```

router-id ***ip-address***

Sets the default IP address used by the router when sourcing various kinds of IP traffic. This address is of particular importance in multicasting. For example, the source address in pings (including multicast pings), traceroute, and tftp packets sent by the router are set to the router ID. In addition, the OSPF router ID is set to the configured router ID.

The router ID must match one of the configured IP interface addresses of the router. If not, it is ignored. When ignored, or just not configured, the default IP address of the router (and its OSPF router ID) is set to the first IP address in the router's configuration.

Note: Configuring a router ID may cause the router's OSPF router ID to change. If this happens, link state advertisements originated by the router before the router ID change persist until they age out (possibly as long as 30 minutes). This may cause an increase in link state database size.

Example:

```
set router-id 128.185.120.209
```

Note: To display the router ID setting, issue the **list all** command.

routing table-size *number-of-entries*

Sets the size of the router's IP routing table. The default size is 768 entries. Setting the routing table size too small causes dynamic routing information to be discarded. Setting the routing table size too large wastes router memory resources.

Example:

```
set routing table-size 1000
```

tag

Configures the per-interface tags associated with received RIP information. These tags can be used to group routes together for later readvertisement through EGP where a tag is treated as if it were a route's source AS. (See the IP **add output-interchange** command.) Tags are also propagated by the OSPF routing protocol.

Example:

```
set tag
```

triggered-rip

Configures the per-interface parameters associated with Triggered RIP. You should only use Triggered RIP on point-to-point connections. If the interface is a dial circuit, you may want to enable call sensitivity to reduce the number of connections made to send RIP updates. On nondial connections (including permanent ISDN and V.25 *bis* connections), or if you require rapid distribution of ‘bad news’ updates, disable call sensitivity.

Note: Call sensitivity is implemented as an extension to the Triggered RIP rfc.

Example:

```
set triggered-rip
```

9.4 IP Configuration and Console Commands

Sizes M

Displays the configured sizes of specific IP parameters.

Syntax: sizes

Example:

```
sizes
Routing table size:           768
Table entries used:           3
Reassembly size:              12000
Largest reassembled pkt:       0
Largest EGP update sent:       0
Size of routing cache:        64
# of cache entries in use:     0
```

Routing table size Indicates the configured number of entries that the routing table maintains.

Table entries used Indicates the number entries used from the routing table.

Reassembly size Indicates the configured size of the reassembly buffer that is used to reassemble fragmented IP packets.

Largest reassembled pkt Indicates the largest IP packet that this router reassembled.

Largest EGP update sent Indicates the largest IP update that this router reassembled.

Size of routing cache Indicates the configured size of the routing cache.

# of cache entries in use Indicates the number of tries currently being used from the cache.

Static M

Displays the list of configured static routes. Configured default gateways and default subnet gateways are also listed.

Each static route’s destination is specified by an address-mask pair. Default gateways appear as static routes to destination 0.0.0.0 with mask 0.0.0.0. Default subnet gateways also appear as static routes to the entire IP subnetted network.

The example below shows a configured default gateway, a configured default subnet gateway (assuming 128.185.0.0 is subnetted), and a static route to network 192.9.10.0:

Syntax: static

Example:

```
static
Net             Mask            Cost  Next hop
0.0.0.0         0.0.0.0         1     128.185.123.18
128.185.0.0     255.255.0.0     1     128.185.123.22
192.9.10.0      255.255.255.0   10    128.185.123.22
```

Net Indicates the network address of the route.

Mask Indicates the subnet mask of the IP address.

Cost Indicates the cost of using this route.

Next hop Indicates the next router a packet passes through using this route.

Traceroute M

Displays the entire path to a given destination, hop by hop. For each successive hop, **traceroute** sends out three probes and prints the IP address of the responder, together with the round-trip time associated with the response. If a particular probe receives no response, an asterisk is printed. Each line in the display relates to this set of three probes, with the leftmost number indicating the distance from the router executing the command (in router hops).

The traceroute is done whenever the destination is reached, an ICMP Destination Unreachable is received, or the path length reaches 32 router hops.

When a probe receives an unexpected result, several indications can be printed. “!N” indicates that an ICMP Destination Unreachable (net unreachable) was received. “!H” indicates that an ICMP Destination Unreachable (host unreachable) was received. “!P” indicates that an ICMP Destination Unreachable (protocol unreachable) was received. Since the probe is a UDP packet sent to a strange port, a port unreachable is expected. “!” indicates that the destination was reached, but the reply sent by the destination was received with a TTL of 1. This usually indicates an error in the destination, prevalent in some versions of UNIX. The destination is inserting the probe’s TTL in its replies. This unfortunately leads to a number of lines consisting solely of asterisks before the destination is finally reached.

Syntax: traceroute *interface-address*

Example:

```
traceroute 128.185.142.239
TRACEROUTE 128.185.124.110: 56 data bytes
 1 128.185.142.7 16 ms 0 ms 0 ms
 2 128.185.123.22 16 ms 0 ms 16 ms
 3 * * *
 4 * * *
 5 128.185.124.110 16 ms ! 0 ms ! 0 ms !
```

TRACEROUTE Displays the destination area address and the size of the packet being sent to that address.

1 Displays the first trace showing the destination’s NSAP and the amount of time it took the packet to arrive at the destination. The packet is traced three times.

Destination unreachable Indicates that no route to the destination is available.

*3 * * *
4 * * ** Indicates that the router is expecting some form of response from the destination, but the destination is not responding.

Triggered-rip M

Displays the Triggered RIP status and counters for all Triggered RIP interfaces.

The example below shows an interface configured for Triggered RIP with Call Sensitivity enabled on a dial-on-demand circuit:

Syntax: triggered-rip

Example:

```
triggered-rip
Interface      Current State     Call Sensitivity
   1             Normal              Enabled

     Counter             Value
      Rx Responses         1
      Tx Responses         2
      Rx Flushes           1
      Tx Flushes           1
      Rx Routes            2
      Tx Routes            2
```

```
Rx Requests         2
Tx Requests         2
Rx Acks             3
Tx Acks             2
Response Retrans    0
Response Retrans    1

Rx Bad Message      0
Rx Bad Version      0
Rx Bad Flush        0
Calls Initiated     0
Failures            0
Link Ups            1
Piggy-backed Resp   0
Suboptimal Used     0
Bad-news Resp Blkd  0
```

Interface Indicates the number of the network interface.

Current State The current state of Triggered RIP on the interface. Valid states are:

- **Power-up**. The router has been started but the data link over the interface is not yet up. The state changes to Power-up whenever the interface goes down.

  When the interface comes up, the state changes to Polling, and a Response is sent with the Flush flag set.

- **Polling**. The data link is up and the router is polling Requests.

  When a response with the flush flag set is received, the state changes to Normal. If no response is received before the initial maximum polls limit is reached, the state changes to Failed.

- **Normal**. A response with the flush flag set has been received from the polled router, and responses with routes are being exchanged.

  If the maximum retransmissions of a response limit is reached, the state switches to Polling state.

- **Failed**. No response to polling requests has been received and the Maximum Initial Poll Retransmissions limit has been reached.

  The router continues to send poll requests until the Maximum Failed-State Poll Retransmissions limit is reached. If any Triggered RIP message is received, the state changes to Polling.

Call Sensitivity

Indicates whether *Call Sensitivity* has been **enabled** or **disabled** on this interface. *Call Sensitivity* should be used on dial-on-demand circuits to reduce the number of connections made.

If *Call Sensitivity* is enabled, Triggered RIP will not make a connection to send routing changes that are ‘bad news’ (worse routing metrics), but will wait for the connection to be made for data transfer. If a route fails (bad news) and then recovers to the previous metric, no update will be sent.

Counters

Rx Responses — The number of Update Response packets received.

Tx Responses — The number of Update Response packets sent.

Rx Flushes — The number of Update Response packets received with the Flush flag set.

Tx Flushes — The number of Update Response packets sent with the Flush flag set.

Rx Routes — The number of new routes received.

Tx Routes — The number of new routes sent.

Rx Requests — The number of Update Request packets received.

Tx Requests — The number of Update Request packets sent.

Rx Acks — The number of Acknowledgement packets received.

Tx Acks — The number of Acknowledgement packets sent.

Response Retrans — The number of Response packets that have been retransmitted.

Request Retrans — The number of Update Requests that have been retransmitted.

Rx Bad Message The number of RIP packets received and rejected because they had an unrecognized command field.

Rx Bad Version The number of packets received with an illegal version number in the packet header.

Rx Bad Flush The number of packets received with an illegal value for the Flush flag. The Flush flag may be **1** to indicate that the receiving router should start timing out its routing entries, otherwise, it should be **0**.

Calls Initiated The number of calls that have been made to send routing updates. On a non-dial circuit, this counter will be 0.

Failures The number of times Triggered RIP has gone into Failed state.

Link Ups The number of times the data link has come up.

Piggy-backed Resp The number of times “bad news” routing updates have been “piggy-backed” with data or with “good news” routing updates.

When *Call Sensitivity* is enabled on a dial-on-demand circuit, routing updates will **not** be sent if:

- The routing changes are all “bad news.” That is, the new routing metrics are worse than the existing metrics.
- The dial circuit is currently down.

The router will not initiate a call to pass on such changes. It will wait until a connection is made to send data or it has an update to send that includes “good news” (better routing metrics). The routing changes that have been held back are then sent.

Suboptimal Used The number of times a suboptimal route has been used.

Triggered RIP keeps track of *all* routes in the network that are reported, and will use the route with the best metric (the optimal route). If the optimal route metric goes down, then the router may use a **suboptimal** route with a better metric.

Bad-news Resp Blkd Indicates the number of routing updates that were blocked because the dial-circuit was down and the update contained exclusively “bad news” metrics. This counter is only used if *Call Sensitivity* is enabled.

This counter indicates the number of unnecessary calls that have been saved.

9.4 IP Configuration and Console Commands

Update C

Lets you update information that was originally added with the **add** command.

Syntax: update packet-filter . . .

packet-filter *name*

Use the **update packet-filter** command at the `IP config>` prompt to assign access control entries. The router prompts you for the name of the filter you want to update. The `IP config>` prompt changes to incorporate the packet filter name you provide.

Refer to Section 9.2.10 for more information about how to configure packet filters and the commands available at this prompt.

Example:

```
update packet-filter
Packet-filter name [ ]? pf-1-in
Packet-filter 'test' Config>
```

Exit C M

Returns to the previous prompt level.

Syntax: exit

Example:

```
exit
```

10

Configuring and Monitoring IPX

This chapter describes how to configure and monitor the IPX protocol using the IPX configuration and console commands.

For additional information about the IPX protocol, refer to the *Routing Protocols Reference Guide*.

10.1 Accessing the IPX Configuration Environment

For information on how to access the IPX protocol configuration and console environments, see Chapter 1.

10.2 IPX Configuration and Console Commands

This section explains the IPX commands. The IPX configuration commands specify the network parameters for router interfaces transmitting IPX packets. Enter the configuration commands at the `IPX Config>` prompt. To make your configuration changes effective, you must restart the router.

The IPX console commands allow you to view the parameters and statistics of the interfaces and networks that transmit IPX packets. Enter the console commands at the `IPX>` prompt.

Table 10–1 summarizes the IPX configuration and console commands.

10.2 IPX Configuration and Console Commands

Table 10–1 IPX Configuration and Console Commands Summary

Command	Tasks	Function
? (Help)	Configuring and Monitoring	Lists all of the IPX configuration commands or lists the options associated with specific commands.
Access-controls	Monitoring	Lists the status of IPX access controls, the IPX access control statements, and a count of how many times each control statement was followed.
Add	Configuring	Adds access control for IPX packets, SAP filters, and IP tunnel addresses.
Cache	Monitoring	Lists the current contents of the routing cache.
Configuration	Monitoring	Lists the network numbers of the router interfaces on which IPX is enabled, the IPX frame/ encapsulator types, and the RIP/SAP timer intervals.
Counters	Monitoring	Displays the number of routing errors, destination errors, and packet overflows.
Delete	Configuring	Deletes access control for IPX packets, SAP filters, and IP tunnel addresses.
Disable	Configuring	Disables specific IPX interfaces, IPX over the point-to-point protocol, an IP tunnel, replies to Get Nearest Neighbor requests, NetBIOS filtering for an interface, the router’s response to keepalive packets, or globally disables IPX.
Disable interface	Monitoring	Disables specific IPX interfaces.
Dump routing tables	Monitoring	Displays the contents of the current IPX RIP routing tables.
Enable	Configuring	Enables specific IPX interfaces, IPX over the point-to-point protocol, an IP tunnel, replies to Get Nearest Neighbor requests, NetBIOS filtering for an interface, the router’s response to keepalive packets, or globally enables IPX.
Enable interface	Monitoring	Enables specific IPX interfaces.
Filters	Monitoring	Lists the current SAP filters and the state of each filter.
Frame	Configuring	Specifies the data link format for Ethernet, token ring, and FDDI interfaces.
IPXWAN	Monitoring	Lists configuration information about IPX running over a WAN interface through the point-to-point protocol.
List	Configuring	Displays the current IPX configuration.

Table 10–1 IPX Configuration and Console Commands Summary (Continued)

Command	Tasks	Function
Move	Configuring	Changes the line numbers set when adding access control.
Set	Configuring	Sets the host number, network number, maximum networks, access control, filters, maximum services parameters, ipxwan over PPP, local and remote cache size, router name, RIP/SAP update intervals for networks and tunnels, and node-id.
Shutdown	Monitoring	Used when running the Novell SHUTDOWN certification test to allow time for RIP and SAP packets to be sent out.
Sizes	Monitoring	Displays the configured sizes and contents for the local node and remote network caches.
Slist	Monitoring	Displays the contents of the current IPX SAP routing tables.
Exit	Configuring and Monitoring	Exits the IPX configuration process and returns to the CONFIG environment.

? (Help) C M

Lists the commands that are available from the current prompt level. You can also enter **?** after a specific command name to list its options.

Syntax: ?

Example:

```
?
DISABLE
ENABLE
EXIT
FRAME
LIST
SET
ADD
DELETE
MOVE access-control
```

10.2 IPX Configuration and Console Commands

Access-Controls M

Lists the status of IPX access controls, the IPX access control statements, and a count of how many times each control statement was followed.

Syntax: access-controls

Example:

```
access-controls
Access Control currently enabled
List of IPX Access Control records:
# T Dest Net         Host  Sck  Sck  Src Net         Host  Sck Sck Count
1 E      179 123456789ABC 1234 1234      176 000000000000    0   0     0
2 I        0 000000000000    0  FFF        0 000000000000  453  453  102
3 I        0 000000000000    0  FFF        0 000000000000  452  452   34
```

Add C

Adds an access control entry. This determines whether IPX packets are dropped or forwarded. The **add** command also adds an IPX SAP filter and the IP tunnel address.

Syntax: add

access-control . . .
filter *. . .*
ip-tunnel-address . . .

access-control ***type dest-net dest-host dest-socket-range src-net src-host src-socket-range***

Determines whether to pass a packet at the Internet Datagram Protocol (IDP) level. IPX access controls provide a global access control functionality at the IDP level for the IPX protocol. The access control list is an ordered set of entries. Each entry can be either Inclusive or Exclusive. Each entry has source and destination network numbers, host addresses, and socket ranges.

When a packet is received from a network for the IPX protocol, and IPX access control is enabled, it is checked against the access control list. It is compared with the net/address/socket pairs in the list until there is a match. If there is a match and the entry is of the Inclusive type, reception of the packet (and potential forwarding) proceeds. If the matching entry is of the Exclusive type, the packet is dropped. If there is no match, the packet is dropped.

Note: You must add entries to the address control list before access control can function.

When enabled, access controls apply to all received packets. If you do not enable reception of RIP (socket 453 hexadecimal) or SAP (socket 452 hexadecimal) packets, the IPX forwarder is nonfunctional. Always start with the following entry:

```
add access I 0 0 452 453 0 0 0 FFFF
```

The arguments for this command are as follows:

Type — Identifies whether packets are sent or dropped for a specific address or set of addresses. Enter I for Include. This allows the packets to be sent. Enter E for Exclude. This causes the router to discard the packets.

Dest-net — Network number of the destination. Enter the network number in hexadecimal. Zero (0) means all networks.

Dest-host — Host number on the destination network. Enter the host number in hexadecimal. Zero (0) means all hosts on the network.

Dest-socket- range — Two numbers that specify an inclusive range of destination sockets. Enter two hexadecimal numbers between zero (0) and FFFF.

Src-net — Network number of the source. Enter the network number in hexadecimal. Zero (0) means all networks.

Src-host — Host number on the source network. Enter the host number in hexadecimal. Zero (0) means all hosts on the network.

Src-socket- range — Two numbers that specify an inclusive range of source sockets. Enter two hexadecimal numbers between zero (0) and FFFF.

Note: It is not necessary to use access controls and SAP filters for IPX to work in a NetWare environment. Use them only if necessary.

Syntax: add access-control *type dest-net dest-host dest-socket-range src-net src-host src-socket-range*

Example:

```
add access-control  E 201 1 0 FFFF 329 0 451 451
```

This access control prevents all nodes on network 329 from accessing the file server with internal network number 201.

10.2 IPX Configuration and Console Commands

filter *hops service-type service-name*

Prevents NetWare bindery overflows for users on large networks by enabling you to determine the number of hops reasonable for a given service. IPX SAP filters allow the protocol to be configured to ignore certain entries in SAP advertisements. This is done to limit the size of the SAP database. This may be necessary due to size limitations in older versions of NetWare file servers. This may also be necessary to limit the amount of SAP data sent across WAN links.

The SAP filters are a global ordered list of filter entries. Each filter entry has a maximum hop count, a service type, and an optional service name.

Different servers use unique server type numbers. Server types up to 8000(hexadecimal) are reserved for *well-known* server types. Other types may be defined and used by third party applications. Table 10–2 is a list of common, well-known servers. If you need more information refer to Novell’s documentation.

Table 10–2 Common IPX Servers

Server Type (hexadecimal)	Description	Server Type (hexadecimal)	Description
0000	Unknown	0007	Print Server
0001	User	0008	Archive Queue
0002	User Group	0009	Archive Server
0003	Print Queue	000A	Job Queue
0004	File Server	000B	Administration
0005	Job Server	00023	Async Server
0006	Gateway	0024	Remote Bridge Server

When a SAP response packet is received, each SAP entry is compared with the filter list. If the SAP entry matches an entry in the filter list and is greater than the specified hops, it is ignored and not entered into the local SAP database. If there is no match, the SAP entry is accepted.

The arguments for this command are as follows:

Hops	Specifies maximum number of hops permitted for the service. The range is 0 to FFFF.

Service-type This field indentifies the type of service the server provides. Novell assigns each type of server a unique Server Type. Specify the type of service you wish to filter, in the range 0000 to FFFF(hexadecimal).

Service-name Identifies a particular function provided by a server. In general, this field is not entered. If you do specify a value, enter a 48-character name.

Syntax: add filter *hops service-type service-name*

Example:

```
add filter 2 039B NOTES-CHICAGO
```

This example filters any SAP advertisements for the Lotus Notes server “NOTES-CHICAGO” at more than 2 hops.

ip-tunnel-address *ip-address*

Used to construct the IPX IP address peer list. You can assign one IP unicast address at a time to form the list.

ip-address IP unicast or multicast address that makes up the peer list.

Syntax: add ip-tunnel-address *ip-address*

Example:

```
add ip-tunnel-address ip-address
```

Cache M

Displays the contents of the IPX routing cache.

Syntax: cache

Example:

```
cache
Dest Net/Node        Use Count    via Net/Node         via Int

152                   56000       161/000000000006     SL/0
162                   56476       162/000000000000     Eth/0
162/0000C0239F71      56476       162/0000C0239F71     Eth/0
```

The first entry shows that the remote network 152 can be reached over the serial link with an IPX network number of 161. The second entry is the IPX network 162. It is an Ethernet directly attached to the router. This entry is a general local network entry. There is one general local network entry for each of the directly attached networks after they have begun forwarding IPX packets. The last entry is a local entry on an Ethernet. This IPX cache entry was used to send 56,476 packets to the IPX node number 0000C0239F71 on net number 162.

Configuration M

Lists the network, encapsulation information, and RIP/SAP timer intervals of all the router interfaces on which the IPX protocol is enabled.

Syntax: configuration

Example:

```
configuration
Router Configuration
Net  Name   Type                         Network/Address
  0  Eth/0  SCC Ethernet                  2/08002BB21BFB
  1   SL/0  SCC Serial Line              12/000000000004
  2   SL/1  SCC Serial Line              11/000000000004

IPX Encapsulation/Frame Types
Net  Name   Type                         Encapsulation
  0  Eth/0  SCC Ethernet                  ETHERNET_II
  1   SL/0  SCC Serial Line               N/A
  2   SL/1  SCC Serial Line               N/A

RIP/SAP Timer Intervals
Net  Name   Type                         SAP Interval   RIP Interval
  0  Eth/0  SCC Ethernet                     1            1
  1   SL/0  SCC Serial Line                  1            1
```

Router Configuration

Net Specifies the interface number.

Name Specifies the interface name.

Type Specifies the hardware type of the interface.

Network/Address Specifies the user-assigned network number and host number. Except for serial lines, the host number is the node address of the network interface. For serial lines, it is the user-configured IPX host number.

IPX Encapsulation/ Frame Types Displays current configured encapsulation and frame type information by interface.

Net Specifies the interface number.

Name Specifies the interface name.

Type Specifies the hardware type of the interface.

Encapsulation Displays the encapsulation/frame type configured for a particular interface.

RIP/SAP Timer Intervals Indicates the delay between the transmission of complete RIP and SAP advertisements on an interface.

Net Indicates the interface number.

Name Indicates the interface name.

Type Specifies the hardware type of the interface.

SAP Interval Indicates the number of minutes between complete SAP advertisements on an interface. The range is 1 through 1440. The default is 1.

RIP Interval Indicates the number of minutes between complete RIP advertisements on an interface. The range is 1 through 1440. The default is 1.

10.2 IPX Configuration and Console Commands

Counters M

Displays the number of routing errors, destination errors, and packet overflows that have occurred. In the example, the counters show no recorded errors.

Syntax: counters

Example:

```
counters
Routing errors
Count     Type
   0      Unknown
   0      Checksum error
   0      Destination unreachable
   0      Hop count expired
   0      Interface size exceeded

Destination errors
Count      Type
   0       Unknown
   0       Checksum error
   0       Non-existent socket
   0       Congestion

IPX input packet overflows
    Net     Count
   Eth/0    0
   TKR/0    0
```

Delete C

Deletes an IPX access control entry, SAP filter entry, or IP tunnel address.

Syntax: delete

access-control . . .
filter . . .
ip-tunnel-address . . .

access-control *line#*

Discards the access control statement that matches the line number you enter. Run the **list** command to display the current line numbers.

Syntax: delete access-control *line#*

Example:

```
delete access-control 2
```

filter *hops service-type service-name*

Discards the specified filter statement. You must type the statement exactly as it appears when you run the **list** command. The arguments are as follows:

Syntax: delete filter *hops service-type service-name*

Example:

```
delete filter 2 039B NOTES-CHICAGO
```

Hops Maximum number of hops permitted for the service.

Service-type Numeric service class. Enter a 2-byte number.

Service-name If the entry you are deleting has a name, specify the name.

ip-tunnel-address *ip-address*

Deletes an IP address from the IP address peer list.

Syntax: delete ip-tunnel-address *ip-address*

Example:

```
delete ip-tunnel-address ip-address
```

ip-address IP unicast or multicast address that makes up the peer list

Note: Deleting all addresses from the peer list results in a nonoperational tunnel.

Disable C M

C From the `IPX Config>` prompt, you can disable specific IPX interfaces or globally disable the IPX protocol. You can reenable the interface using the **enable** command.

10.2 IPX Configuration and Console Commands

M Using the monitoring command from the `IPX>` prompt, you can disable only a specific interface (*interface#*) from sending IPX packets, **or globally disable IP tunneling**. You can reenable the specific interface using the **enable** command.

Syntax: disable

interface *interface#*. . .
ipx
ipxwan . . .
ip-tunnel
keepalive . . .
netbios . . .
reply-to-get-nearest-server

interface ***interface#***

Prevents the router from sending IPX packets over specific interfaces.

Example:

```
disable interface 2
```

ipx

Prevents the router from sending IPX packets over any of the interfaces.

Example:

```
disable ipx
```

ipxwan ***interface#***

Prevents IPX from functioning over an interface supporting the point-to-point protocol.

Example:

```
disable ipxwan interface 2
```

ip-tunnel

Disables IPX on the IP network.

Example:

```
disable ip-tunnel
```

keepalive ***interface#***

Prevents the router from responding to keepalive packets on the specified interface.

Example:

```
disable keepalive 3
```

Example:

```
disable keepalive
Which interface [0]?
```

netbios *interface#*

Prevents the router from filtering NetBIOS packets on the specified interface.

Example:

```
disable netbios 3
```

Example:

```
disable netbios
Which interface [0]?
```

reply-to-get-nearest-server

Prevents the router from responding to GET NEAREST SERVER requests from workstations that are attempting to locate a server.

Note: Disable this feature with great caution. Use this command only when there are multiple routers (or servers) on an IPX network and it is known that the "best" server is not behind this router.

Example:

```
disable reply 3
```

Dump Routing Tables M

Displays the contents of the current IPX RIP network routing tables.

Syntax: dump routing tables

Example:

```
dump routing tables
Type  Dest net  Hops  Delay  Age(M:S)     via Router
 Dir      124     0      1       0: 0     124/AA0004001A04
 Dir      131     0      1       0: 0     131/00000000001A
 Dir      177     0      1       0: 0     177/00000000001A
 Dir       41     0      1       0: 0      41/4000C90401FA
 Dir      249     0      1       0: 0     249/0000C9084F34
 RIP      250     1      2       0:10     249/0000C9093250
 RIP 2C39ABE9     2      3       0:10     249/0000C9093250
 RIP       BB     1      2       0:50      41/4000C9050971
 RIP        1     2      3       0:50      41/4000C9050971
 RIP       31     2      3       0:50      41/4000C9050971
 RIP      703     1      2       0:20      41/4000C9041243
 RIP      704     1      2       0:30      41/4000C9041243
```

```
12 route entries used out of 32
12 net entries used out of 32
```

Type Specifies one of the following:

- **Dir** – Specifies that this network is directly connected to the router.
- **RIP** – Specifies that this route was provided by the IPX routing protocol.
- **Del** – Specifies that this route has timed out and is no longer being used. The route is kept in this state for a while to inform other routers that the route is bad. After that time, it is removed and is no longer displayed.

Dest net Specifies the destination network number.

Hops Specifies the number of router hops to this destination.

Delay Specifies the estimate of how long it takes for a packet to be transmitted and arrive at its destination. The unit of delay is the number of IBM PC clock ticks to send a 576-byte packet, which is 18.21 clock ticks per second. The minimum delay is 1 unit.

Age Specifies the age of the routing information in minutes and seconds. If an entry in the routing table is not updated, the router does the following:

- After 3 RIP timer intervals have passed, the router declares the route eligible for replacement and is no longer advertised. The route type is then specified as **DEL**. A discussion on RIP timer intervals can be found in the **set rip-update-interval** configuration command section.
- After an additional 60 seconds, the route is garbage-collected and does not appear in the **dump** display.

via Router Specifies the next hop for packets going to networks that are not directly connected. For directly connected networks, this is the address of the router interface that transmits the packet.

At the bottom of the display is the number of route and network entries used and the total available. If all the network entries are used, it is likely that the routing table is not large enough. Use the IPX configuration **set maximum networks** command to increase the size.

If all of the route entries are used, then there may be routes to IPX networks that cannot be kept, including new, incoming networks. If you do not want to increase the number of available routes, reduce the number of alternate routes per network.

Enable C M

C From the `IPX Config>` prompt, you can enable specific IPX interfaces or globally enable the IPX protocol.

M From the `IPX>` prompt, you can only interactively enable IPX on a specific interface. The interface does not transmit and receive IPX packets unless it has an IPX network number configured and has passed self-test (is up).

Syntax: enable

interface . . .
ipx
ipxwan . . .
ip-tunnel
keepalive . . .
netbios . . .
reply-to-get-nearest-server

interface ***interface#***

Allows the router to send IPX packets over specific interfaces.

Example:

```
enable interface 2
```

ipx

Allows the router to send IPX packets over all of the enabled interfaces.

Example:

```
enable ipx
```

ipxwan ***interface# timeout retry_timer***

Allows the routing of IPX traffic over an interface that supports the point-to-point protocol. This command also queries for a connection timer value and a retry timer value. The **enable** command prompts for the same parameters as the **set ipxwan** command. This allows you to initially set IPXWAN parameters without having to use the **set** command. If you need to modify preconfigured parameters, then use the **set ipxwan** command.

Example:

```
enable ipxwan 0 60 60
```

ip-tunnel ***net# rip_interval sap_interval***

Enables IPX on the IP network.

net#	The hexadecimal form of the network number.
rip_interval	The RIP timer interval in minutes. The range is 1 to 1440. The default is 1.
sap_interval	The SAP timer interval in minutes. The range is 1 to 1440. The default is 1.

Example:

```
enable ip-tunnel 0 60 60
```

keepalive ***interface# timeout***

Allows the router to respond to keepalive packets on the interface specified by *interface#* for the number of minutes specified in *timeout.* The allowable values for *timeout* are from 0 through 1440 minutes (24 hours) with a default of 0 minutes (no timeout). This command is valid only for dial circuits.

Note: You can use a nonzero timeout to try to overcome file server limitations on the number of connections that it allows. Setting a nonzero timeout value can help to terminate idle connections.

Example:

```
enable keepalive 3 60
```

Example:

```
enable keepalive
Which interface [0]?
Timeout (in min) [0]?
```

netbios ***interface#***

Allows the router to filter NetBIOS packets on the specified interface.

Example:

```
enable netbios 3
```

Example:

```
enable netbios
Which interface [0]?
```

reply-to-get-nearest-server

Allows the router to respond to GET NEAREST SERVER requests from workstations that are attempting to locate a server. This is the default setting.

Example:

```
enable reply 2
```

Filters M

Lists the current filters and the state of these filters.

Syntax: filters

Example:

```
filters
IPX SAP Filter currently enabled
List of IPX SAP Filter records:
Count  Max Hops  Type  Service Name
  0           8     4  ?
  0           1  1234  SomeServer
```

Frame C

Specifies the packet format for IPX interfaces. (Encapsulation can also be set using the CONFIG **network** command.)

Note: When there are incorrect or invalid configuration records, the default frame values are used.

Syntax: frame

ethernet_II . . .
ethernet_8022 . . .
ethernet_8023 . . .
ethernet_snap . . .
token-ring msb . . .
token-ring lsb . . .
token-ring_snap msb. . .
token-ring_snap lsb. . .

fddi . . .
fddi_snap . . .

ethernet_*type* ***interface#***

Selects the Ethernet encapsulation format. This is required if you are using NetWare Open VMS on the Ethernet, and is often used when there are ISO nodes on the same Ethernet. The following options are available:

- ethernet_II (default of NetWare 4.0 and greater) – Ethernet_II uses Ethernet version 2.0 protocol 81-37.
- ethernet_8022 – Ethernet_8022 uses Ethernet 802.3 with 802.2 SAP E0.
- ethernet_8023 (default of pre-NetWare 4.0 and lower), router default – Ethernet_8023 uses Ethernet 802.3 without any 802.2 header.
- ethernet_SNAP – Ethernet_SNAP uses 802.3, 802.2 with SNAP PID 00-00-00-81-37.

Note: The ethernet_SNAP encapsulation it is not architecturally valid and is not fast-pathed. No cache entries appear for network entries using this encapsulation.

The default value for Ethernet frames is “ethernet_8023.”

Example:

```
frame ethernet_II 4
```

token-ring_*type interface#*

Selects the Token Ring encapsulation format. The default value is “token-ring MSB.” The following options are available:

- token-ring MSB . . .
- token-ring LSB . . .
- token-ring_SNAP MSB . . .
- token-ring_SNAP LSB . . .

Token-ring MSB uses 802.5, 802.2 SAP E0.

Example:

```
frame token-ring MSB 3
```

Token-ring SNAP uses 802.5, 802.2 with SNAP PID 00-00-00-81-37.

Example:

```
frame token-ring_SNAP MSB 3
```

FDDI_*type interface#*

Selects the FDDI encapsulation format. FDDI_SNAP is the default value for FDDI interfaces. The following options are available:

- FDDI *interface#*
- FDDI_SNAP *interface#*

FDDI uses 802.2 SAP E0.

Example:

```
frame FDDI 4
```

FDDI_SNAP uses 802.2 SNAP PID 00-00-00-81-37.

Example:

```
frame FDDI_SNAP 4
```

IPXWAN M

Lists the current configuration information for IPX running over a WAN interface through the point-to-point protocol.

Syntax: ipxwan

detailed . . .

summary . . .

detailed *intrfc#*

Lists the complete current configuration information for IPX running over a WAN interface through the point-to-point protocol.

Example:

```
ipxwan detailed 2
Detailed information for IPXWAN link over interface 2, PPP/1
This side is the IPXWAN slave
Neighbor Name: SKYSURF2
Neighbor Node ID: 727299
Negotiated Routing Type: RIP/SAP
Link Delay: 330 1/18th sec ticks
Common Net#: 132
```

```
Connection Timeouts: 0
Connection Retries: 0
Timer Requests Sent: 1
Timer Requests Received: 1
Timer Responses Sent: 1
Timer Responses Received: 0
Info Requests Sent: 0
Info Requests Received: 1
Info Responses Sent: 1
Info Responses Received: 0
```

Neighbor Name
Router name of the neighbor as received in the RIP/SAP Information Request Packet.

Neighbor Node ID
Node ID (also known as the primary network number) of the neighbor. This is an IPX network number unique to the entire internetwork. It is a 32-bit quantity.

Negotiated Routing Type
Negotiated routing type. Digital currently supports RIP/SAP. The default is RIP/SAP.

Link Delay
Link delay in 1/18th second ticks calculated by the master. It is a 16-bit quantity. It is always calculated, therefore, there is no default.

Common Net#
Network number agreed upon by both ends of the link. This number must be unique to the entire internetwork. It is a 32-bit quantity. There is no default, it must be negotiated.

Connection Timeouts
Number of times the connection timed out. A connection times out periodically if the exchange of IPXWAN packets does not proceed. The timeout period is configurable. The default for the timeout period is 60 seconds.

Connection Retries
Number of times the connection is retried after timing out. The amount of time to wait after timing out before retrying is configurable. It defaults to 60 seconds.

Timer Requests Sent
Number of IPXWAN Timer Request packets sent.

Timer Requests Received
Number of IPXWAN Timer Request packets received.

Timer Responses Sent
Number of IPXWAN Timer Response packets sent.

Timer Responses Received
Number of IPXWAN Timer Response packets received.

Info Requests Sent Number of IPXWAN Information Request packets sent.

Info Requests Received Number of IPXWAN Information Request packets received.

Info Responses Sent Number of IPXWAN Information Response packets sent.

Info Responses Received Number of IPXWAN Information Response packets received.

summary

Lists a summary of the current configuration information for IPX running over WAN interfaces through the point-to-point protocol.

Example:

```
ipxwan summary
Net  Name   Common Net#  NodeID  Neighbor Name
6    PPP/1  132           727299  SKYSURF2
```

Net Network interface number.

Name Network interface name.

Common Net# Network number agreed upon by both ends of the link. This number must be unique to the entire internetwork.

NodeID Node ID (also known as the primary network number) of the neighbor. This is an IPX network number unique to the entire internetwork.

Neighbor Name Router name of the neighbor as received in the RIP/SAP Information Request Packet.

10.2 IPX Configuration and Console Commands

List C

Displays the current IPX configuration.

Syntax: list

Example:

```
list
IPX globally                enabled
Host number (serial line)   000000000000
Router Name (IPXWAN)
NodeID (IPXWAN)             0
Maximum networks            32
Maximum total alt. route entries        128
Maximum alt. routes per dest. network   3
Maximum services                        32
Maximum Network Cache entries           64
Maximum Local Cache entries             64


List of configured interfaces:
                Frame
Ifc  IPX net #  Encapsulation       SAP nearest server reply   IPXWAN

0    177        FDDI_SNAP              Enabled                 N/A
1    183        TOKEN-RING      MSB    Enabled                 N/A
2    184        TOKEN-RING      MSB    Enabled                 N/A

RIP/SAP Timer Intervals
Net  IPX net #        SAP Interval(Minutes)   RIP Interval(Minutes)
  0       177                  1                       1
  1       183                  1                       1
  2       184                  1                       1
IPX SAP Filter is: disabled
No IPX SAP Filter records in configuration.
IPX Access Controls are: disabled
No IPX Access Control records in configuration.

NetBIOS Filtering Configuration
Net   IPX net #         NetBIOS Filtering
  3         177         Enabled
  4         178         Disable
  5         179         Enabled

Keepalive Configuration
Net   IPX net #         Keepalive                 Timeout(Minutes)
  3         177         Enabled                            60
  4         178         Disabled                           60
  5         179         Enabled                          None
```

10.2 IPX Configuration and Console Commands

IPX globally Indicates if IPX is globally enabled or disabled.

Host number The host number assigned to IPX. You can change this number with the IPX **set** command.

Router name The user-assigned router name for IPXWAN.

NodeID The user-assigned node ID for IPXWAN.

Maximum networks The size of the IPX RIP routing table, which is the maximum number of IPX networks.

Maximum routes The number of configured maximum routes, which is the maximum number of routes to IPX networks.

Maximum routes-per-network The number of configured maximum routes-per-network.

Maximum services The size of the IPX SAP service table.

Maximum Network Cache entries The number of network cache entries.

Maximum Local Cache entries The number of local cache entries.

List of configured interfaces Lists each interface number and its associated IPX network number. It also displays the type of encapsulation enabled for that interface, whether the “get nearest server” feature is enabled, and which interface has IPX enabled for WAN traffic (IPXWAN).

IPX SAP filter Indicates whether the IPX SAP filter function is enabled or disabled and whether there are any IPX SAP filters in the configuration.

IPX Access Controls Indicates whether the IPX access controls are enabled or disabled and whether there are access control records in the configuration.

RIP/SAP Timer Intervals The delay between the transmission of complete RIP and SAP advertisements on an interface.

IPX Net # The IPX network number designated for a particular interface.

10.2 IPX Configuration and Console Commands

SAP Interval (Minutes) The number of minutes between complete SAP advertisements on an interface. The range is 1 through 1440 (24 hours). The default is 1.

RIP Interval (Minutes The number of minutes between complete RIP advertisements on an interface. The range is 1 through 1440 (24 hours). The default is 1.

Move C

Changes the line numbers for the access controls. After you move the lines, they are renumbered to reflect the new order.

Syntax: move access-control *line# line#*

Example:

```
move 5 2
Are you sure this is what you want to do (Yes or [No]):
```

Set C

Sets the IPX configuration parameters listed below.

Syntax: set

access-control . . .
filter . . .
host-number . . .
ip-tunnel-net-number . . .
ipxwan . . .
keepalive . . .
local-cache size. . .
maximum alternate-routes-per-destination . . .
maximum networks . . .
maximum services . . .
maximum total-alternate-route-entries . . .
name . . .
net-number . . .
node-id . . .
remote-cache size
rip-ip-tunnel-update-interval . . .

10.2 IPX Configuration and Console Commands

sap-ip-tunnel-update-interval . . .
rip-update-interval . . .
sap-update-interval . . .
split-horizon . . .

access-control ***toggle***

Turns the access controls globally on or off. Enter **on** or **off** as the toggle value.

Example:

```
set access-control on
```

filter ***toggle***

Turns the IPX SAP filters globally on or off. Enter **on** or **off** as the toggle value.

Example:

```
set filter on
```

host-number ***host#***

Specifies the host number used for serial interfaces running IPX. Each IPX router operating over serial lines must have a unique host number. This is required because the serial lines do not have hardware node addresses from which to build a host number. The host number is a 12-digit hexadecimal number. This number must be unique on each router.

Example:

```
set host-number 0000000000F4
```

ip-tunnel-net-number ***ipx-net#***

Specifies the IPX network number used for the IP-tunnel. The IP-tunnel must have been previously enabled before you can execute this command. The *ipx-net#* is the IPX net number in 8 hexadecimal digits (1 to FFFFFFFF).

Example:

```
set ip-tunnel-net-number 100
```

10.2 IPX Configuration and Console Commands

ipxwan ***interface# timeout retry_timer***

Sets or modifies an interface to support the routing of IPX traffic over a PPP interface that supports the point-to-point protocol. Before the **set ipxwan** command can be invoked, IPXWAN must be enabled on the interface using the **enable ipxwan** command. This command also queries for a connection timer value and a retry timer value.

Example:

```
set ipxwan 0 60 60
```

keepalive ***interface# timeout***

Sets or modifies an interface specified by *interface#* to respond to keepalive packets for the number of minutes specified in *timeout.* The allowable values for *timeout* are from 0 through 1440 minutes (24 hours) with a default of 0 minutes (no timeout). This command is valid only for dial circuits.

Note: You can use a nonzero timeout to try to overcome file server limitations on the number of connections that it allows. Setting a nonzero timeout value can help to terminate idle connections.

Example:

```
set keepalive 3 60
```

Example:

```
set keepalive
Which interface [0]?
Timeout (in min) [0]?
```

local-cache size #

Specifies the size of the routing table local cache. The range is 1 to 10000. The default size is 32. For a description of local and remote cache, refer to the *Routing Protocols Reference Guide*.

Example:

```
set local-cache size 64
```

maximum alternate-routes-per-destination #

Specifies the number of alternate routes that you want to assign to a given destination network. The range is 1 to 64. The default value is 3.

Example:

```
set maximum alternate-routes-per-destination 8
```

maximum networks #

Specifies the size of IPX's RIP routing table. This reflects the number of networks in the IPX internet on which the router operates. The range of values is 1 to 2048. The default is 32.

Example:

```
set maximum networks 30
```

maximum services #

Specifies the size of IPX's SAP service table. This reflects the number of services (such as file servers or SNA gateways) on the IPX internet on which the router operates. The range is 1 to 2048. The default is 32.

Example:

```
set maximum services 30
```

maximum total-alternate-route-entries #

Specifies the total number of entries available for alternate routes.

The range is 1 to 4096. The default is 32.

Example:

```
set maximum total-alternate-route-entries 40
```

name

Lets you assign a symbolic name to the router. IPXWAN requires that a router have a primary network number and a name. The name can be from 1 to 47 characters in length and can contain the characters “A” through “Z,” underscore (_), hyphen (-), and the “at” sign (@).

Example:

```
set name newyork_accounting
```

net-number ***interface# ipx-net#***

Assigns an IPX network number to the associated directly connected network. Every IPX interface must have a unique network number. The only exception is that serial lines can be assigned network numbers of zero. (Serial lines without network numbers do not pass IPX NetBIOS emulation packets.) The interface number is decimal and the net number is a hexadecimal number. The IPX net number is 8 digits in hexadecimal format (1 to FFFFFFFF).

Example:

```
set net-number 2 180
```

10.2 IPX Configuration and Console Commands

node-id ***primary-net#***

Lets you assign a primary network number. IPXWAN requires a router to have a primary network number and a name. The “node-id” is the primary network number for the router and must be assigned before the exchange of IPXWAN packets can begin. The *primary-net#* must be a 1- to 8-digit hexadecimal number.

This number is for the router as a whole. In NetWare file server terms, it is the “internal” network number. This number must be unique among all the network numbers in the IPX internet.

Example:

```
set node-id 500
```

remote-cache size ***#***

Specifies the size of the routing table remote cache. The range is 1 to 10000. The default size is 32. For a description of local and remote cache, refer to the *Routing Protocols Reference Guide*.

Example:

```
set remote-cache size 64
New IPX remote network cache size [32]?
```

rip-ip-tunnel-update-interval ***interface# delay#***

Specifies the time delay in minutes between transmissions of complete RIP updates through an IP tunnel on an interface. The range is 1 through 1440. The default is 1. In the following example, the RIP IP tunnel interval on interface 0 is being set to 2 minutes.

Example:

```
set rip-ip-tunnel-update-interval 0 2
```

rip-update-interval ***interface# delay#***

Specifies the time delay in minutes between transmissions of complete RIP updates on an interface. The range is 1 through 1440. The default is 1. In the following example the RIP interval on interface 0 is being set to 2 minutes.

Example:

```
set rip-update-interval 0 2
```

sap-ip-tunnel-update-interval ***interface# delay#***

Specifies the time delay in minutes between transmissions of complete SAP updates through an IP tunnel on an interface. The range is 1 through 1440. The default is 1. In the following example, the SAP IP tunnel interval on interface 0 is being set to 2 minutes.

Example:

```
set sap-ip-tunnel-update-interval 0 2
```

sap-update-interval ***interface# delay#***

Specifies the time delay in minutes between transmissions of complete SAP updates on an interface. The range is 1 through 1440. The default is 1. In the following example the SAP interval on interface 0 is being set to 2 minutes.

Example:

```
set sap-update-interval 0 2
```

split-horizon ***disabled*** or ***enabled*** or ***heuristic***

Allows adjustment of the split-horizon behavior of IPX on Frame Relay interfaces, as needed. Otherwise, it is recommended not to adjust this value. Setting the value to *disabled* causes all routes to be advertised on that interface. Setting the value to *enabled* causes routes not to be advertised. If the value is set to *heuristic* (default), and the interface is not Frame Relay, split-horizon is enabled. If the interface is Frame Relay and is sparsely connected, split-horizon is disabled; otherwise, split-horizon is enabled.

Example:

```
set split-horizon disabled
```

Shutdown M

Used when running the Novell SHUTDOWN certification test to cause an orderly shutdown of IPX functions on the router. The **disable** *interface#* command should not be used for this test.

Syntax: shutdown

10.2 IPX Configuration and Console Commands

Sizes M

Lists the current size and number of entries in use for the local node and remote network caches.

Syntax: sizes

Example:

```
sizes
Current IPX cache size:
Remote network cache size(max entries): 64
       2 entries now in use

Local node cache size(max entries): 128
       1 entries now in use
```

Slist M

Displays the contents of the current IPX SAP tables. This command is similar to the NetWare **slist** command.

Syntax: slist

Example:

```
slist
State Typ Service Name Hops Age(M:S)  Net/   Host     /Sock
 SAP 0004 PCS12         3   0:50        1/000000000048/0451
 SAP 0004 ACMPCS        3   0:50        1/00000000004A/0451
 SAP 0004 DEVEL2        1   0:50       11/0000000000B4/0451
 SAP 0004 PLANNING      2   0:50       BB/0000000000B7/0451
 SAP 0004 DEVEL         2   0:50       BB/0000000000EE/0451
 SAP 0004 SOFT2         1   0:30      704/000000000094/0451
 SAP 0004 SKYSURF1      2   0: 5 2C39ABE9/000000000001/0451
 SAP 0278 DIRTREE       2   0: 5 2C39ABE9/000000000001/4005
 SAP 026B DIRTREE       2   0: 5 2C39ABE9/000000000001/0045


9 services used out of 32
```

State Indicates one of the following:

- **SAP** – This service was provided by the SAP protocol.
- **DEL** – This service has timed out and is no longer being used. The service is kept in this state for 5 to 10 seconds to inform other routers that the service is bad. After that time, it is removed and no longer displayed.

Typ The server type in hexadecimal. File servers are type 0004. Other type numbers are assigned by Novell. (See Table 10–2).

Service name The server’s unique name for this type of server. Only the first 30 characters of the 48-character name are printed to conserve space.

Hops The number of router hops from this router to the server.

Age Specifies the age of the service information. If an entry in the SAP table is not updated, the router does the following:

- After 3 SAP timer intervals have passed, the service is no longer used, but is broadcast as DEAD. The service state is then specified as **DEL**. A discussion on SAP timer intervals can be found in the **set sap-update-interval** configuration command section.
- After an addition 60 seconds, the service is garbage-collected and does not appear in the **slist** display.

Net/Host/Sock Specifies the address of the service. The address includes:

- Network number
- Net host number (the address of the first interface on the network)
- Socket number at which the service can be reached

At the bottom of the display is the number of entries used and the total available. If all the entries are used, it is likely that the service table is not large enough. Use the IPX configuration **set maximum services** command to increase the size.

Exit C M

Returns to the previous prompt level.

Syntax: exit

Example:

```
exit
```

11

Configuring and Monitoring OSI/DNA V

This chapter describes the OSI configuration and console commands.

Note: When operating DNA IV networks together with DNA V networks, all DNA IV configuring and monitoring must be done from the DNA IV `NCP>` configuration process. For information about configuring DNA IV, refer to Chapter 7. For more information about DNA IV and DNA V compatibility, refer to the *Routing Protocols Reference Guide*. From this point on, the use of the term “OSI” refers to both the OSI and DNA V environments unless otherwise indicated.

For more information about OSI and DNA V, refer to the *Routing Protocols Reference Guide*.

11.1 Accessing the OSI Configuration Environment

For information about accessing the OSI configuration environment, see Chapter 1.

11.2 Basic Configuration Procedure

This section outlines the minimum configuration steps that you are required to perform to get the OSI/DNA V protocol up and running over a LAN (Ethernet, Token Ring, and FDDI), serial lines, X.25 packet-switching networks, and Frame Relay. Before beginning any configuration procedure, use the **list device** command from the **config** process to list the interface numbers of the different devices. If you desire any further configuration command explanations, refer to the configuration commands described in this chapter.

Note: You must restart the router for new configuration changes to take effect.

11.3 Configuring OSI over an Ethernet, Token Ring, or FDDI LAN

Do the following basic configuration procedure before beginning the specialized procedures described in the following sections:

- **Setting the Network Entity Title (NET)**. If you are using OSI routing, you must set the router's NET using the **set network-entity-title** command. The NET consists of the router's system ID and its area address. Use the **list globals** command to verify that the NET is configured correctly.

Note: If the router is being configured for both DNA IV and OSI/DNA V, then do *not* configure the NET. You must configure the Phase IV prefix using the **set phaseivpfx** command and the NET will be set automatically.

- **Globally enabling OSI**. Enable the OSI software to run on the router using the **enable OSI** command. Use the **list globals** command to verify that the OSI protocol is enabled.
- **Setting up DEC-mode**. If the router uses X.25 routing circuits compatible with other DNA routers you must set the router to DEC-mode using theNCP command **define executor type**. The **add prefix-address** command will use extended format for all new prefix addresses when DEC-mode is used.

11.3 Configuring OSI over an Ethernet, Token Ring, or FDDI LAN

To configure the OSI protocol to run over an Ethernet, Token Ring, or FDDI LAN, use the **set subnet** command. There is a one-to-one correspondence between subnetworks and interfaces. Use the **set subnet** command to configure all LAN subnets (Ethernet, Token Ring, and FDDI). Use the default multicast addresses for Ethernet and FDDI. When configuring a Token Ring subnet, use the addresses listed in Table 11–1. Use the **list subnet detailed** or **list subnet summary** commands to verify that you have configured the subnets correctly.

Table 11–1 Functional Addresses for Token Ring

Parameter	Functional Address 802.5
All ESs [09002B000004]	C00000004000
All ISs [09002B000005]	C00000008000
All L1 ISs [0180C2000014]	C00000008000
All L2 ISs [0180C2000015]	C00000008000

11.4 Configuring OSI over X.25 or Frame Relay

- **In DEC-mode**, to configure the OSI protocol to run over the X.25 interface, do the following:
  - Set the subnet. Use the **set subnet** command to set the interface to X.25. Use the defaults for all the required information. Use the **list subnet detailed** or **list subnet summary** commands to verify that you configured the subnets correctly.
  - Add the routing circuit. Use the **add routing circuit** command to establish a DECnet-compatible X.25 circuit.
- **Otherwise**, to configure the OSI protocol to run over the X.25 or Frame Relay interface, do the following:
  - Set the subnet. Use the **set subnet** command to set the interface to X.25 or FRL (Frame Relay). Use the defaults for all the required information. Use the **list subnet detailed** or **list subnet summary** commands to verify that you configured the subnets correctly.
  - Set the virtual circuit. Use the **set virtual-circuit** command to establish a virtual circuit between the router and X.25 PSN (Packet Switching Node) or the frame relay switch.

Note: The router prompts you for a DTE address. For frame relay, enter the DLCI (Data Link Control Identifier) number. For X.25, enter the PSN’s DTE address.

11.5 Enabling Compression on DLM and DA Circuits

To configure data compression over a DA or DLM circuit, use the **add template** command and set the Compression option to *enable*. You must also enable compression on the X.25 network interface. Refer to the *Network Interface Operations Guide* for details about configuring the X.25 network interface.

11.6 Configuring OSI over a Serial Line

To configure the OSI protocol to run over a serial line, use the **set subnet** command to set the interface to SL (serial line). In this context a Serial Line may be a PPP leased line, but it could also include any of the variations of PPP (for example PPP over Frame Relay) and dial circuits (V.25 *bis* or ISDN).

11.7 Configuring a DNA V Router for a DNA IV Environment

There is a one-to-one correspondence between subnetworks and interfaces. Use the defaults for all the required information. Use the **list subnet detailed** or the **list subnet summary** command to verify that you configured the subnets correctly.

11.7 Configuring a DNA V Router for a DNA IV Environment

When configuring a DNA V router, it may be necessary to configure an interface to run in a DNA IV environment. For example, the router is attaching to both a DNA V and DNA IV network, or a DNA IV ES is attached to a DNA V router.

Before beginning the steps below, use the appropriate preceding section to configure OSI over a LAN, X.25, Serial Line, or Frame Relay.

1. Disable OSI. Use the **disable osi** command at the `OSI Config>` prompt.
2. Enter the DN configuration process. Exit `OSI Config>` and enter `NCP>`. Use the **protocol DN** command at the `Config>` prompt.
3. Define the global DNA address. Use the **define executor address** command to configure the DNA node and area number of the router.
4. Globally enable DNA. Use the **define executor state** command to enable the DNA protocol to run on the router. If you are using DECnet-compatible X.25 circuits, define the executor type as **DEC-area** if this is a level 2 router or **DEC-routing-iv** if it is a level 1 router.
5. Enable inter-area routing. If the L2 routing algorithm is distance vector at level 2, use the **define executor type area** command to ensure that this router can exchange DNA IV level 2 routing information.
6. Enable the DNA IV circuit. Enable the circuit that the router uses to exchange the routing information. Use the **define circuit** *type* **state on** command.
7. Enter the OSI configuration process. Exit `NCP>` and enter `OSI Config>`. Use the **protocol osi** command at the `Config>` prompt.
8. Set the Phase IV prefix using the **set phaseivpfx** command. The system uses the prefix to set the NET to the correct default value compatible with the DNA IV address.
9. Enable OSI. Use the **enable osi** command at the `OSI Config>` prompt.
10. Restart the router for the changes to take effect.

11.8 DNA IV and DNA V Algorithm Considerations

DNA IV uses a distance-vector routing algorithm. DNA V can use either a distance-vector or a link-state routing algorithm. The algorithm is selected according to what is enabled and disabled, and combinations that can result from these two protocols.

- **DNA IV disabled. OSI/DNA V enabled.** This combination is considered a pure OSI/DNA V environment and the algorithm is automatically set to link-state at both levels 1 and 2 regardless of how the **set algorithm** command is configured.
- **DNA IV enabled. OSI/DNA V disabled.** This combination is considered a pure DNA IV environment and the algorithm is set automatically to distance-vector regardless of how the **set algorithm** command is configured.
- **DNA IV enabled. OSI/DNA V enabled.** This a mixed environment, and the algorithm information is configured and read out of SRAM. Use the **set algorithm** command to configure this information into SRAM.

11.9 OSI Configuration and Console Commands

This section summarizes and explains the OSI configuration and console commands.

The OSI configuration commands allow you to create or modify an OSI configuration. Enter all the OSI configuration commands following the `OSI Config>` prompt. Defaults for any command and its parameters are enclosed in brackets immediately following the prompt.

The console commands allow you to monitor the parameters and statistics of the current OSI configuration. The monitoring capabilities include the following: configured parameters such as the router’s NET and area address, the routing algorithm used, and various protocol statistics. Enter the OSI console commands following the `OSI>` prompt.

Table 11–2 summarizes the OSI configuration and console commands.

11.9 OSI Configuration and Console Commands

Table 11–2 OSI Configuring and Console Commands Summary

Command	Tasks	Function
? (Help)	Configuring and Monitoring	Lists the configuration or monitoring commands, or lists any parameters associated with a command.
Add	Configuring	Adds areas this node supports, receive-passwords for authentication purposes, permitted neighbors for routing security, prefix addresses for other domains, summary addresses, aliases, templates, filters, and X.25 circuits.
Addresses	Monitoring	Displays the router's NET and area addresses.
Change	Configuring	Modifies a prefix address, summary address, permitted neighbor, filter or template.
Change Metric	Monitoring	Modifies the cost of a circuit.
Clear	Configuring	Clears a receive password, transmit password, or SRAM.
CLNP-Stats	Monitoring	Displays OSI CNLP statistics.
Delete	Configuring	Deletes areas, virtual circuits, permitted neighbors, prefix-addresses, summary addresses, adjacencies, aliases, subnets, and X.25 routing circuit parameters.
Designated-router	Monitoring	Displays the designated router for the LAN.
Disable	Configuring	Disables a subnet, the OSI protocol, or an X.25 routing circuit.
DNAV-Info	Monitoring	Displays the routing algorithm currently running on the router.
Enable	Configuring	Enables a subnet, the OSI protocol, or an X.25 routing circuit.
ES-adjacencies	Monitoring	Displays all the End System (ES) adjacencies that are either configured or were learned through the ES-IS protocol.
ES-IS-Stats	Monitoring	Displays all the Intermediate System (IS) adjacencies that are either configured or were learned through the IS-IS protocol or DNA IV
IS-adjacencies	Monitoring	Displays all the IS adjacencies in the adjacency database learned through the IS-IS protocol.
IS-IS-Stats	Monitoring	Displays statistics associated with the IS-IS protocol.
L1-routes	Monitoring	Displays all the L1 routes in the level 1 database.
L2-routes	Monitoring	Displays all the L2 routes in the level 2 database.

Table 11–2 OSI Configuring and Console Commands Summary (Continued)

Command	Tasks	Function
L1-summary	Monitoring	Displays a summary of the level 1 link state database.
L2-summary	Monitoring	Displays a summary of the level 2 link state database.
L1-update	Monitoring	Displays the information contained in L1 link state update packet.
L2-update	Monitoring	Displays the information contained in L2 link state update packet.Displays the information contained in L2 link state update packet.
List	Configuring	Displays the current configuration of adjacencies, aliases, virtual circuits, permitted neighbors, prefix-addresses, summary addresses, subnets, algorithm, phaseivpfx, global information, X.25 routing circuits, filters, or templates.
Route	Monitoring	Displays the route a packet takes to a specified destination.
Send echo packet	Monitoring	Encodes an echo request message in the CLNP packet.
Set	Configuring	Configures the properties associated with OSI parameters (switches, globals, NETs, timers, subnets, transmit-password, prefix-addresses, adjacencies, virtual circuits, algorithm, and phaseivpfx).
Show routing circuits	Monitoring	Displays the state of user-defined routing circuits for the specified interface.
Subnets	Monitoring	Displays the state of all operational subnets.
Toggle	Monitoring	Enables or disables the NSAP alias substitution function.
Traceroute	Monitoring	Displays the route a packet travels to its destination.
Virtual-circuits	Monitoring	Displays information about all X.25 virtual circuits.
Exit	Configuring and Monitoring	Exits the OSI configuration or console prompt and returns to the previous environment.

11.9 OSI Configuration and Console Commands

? (Help) C M

Lists the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example:

```
?
LIST
ADD
CHANGE
DELETE
ENABLE
DISABLE
SET
CLEAR
EXIT
```

Example:

```
list ?
GLOBALS
SUBNETS
VIRTUAL-CIRCUIT
ADJACENCIES
PREFIX-ADDRESSES
ALIAS
TIMERS
ALGORITHM
PHASEIVPFX
INTEGRATED-ISIS
SUMMARY-ADDRESSES
ROUTING-CIRCUIT
PERMITTED-NEIGHBOR
TEMPLATE
FILTER
```

11.9 OSI Configuration and Console Commands

Add C

Configures area and prefix addresses, receive passwords, and address aliases.

Syntax: add

alias
area . . .
filter
permitted-neighbor
prefix-address
receive-password
routing-circuit
summary-address
template

alias

Adds an ASCII string that designates a particular area address or system ID. The ASCII string can be a-z, A-Z, 0-9, and a few other characters including the hyphen (-), comma (,), and underscore (_). Do not use escape characters.

The offset indicates the position, in semi-octets (nibbles), where the ASCII string begins within the address (aliases used for system IDs have an offset of 1). The maximum allowable alias is 15 characters, but the alias string must be two characters shorter than the number of hexadecimal digits in the segment it is designating (there are two hexadecimal digits per byte). Otherwise you get an error message:

Example:

```
add alias
Alias []?AliasNameTooLong
Segment []?AA0004000134
Offset [1]?
Maximum ALIAS length for a SEGMENT length of 6 bytes is 10 characters
```

Note: When using an alias input, you must surround it with brackets. For example: **`l1_update  47[newname]99999000012341234.`**

Example:

```
add alias
Alias []?
Segment []?
Offset [1]?
```

Alias The character string you want to use.

Segment The NSAP segment that the alias is replacing.

Offset The location of the alias (in 4-bit, semi-octets) within the NSAP. The offset is determined from the beginning (left) of the NSAP as it is displayed on the console.

area ***area-addr***

Adds area addresses (18-byte maximum) that the node supports. An L1 node that supports other areas considers those synonymous areas. One area address is the area portion of the configured NET. If you try to add a duplicate area address, the router displays an error message.

Example:

```
add area 47000580999999000012341234
```

Note: When adding synonymous areas to an L1 node, use the **set globals** command to configure the maximum number of synonymous areas allowed for this node. All routers within an area must use the same maximum number of synonymous areas. Adjacencies cannot be established if they are different.

If the router will operate in a mixed DNA IV OSI/DNA V network, the area address is derived automatically If DNA IV routing is enabled, you cannot add a DNA IV compatible area address (that is, an area address with the last four digits less than 0040, hexadecimal) as this would result in the router being in multiple Phase IV areas concurrently.

Example:

```
add area 49003F
Can’t add phase IV compatible area address
```

filter ***filter-name***

Adds parameters by which the router bases its acceptance of incoming X.25 calls on a routing circuit, either a static incoming circuit or a DA circuit.

Example:

```
add filter
Filter Name []?
Routing Circuit Name []?
DTE Address[]?
Call UserData (OSI/DEC/USER)?
Priority (1-10) [5]?
```

Filter Name Name you give the filter.

Routing Circuit Name Name of the routing circuit with which the filter is associated.

DTE Address Address of the calling router.

The local router checks the DTE address of an incoming call against a prioritized list of filters for all circuits.

Priority A higher filter *priority* means that a connection to that filter’s calling DTE address is made first. You are recommended to assign a higher priority to filters for static circuits than for DA circuits. This can prevent an incoming static call from *calling-DTE* being assigned a DA circuit.

Call UserData There are three types:

- **USER –** Prompts for an additional entry of up to 16 octets. Enter text to constrain the acceptance of incoming calls. The *call-userdata* field of the incoming call must match this entry.
- **OSI** – Causes the router to automatically configure an ISO protocol discriminator for call-userdata and requires the call to be from an OSI node.
- **DEC** – Causes the router to automatically supply the expected call-userdata defined for DECnet Phase IV DLM (static) circuits.

permitted-neighbor

Adds a verifier (password) for system-level security. Any system that tries to use this routing circuit to communicate with your system must supply this verifier. Create a PERMITTED NEIGHBOR entry for each system from which you want to accept connections.

Example:

```
add permitted-neighbor
Permitted Neighbor Entry Name []?
System ID []?
Verifier []?
```

Permitted Neighbor Entry Name — Name you give the permitted neighbor entry. The name you use should be based on the name of the remote system concerned.

System ID — Specifies the system ID of the remote system that will have to supply a verifier.

Verifier — Value of the verifier as a hexadecimal octet string. There must be an even number of digits in the verifier. An example verifier value is 12AB.

Note: If the remote system is a DECnet Phase IV system, it specifies the verifier as a text string. In this case, you convert each letter in the text string to its hexadecimal value in ASCII, and enter this as the value.

For example, the text string SECRET becomes 534543524554.

prefix-address

Adds static routes to destinations external to the IS-IS domain. This parameter prompts you for different information depending on the type of subnet (X.25, SL, LAN, or FRL) that was configured using the **set subnet** command.

There are two formats for the prefix address—a short format, compatible with earlier versions of the routing software, and an extended format. All prefix addresses must be the same format.

If the router is operating in DEC-mode, then this command will use the extended format, otherwise you may choose the format the first time that you add a prefix address. If you use the extended format you can specify the circuit type as Inbound or Outbound.

If no Address Prefix is entered, the default prefix is assumed.

Example:

```
add prefix-address
```

LAN Subnet:

```
Interface Number [0]?
Subnet Type [LAN]?
Address Prefix []?
MAC Address []?
Default Metric [20]?
Metric Type [Internal]?
State [ON]?
```

X.25 Subnet:

```
Interface Number [0]?
Subnet Type [X.25]?
Routing Circuit []?
Address Prefix [Name]?
Type [OUTBOUND]?
Metric [20]?
Metric Type [Internal]?
Mapping Type [X121]?
Data Format [5(V)]?
State [ON]?
```

Serial Line Subnet:

```
Interface Number [0]?
Subnet Type [SL]?
Address Prefix []?
Default Metric [20]?
Metric Type [Internal]?
State [ON]?
```

Frame Relay Subnet:

```
Interface Number [0]?
Subnet Type [FRL]?
Address Prefix []?
DTE Address []?
Default Metric [20]?
Metric Type [Internal]?
State [ON]?
```

Note: If the subnet does not exist, you receive this error message:

```
Subnet does not exist - cannot define a reachable
address.
```

Interface Number Defines the interface over which the address is reached.

Subnet Type Specifies the type of subnet on the interface. The default value depends on the current configuration of the interface.

Routing Circuit Name Specifies the name of the DA circuit that you want to activate if the adjacency of the static route is not active. This prompt appears only if the router is in DEC-mode. (See the **define executor type** command in Chapter 7 for more information about the router modes).

Address Prefix Defines the NSAP prefix (20 bytes maximum).

Type Specifies whether the address prefix is Inbound or Outbound. This prompt appears only if you are using extended format prefix addresses.

- **INBOUND** - the address prefix corresponds to a Phase IV area that is reachable through this node and circuit by inbound traffic.
- **OUTBOUND** - the address prefix is in an external domain that is reachable over this circuit by outbound traffic.

MAC Address Defines the destination MAC address. You must specify this address if the interface corresponds to a LAN subnet. This prompt only appears if the interface is connected to a LAN subnet.

Mapping Type Defines how the destination physical address is determined, manual or X.121.

- If manual, the protocol prompts for the DTE address.
- If X.121, the protocol does not prompt you for the DTE address. The DTE address in this instance is extracted from the NSAP.

DTE Address Defines the destination DTE address. You must specify this address if the interface is X.25 and the mapping type is manual.

This prompt only appears if the interface is configured for X.25 and the mapping type is manual.

Default Metric — Defines the cost of the address.

Metric Type — Defines whether the metric cost is used for external (**E**) routing or internal (**I**) routing.

Data Format — Specifies whether the data format of the circuit is DECnet Phase V or DECnet Phase IV. Enter 5 for Phase V or 4 for Phase IV.

This prompt appears only if you are using extended format prefix addresses with *Type* set to OUTBOUND.

State — When set to ON, this prefix-address is advertised to other L2 routers. When set to OFF, this is a non-functional prefix address.

receive-password

Adds an ASCII character string (16-character maximum) that authenticates all incoming packets. An incoming packet whose password matches one of the set of receive-passwords, is processed through the IS. Any incoming packets whose passwords do not match are dropped.

Example:

```
add receive-password
Password type [Domain]?
Password []?
Reenter password:
```

Note: To use IS-IS authentication for the receive password, you must enable IS-IS authentication with the **set switches on** command. You receive an error message if you use an invalid password type.

Password type — Designates one of the two types of passwords: Domain or Area.

- Domain passwords are used with L2 LSPs (Level 2, Link State Packets) and SNPs (Sequence Number PDUs).
- Area passwords are used with L1 LSPs and SNPs.

Password — Designates the character string that you are using for authentication. Maximum allowable string is 16 characters.

11.9 OSI Configuration and Console Commands

routing-circuit

Adds a communications channel for X.25 switched virtual circuits (SVCs) that the routing layer uses to send and receive data. You can specify one of these types of routing circuit:

- A *static-in circuit* handles incoming X.25 calls. A call filter (see **add filter**) specifies the criteria that the router uses to accept or reject incoming calls on the circuit.
- A *static-out circuit* initiates outgoing X.25 calls. The router uses a call template (see **add template**) to make outgoing calls.
- A *DA circuit* can have multiple SVCs running simultaneously. Unlike static circuits, the router uses a DA circuit when user traffic arrives at the router to create an adjacency if it is not active at the time. The router terminates the adjacency when no data flow occurs during a period set by the routing circuit idle timer.

When data arrives at an interface and the next hop for forwarding the data has an active adjacency, the router forwards the data on that adjacency.

When data arrives and the next hop has no active adjacency, but the next hop is defined in the reachable address table (the static routing table), the router creates an adjacency to the next hop. It does this by initiating a call request to establish a circuit as defined by a DA routing circuit.

When data arrives and the next hop has no active adjacency and is not defined in the reachable address table, the router discards the packet because it cannot forward the data.

Example:

```
add routing-circuit
 Interface Number [0]?
 Circuit Name []?
 Circuit Type (STATIC/DA) [STATIC]?
 Circuit Direction (OUT/IN) [OUT]?
```

If you select **STATIC** and **OUT**, the following prompts appear:

```
     Recall Timer (0-65535) [60]?
     Max Call Attempts (0-255) [10]?
     Initial Min Timer (1-65535) [55]?
     Enable IS-IS [YES]?
     Level 2 only [NO]?
     External Domain [NO]?
     Default Metric [20]?
     ISIS Hello Timer [3]?
     Enable DECnetV Link Initialization [YES]?
     Modify Receive Verifier (YES/NO) [NO]?
     Modify Transmit Verifier (YES/NO) [NO]?
     Explicit Receive Verification (TRUE/FALSE) [TRUE]?
```

If you select **STATIC** and **IN**, the following prompts appear:

```
    Initial Min Timer (1-65535) [55]?
    Enable IS-IS [YES]?
    Level 2 only [NO]?
    External Domain [NO]?
    Default Metric [20]?
    ISIS Hello Timer [3]?

    Enable DECnetV Link Initialization [YES]?
    Modify Receive Verifier (YES/NO) [NO]?
    Modify Transmit Verifier (YES/NO) [NO]?
    Explicit Receive Verification (TRUE/FALSE) [TRUE]?
```

If you select **DA** for the circuit type, the following additional prompts appear:

```
    Recall Timer (0-65535) [60]?
    Reserve Timer (1-65536) [600]?
    Idle Timer (1-65536) [30]?
    Max SVCs (1-65535) [1]?
```

Interface Number Specifies the X.25 interface for this routing circuit.

Circuit Name Sets up the alphanumeric name of this routing circuit record.

Circuit Type Specifies whether this routing circuit is STATIC or DA.

Circuit Direction Specifies whether the SVC of the static circuit is established with an incoming call request or an outgoing call request.

In both cases, the SVC is initially established upon operator action, but the circuit is not fully enabled until both ends of the circuit have initialized successfully.

11.9 OSI Configuration and Console Commands

Recall Timer — Defines the time in seconds that an out-static circuit or a DA circuit must wait before attempting a new call request. This is a result of the initial call request failing or a subsequent call having been cleared.

Max Call Attempts — If a call request fails, *Max Call Attempts* defines the maximum number of subsequent call requests that the out-static circuit attempts. At this point, a call failure is logged and operator intervention is required to activate the out-static circuit.

Initial Min Timer — Specifies the amount of time in seconds an out-static circuit waits for a link to be initialized (reception of either an ESH or an ISH) after the call request has been accepted. If *Initial Min Timer* expires before the link is fully initialized, the SVC is cleared and an event generated that indicates initialization failure.

Enable IS-IS — Defines whether the IS-IS protocol is enabled on this routing circuit.

Level 2 only — Specifies if this routing circuit is used for level 2 routing only.

External Domain — Specifies whether the router transmits and receives messages to and from a domain outside its IS-IS routing domain.

Default Metric — Defines the cost of this address.

ISIS Hello Timer — Defines the time interval between transmission of IS-IS hellos.

Enable DECnetV Link Initialization — Defines whether DEC-style link initialization for this circuit is enabled.

Modify Receive Verifier — Specifies verification data to be checked against on receiving an XID when verifying by circuit.

Modify Transmit Verifier — Specifies verification data to be included in the XID.

Explicit Receive Verification — Defines whether verification is by circuit or by system. TRUE specifies verification by circuit, and FALSE specifies by system.

Reserve Timer Defines the time after the idle timer expires during which the router still considers a remote node on a DA circuit as active. The router can forward data on the DA circuit until the reserve timer expires.

Idle Timer Defines the length of time a DA adjacency can be idle (no data transmission) before it is cleared.

Max SVCs Defines the maximum number of SVC adjacencies supported by this DA circuit. If no call can be placed because the maximum SVC adjacencies has been reached, then an event “Exceed Max SVC adjacencies” is generated.

summary-address

Creates an IP summary address to reduce network traffic by using summary routes. To do this, you configure your network so that each IP subnet is in a different DECnet V/OSI area, and set a summary address routing entry on each level 2 router. In this way, only summary IP information is sent between areas.

All the level 2 routers in an area must be configured with the same set of summary addresses.

Note: You can only set up summary IP addresses on the router if it is configured to run integrated IS-IS and link state routing at level 2.

Example:

```
add summary-address
Summary Address [0.0.0.0] ? 17.133.0.0
Address Mask [255.0.0.0] ? 255.255.0.0
Default Metric [1]?
```

Summary Address IP address used to summarize a range of subnets.

In this example, all subnets in the range 17.133.0.1 to 17.133.255.254 will be abbreviated to subnet 17.133.0.0

Address Mask IP subnet mask of the summary address.

Default Metric The value used for the metric when abbreviating the route at level 2.

11.9 OSI Configuration and Console Commands

template *template-name*

Creates a template by which the router makes outgoing calls for an X.25 static-out routing circuit and for X.25 DA routing circuits. Templates for static-out and DA circuits are analogous to filters for static-in and DA circuits (see **add filter**).

Example:

```
add template
Template Name []?
Routing Circuit Name []?
DTE Address[]?
Call Data (OSI/DEC/USER)?
Compression [DISABLED]?
```

Template Name Name you give the template.

Routing Circuit Name Name of the routing circuit with which the template is associated.

DTE Address Address for the remote router of up to 14 digits.

Call Data Must match the call-userdata set up for a filter on the remote circuit. There are three types of call-userdata:

- **USER** – Prompts for an additional entry of up to 16 octets. Enter text to match the user data of the appropriate filter on a remote router.
- **OSI** – Causes the router to automatically configure an ISO protocol discriminator for the call data and requires the call to go to an OSI router.
- **DEC** – Specifies that the circuit will use the standard call user data compatible with DECnet Phase IV DLM circuits.

Compression Set Stac LZS data compression to **enable** or **disable**. Compression is disabled by default.

11.9 OSI Configuration and Console Commands

Addresses M

Lists the router's NET and the area addresses configured for this router.

Syntax: addresses

Example:

```
addresses
Network Entity Title:
4700-0500-01  000-9310-04F0
Area Addresses:
4700-0500-01
4900-02
```

Network Entity Title Identifies the router. The NET consists of an area address and a system ID.

Area Addresses Indicates addresses within the routing domain. The router can have a maximum of three area addresses configured at any one time.

Change C

Allows you to modify OSI/DNA V records in the permanent database.

Syntax: change

filter
permitted-neighbor
prefix-address
routing-circuit
summary-address
template

11.9 OSI Configuration and Console Commands

filter ***filter-name***

Changes the values for routing circuit filter parameters. See the **add filter** command for more information on these parameters.

Example:

```
change filter
Filter Name [current-value]?
DTE Address [current-value]?
Call UserData (OSI/DEC/USER) [current-value]?
```

If you select **user**, the following prompts appear:

```
(max 16 octets) [current-value]?
Priority (1-10) [current-value]?
```

permitted-neighbor

Changes the permitted neighbor verifier for system-level security. See the **add permitted-neighbor** command for more information on these parameters.

Example:

```
change permitted-neighbor
Permitted Neighbor Entry Name []?
System ID [current-value]?
Verifier [current-value]?
```

prefix-address

Changes the address data for subnets. The router prompts you for different information depending on the type of subnet that you configured using the **set subnet** command.

Example:

```
change prefix-address
```

LAN Subnet:

```
Interface Number [0]?
Address Prefix []?
MAC Address []?
Default Metric [20]?
Metric Type [Internal]?
State [ON]?
```

X.25 Subnet:

```
Interface Number [0]?
Routing Circuit Name []?
Address Prefix []?
Type [OUTBOUND]?
Default Metric [20]?
Metric Type [INTERNAL]?
Mapping Type [Manual]?
DTE Address []?
Data Format [5(V)]?
State [ON]?
```

Serial Line Subnet:

```
Interface Number [0]?
Address Prefix []?
Mapping Type [Manual]?
Default Metric [20]?
Metric Type (Internal or external)[Internal]?
State [ON]?
```

Frame Relay Subnet:

```
Interface Number [0]?
Address Prefix []?
DTE Address []?
Default Metric [20]?
Metric Type [Internal]?
State [ON]?
```

Interface Number Indicates the interface over which the address is reached.

Routing Circuit Name Specifies the name of the DA circuit that you want to activate if the adjacency of the static route is not active.

This prompt appears only if the router is in DEC-mode. (See the **define executor type** command in Chapter 7 for more information about the router modes).

11.9 OSI Configuration and Console Commands

Address Prefix Indicates the destination NSAP prefix (20 bytes maximum).

Type Specifies whether the address prefix is Inbound or Outbound. This prompt appears only if you are using extended format prefix addresses.

- **INBOUND** – The address prefix corresponds to a Phase IV area that is reachable through this node and circuit by inbound traffic.
- **OUTBOUND** – The address prefix is in an external domain that is reachable over this circuit by outbound traffic.

MAC Address Indicates the destination MAC address. You must specify this address if the interface corresponds to a LAN subnet. This prompt only appears if the interface is connected to a LAN subnet.

Mapping Type Indicates how the destination physical address is determined, manual or X.121.

- If manual, the protocol prompts you for the DTE address.
- If X.121, the protocol does not prompt you for the DTE address. The DTE address in this instance is extracted from the NSAP.

Default Metric Indicates the cost of the address.

Metric Type Indicates whether the metric cost is used for external (**E**) routing or internal (**I**) routing.

DTE Address Defines the destination DTE address. You must specify this address if the interface is X.25 and the mapping type is manual. This prompt only appears if the interface is configured for X.25 and the mapping type is manual.

Data Format Specifies whether the data format of the circuit is DECnet Phase V or DECnet Phase IV. Enter 5 for Phase V or 4 for Phase IV.

This prompt appears only if you are using extended format prefix addresses with *Type* set to OUTBOUND.

State When set to ON, this address receives packets. When set to OFF, this is a nonfunctional address.

routing-circuit ***routing-circuit-name***

Changes the configuration for a routing circuit. The parameters displayed vary, depending on the circuit's configuration. The following example is for an outgoing static circuit. See the **add routing-circuit** command for more information on these parameters.

Example:

```
change routing-circuit
Circuit Name [current-value]?
Recall Timer (0-65535) [current-value]?
Max Call Attempts (0-255) [current-value]?
Initial Min Timer (1-65535) [current-value]?
Enable ES-IS [current-value]?
Enable IS-IS [current-value]?
Level 2 only [current-value]?
External Domain [current-value]?
Default Metric {current-value]?
ESIS IS Hello Timer [current-value]?
ISIS Hello Timer [current-value]?
Enable DECnetV Link Initialization [current-value]?
Modify Receive Verifier (YES/NO) [current-value]?
Modify Transmit Verifier (YES/NO) [current-value]?
Explicit Receive Verification (TRUE/FALSE) [current-value]?
```

summary-address

Changes the summary address information. See the **add summary-address** command for more information on these parameters.

Example:

```
change summary-address
Summary Address [0.0.0.0]?
Address Mask [255.0.0.0]?
Default Metric [1]?
```

template ***template-name***

Changes the values of the template for static-out and dynamically assigned (DA) routing circuits. See the **add template** command for more information on these parameters.

Example:

```
change template
Template Name [current-value]?
DTE Address [current-value]?
Call UserData (OSI/DEC/USER) [current-value]?
 (max 16 octets) [current-value]?
Compression [DISABLED]?
```

The `(max 16 octets) [current-value]?` prompt only appears if you select **user** for Call UserData.

Change Metric M

Modifies the cost of the circuit.

Syntax: change metric

Example:

```
change metric
Circuit [0]?
New Cost [0]?
```

Circuit Indicates the circuit number that you want to change.

New Cost Indicates the new cost of the circuit. Range: 1 to 63.

Clear C

Erases SRAM, or removes the receive or transmit password.

Syntax: clear

receive-password
sram
transmit-password

receive-password

Removes all of the receive passwords previously configured using the **add receive-password** command.

Note: You receive an error message if you use an invalid password type.

Example:

```
clear receive
Password Type [Domain]?
```

Password Type Specifies the type of password being used: Domain or Area. Refer to the **add receive-password** command for descriptions of these passwords.

SRAM

Warning: Using this parameter erases the OSI configuration from SRAM. Use this command only if you intend to erase the configuration.

Example:

```
clear sram
Warning:  All OSI SRAM Information is erased .
Do you want to continue? (Y/N) [N]?
```

transmit-password

Removes the transmit password previously configured using the **set transmit-password** command. The output for this parameter is the same as that of the receive-password parameter.

Note: You receive an error message if you use an invalid password type.

Example:

```
clear password transmit
Password Type [Domain]?
```

11.9 OSI Configuration and Console Commands

CLNP-Stats M

Displays the OSI Connectionless Layer Network Protocol (CLNP) information.

Syntax: clnp-statistics

Example:

```
clnp-statistics
Received incomplete packet                      0
Received packet with bad NSAP length            0
Received packet with bad checksum               0
Received packet with bad version number         0
Received packet with bad type                   0
Received packet with expired lifetime           0
Received packet with bad option                 0
Received packet with unknown destination        0
Received packet with no segmentation permitted  0
Received data packet cannot be forwarded        0
No buffer available to send error packet        0
No route to send error packet                   0
Received OK CLNP packet                         0
Cannot forward error packet                     0
IS0 unknown initial protocol ID                 0
Received error packet                           0
Received local data packet                      0
Sent error packet                               0
```

Received incomplete packet Indicates that a data packet fragment recognized as an ISO CLNP data packet was received.

Received packet with bad NSAP length Indicates that an ISO CLNP data packet was received with an illegal NSAP length.

Received packet with bad checksum Indicates that an ISO CLNP data packet was received with a bad checksum.

Received packet with bad version number Indicates that an ISO CLNP data packet was received with an incorrect or unsupported version number.

Received packet with bad type Indicates that an ISO CLNP data packet was received with an incorrect or unsupported type field.

Received packet with expired lifetime Indicates that an ISO CLNP data packet was received with an expired lifetime.

Received packet with bad option Indicates that an ISO CLNP data packet was received with a bad optional parameter.

Received packet with unknown destination Indicates that an ISO CLNP data packet was received but was not routed. The routing table contains no entry for the destination.

Received packet with no segmentation permitted Indicates that an ISO CLNP data packet was received that needed segmentation. The segmentation permitted flag was not set.

Received data packet cannot be forwarded Indicates that an ISO CLNP data packet was received but was not routed because of a handler error.

No buffer available to send error packet An attempt to send an ISO CLNP error packet failed because of a lack of system I/O buffers.

No route to send error packet An attempt to send an ISO CLNP error packet failed because it was not routed.

Received OK CLNP packet Indicates that an ISO CLNP data packet was received and passed error checking.

Cannot forward error packet Indicates that an ISO CLNP error packet was not routed because of a handler error.

ISO unknown initial protocol ID Indicates that an ISO CLNP packet was received with an unknown or unsupported initial protocol identifier.

Received error packet Indicates that an ISO CLNP error packet was received for this router.

Received local data packet Indicates that an ISO CLNP data packet was received with the destination NSAP indicating one of the router’s NSAPs.

Sent error packet Indicates that ISO CLNP error packet was sent on receipt of a bad packet.

11.9 OSI Configuration and Console Commands

Delete C

Removes parameters previously configured using the **set** or **add** commands.

Syntax: delete

adjacency
alias
area . . .
filter
permitted-neighbor
prefix-address
routing-circuit
subnet
summary-address
template
virtual-circuit

adjacency

Removes a statically configured ES adjacency previously configured with the **set adjacency** command.

Example:

```
delete adjacency
Interface Number [0]?
Area Address []?
System ID []?
```

Interface number Indicates the interface where the adjacency is located.

Area address Indicates the area address of the adjacency.

System ID Indicates the portion of the NET that identifies the adjacency within the area.

11.9 OSI Configuration and Console Commands

alias

Removes the ASCII string that designates a portion of an area address or system ID.

Example:

```
delete alias
ALIAS []?
```

area *address*

Removes the area address (*address*) previously configured with the **add area** command.

Example:

```
delete area 47000580999999000012341234
```

filter *filter-name*

Removes a filter record from the permanent database.

Example:

```
delete filter dlm-1
```

permitted-neighbor *permitted-neighbor-entry-name*

Removes a permitted neighbor previously configured with the **add permitted-neighbor** command.

Example:

```
delete permitted-neighbor
Permitted Neighbor Entry Name []?
```

prefix-address

Removes the prefix address previously configured with the **set prefix-address** command.

Example:

```
delete prefix-address
Interface Number [0]?
Address Prefix []?
```

Interface Number	Indicates the interface number over which the prefix address is configured.
Address Prefix	Indicates the destination NSAP prefix.

11.9 OSI Configuration and Console Commands

routing-circuit ***routing-circuit-name***

Removes from the permanent database an X.25 routing circuit that was established with **add routing-circuit**.

Example:

```
delete routing-circuit p_system2
```

subnet ***intfc#***

Removes a subnet that was previously configured with the **set subnet** command. *Intfc#* indicates the interface number of the configured subnet.

Example:

```
delete subnet 1
```

summary-address

Removes a summary address that was previously configured with the **add summary-address** command.

Example:

```
delete summary-address
Summary Address [0.0.0.0]?
```

template ***template-name***

Removes from the permanent database the template for a routing circuit that you created using the **add template** command.

Example:

```
delete template gates2
```

virtual-circuit

Removes an X. 25, SVC, or a Frame Relay virtual circuit that was previously configured with the **set virtual-circuit** command.

Example:

```
delete virtual-circuit
Interface number [0]?
DTE address[]?
```

Interface number	Indicates the interface number over which the virtual circuit is configured.
DTE address	Indicates the DTE address of the X.25 network to which you are connecting or the DLCI of Frame Relay network to which you are connecting.

11.9 OSI Configuration and Console Commands

Designated-router M

Displays the designated router for the LAN subnets that are physically attached to this router and actively running IS-IS.

Syntax: designated-router

Example:

```
designated-router
Designated Router Information:
Hdw    Int#     Circ    L1DR                   L2DR
Eth/3    4       2      0000-0000-0025-02      0000-0000-0025-02
Eth/1    2       1      0000-0000-0025-01      0000-0000-0025-01
```

Hdw Indicates the type and instance of LAN attached to this router.

Int# Indicates the interface number of this router that attaches to the LAN.

Circ Indicates the circuit number assigned by the router.

L1DR Indicates the LAN ID of the designated router. If the use of alias is enabled, this command displays the alias of the particular segment. The LAN ID is the designated router's system ID concatenated with a 1-byte locally assigned circuit ID.

L2DR Description is the same as L1DR described above.

Note: If the designated router was not elected yet, `Not Elected` is displayed instead of a LAN ID.

Disable C

Selectively disables the integrated IS-IS protocol, the OSI protocol, a routing circuit, or an OSI subnet, previously enabled using the **enable** command.

Syntax: disable

integrated-isis . . .
osi
routing-circuit
subnet . . .

11.9 OSI Configuration and Console Commands

integrated-isis

Disables the integrated IS-IS protocol on the router.

Example:

```
disable integrated-isis
```

osi

Disables the OSI protocol on the router.

Example:

```
disable osi
```

routing-circuit ***routing-circuit-name***

Disables the specified routing circuit. Use **add routing-circuit** to set up routing circuits.

Example:

```
disable routing-circuit p_system2
```

subnet ***interface#***

Disables the OSI protocol on the specified subnet (*interface#*).

Example:

```
disable subnet 0
```

DNAV-info M

Displays the routing algorithm that is currently running on the router.

Syntax: dnav-info

Example:

```
dnav-info
DNA V Level 1 Routing Algorithm: Distance-vector
DNA V Level 2 Routing algorithm: Distance-vector
```

Note: Depending on whether or not DNA IV is enabled or disabled, the routing algorithm displayed here may differ from what is configured in SRAM using the `OSI Config>` **set algorithm** command.

If DNA IV is enabled, the routing algorithm is the one configured in SRAM. If DNA IV is disabled, the routing algorithm is set to link state and may differ from that set in SRAM.

Enable C

Enables the integrated IS-IS protocol, the OSI protocol, a routing circuit, or an OSI subnet.

Syntax: enable

integrated-isis
osi
routing-circuit
subnet . . .

integrated-isis

Enables the integrated IS-IS protocol on the router.

Example:

```
enable integrated-isis
Import RIP Routes [OFF]?
Import EGP Routes [OFF]?
Import BGP Routes [OFF]?
Compare Other IP Routes To Metric Type [EXTERNAL]?
Originate Default Route [OFF]? on
Originated Default Route Metric Type [EXTERNAL]?
Originated Default Route Metric [20]?
```

Import RIP Routes Determines whether to import routes learned via RIP into IS-IS.
Values: ON or OFF
Default: OFF

Import EGP Routes Determines whether to import routes learned via EGP into IS-IS.
Values: ON or OFF
Default: OFF

Import BGP Routes Determines whether to import routes learned via BGP into IS-IS.
Values: ON or OFF
Default: OFF

Compare Other IP Routes To Metric Type
Determines whether routes that have been learned via RIP, EGP, or BGP are treated as equivalent to IS-IS internal or external routes. If you select internal, then imported routes will have a higher preference than IS-IS external routes.
Values: INTERNAL or EXTERNAL
Default: EXTERNAL

Originate Default Route
Determines whether to announce default IS-IS routes.
Value: ON or OFF
Default: OFF

Originated Default Route Metric Type
If the Originate Default Route is set ON, then this determines whether the default route is announced as internal or external.
Value: INTERNAL or EXTERNAL
Default: EXTERNAL

Originated Default Route Metric
If the Originate Default Route is set ON, then this value is used as the metric for the default route.
Range: 1 to 63
Default: Initially 20, otherwise, it is the current value

osi

Enables the OSI protocol on the router.

Note: If DNA IV is not enabled, then OSI is enabled using the configured NET. If DNA IV is enabled and the Phase IV prefix is configured, then the NET defaults to the DNA IV compatible value. Otherwise, an error message indicates the corrective action.

Example:

```
enable osi
```

DNA IV routing enabled and valid Phase IV prefix configured:

```
Phase IV routing is enabled
Defaulting NET to Phase IV compatible [471212000D:AA0004000134]
```

DNA IV routing enabled and DNA IV compatible area addresses defined:

```
Phase IV routing is enabled
Delete phase IV compatible area address before enabling OSI
```

DNA IV routing enabled but Phase IV prefix not configured:

```
Can’t enable OSI until a Phase IV prefix has been configured
```

routing-circuit ***routing-circuit-name***

Enables the specified routing circuit. Use **add routing-circuit** to set up routing circuits.

Example:

```
enable routing-circuit p_system2
```

subnet ***interface#***

Enables the OSI protocol on the specified subnet (*interface#*).

Example:

```
enable subnet 0
```

ES-Adjacencies M

Displays all the end system (ES) adjacencies that are either configured or were learned through the ES-IS protocol.

Syntax: es-adjacencies

Example:

```
es-adjacencies
End System Adjacencies
System ID        MAC Address       Interface     Lifetime     Type
6666-6666-6666   1234-FEAA-041C        0         50           DNAIV
0000-9310-0040   4221-FEAA-03B2        1         static       MNUAL
AA00-0400-0C04   AA00-0400-0C04        1         128          OSI
```

System ID Provides the system ID of the ES adjacency.

MAC Address Indicates the MAC address of the ES on the subnet.

Note: For X.25 subnets, this is the DTE address of the adjacent node over the X.25 routing circuit.

Interface Indicates the router's interface number where the ES adjacency was learned.

Lifetime Indicates the amount of time (in seconds) that the router has left before the information received in the last ES Hello message is discarded.

In the case of static or a manually configured ES-Adjacency, this field reads "*Static*."

Type — Indicates the type of ES adjacency, OSI, DNAIV, DNAIV’, and MANUAL for statically configured adjacencies.

ES-IS-Stats M

Displays the statistics for the ES-IS protocol.

Syntax: es-is-stats

Example:

```
es-is-stats
ESIS input queue overflow                   0
Received incomplete packet                  0
Received packet with bad checksum           0
Received packet with bad version            0
Received packet with bad type               0
No iob available to send hello              0
Cannot send hello due to packet handler error  0
Sent hello                                  3672
Received packet with bad header             0
Received hello with bad nsap                0
Received hello packet with bad option       0
Received hello                              0
Received hello with unsupported domain source  0
No resources to install route               0
Received hello with conflicting route       0

Timed out route reactivated                 0
No resources to send redirect               0
Redirect not sent - handler error           0
Sent redirect                               0
Timed out route                             0
```

ESIS input queue overflow — The ES-IS packet was dropped because a task input queue overflowed.

Received incomplete packet — A packet fragment recognized as an ES-IS packet was received.

Received packet with bad checksum — An ES-IS packet with a bad checksum was received.

Received packet with bad version — An ES-IS packet with a bad or unsupported version was received.

Received packet with bad type
An ES-IS packet with a bad or unsupported type field was received.

No iob available to send hello
An attempt to send an ES-IS hello failed because of a lack of system I/O buffers.

Cannot send hello due to packet handler error
An ES-IS hello was not sent because of a handler error.

Sent hello
An ES-IS hello was sent through an interface.

Received packet with bad header
An ES-IS hello packet with a bad holding time or received field was received.

Received hello with bad nsap
An ES-IS hello packet with a bad NSAP or one that overflowed the field was received.

Received hello packet with bad option
An ES-IS CLNP data packet was received with a bad option parameter.

Received hello
An ES-IS hello packet was received on the interface.

Received hello with unsupported domain source
An ES-IS hello packet was received from an unspecified domain source.

No resources to install route
An ES-IS hello packet was received, but there were no resources to install the route.

Received hello with conflicting route
An ES-IS hello packet was received but was not entered into the database. A previously defined static or dynamic route in the database was conflicting with the route in the hello.

Timed out route reactivated
An ES-IS hello packet with a previously timed out route was received.

No resources to send redirect
An ES-IS redirect packet was not sent because of a lack of resources.

Redirect not sent - handler error
An ES-IS redirect packet was not sent because of a handler error.

Sent redirect
An ES-IS redirect packet was sent out the interface.

Timed out route
An ES-IS hello route has timed out.

11.9 OSI Configuration and Console Commands

IS-Adjacencies M

Lists all the IS adjacencies that are learned through the IS-IS protocol.

Syntax: is-adjacencies

Example:

```
is-adjacencies
End System Adjacencies
System ID       MAC Address    Int  Level  Usage   State   Life   Type
0000-9310-04C8  AA00-0400-EF04  0     L1   L1/L2   DOWN            OSI
0000-9310-04C8  AA00-0400-EF04  0     L2   L1/L2   DOWN          DNAIV
AA00-0400-0504  AA00-0400-0504  1     L2     L2     UP     5390    OSI
```

System ID Provides the system ID of the IS adjacency.

MAC Address Indicates the MAC Address of the IS adjacency.

Int Indicates the router’s interface number that connects to the IS adjacency.

Level For LANs, this indicates the neighbor system level from type of hello message: L1 or L2. For point-to-point, this indicates the neighbor system type: L1 only, otherwise L2.

Usage Indicates from the hello packet circuit type: L1 only, L2 only, or L1 and L2.

State Indicates the operational state of the IS adjacency: up or down.

Life Indicates the amount of time (in seconds) before discarding the last IS Hello message.

Type Indicates the routing protocol type of the IS adjacency: OSI or DNA IV.

11.9 OSI Configuration and Console Commands

IS-IS-Stats M

Displays information associated with the IS-IS protocol.

Syntax: is-is-stats

Example:

```
is-is-stats
Link State Database Information

no. of level 1 LSPs      1       no. of level 2 LSPs        0
no. of L1 Dijkstra runs  21      no. of L2 Dijkstra runs    0
no. of L1 LSPs deleted   0       no. of L2 LSPs deleted     0
no. of routing table entries allocated                      6

Packet Information

level 1 lan hellos rcvd  0     level 1 lan hellos sent    10967
level 2 lan hellos rcvd  0     level 2 lan hellos sent    10967
pnt to pnt  hellos rcvd  0     pnt to pnt  hellos sent    0
level 1 LSPs rcvd        0     level 1 LSPs sent          40
level 2 LSPs rcvd        0     level 2 LSPs sent          0
level 1 CSNPs rcvd       0     level 1 CSNPs sent         0
level 2 CSNPs rcvd       0     level 2 CSNPs sent         0
level 1 PSNPs rcvd       0     level 1 PSNPs sent         0
level 2 PSNPs rcvd       0     level 2 PSNPs sent         0
```

no. of level 1/level 2 LSPs Indicates the number of L1 and L2 link state packets that are in the database.

no. of L1/L2 Dijkstra runs Indicates the number of times the router computed the L1 and L2 routing tables.

no. of L1/L2 LSPs deleted Indicates the number of L1 and L2 link state packets that were deleted from the database.

no. of routing table entries allocated Indicates the number of entries the routing able currently holds.

level 1/level 2 lan hellos rcvd Indicates the number of LAN hellos the router received.

level 1/level 2 hellos sent Indicates the number of LAN hellos the router sent.

pnt to pnt hellos rcvd Indicates the number of point-to-point hellos that the router received.

pnt to pnt hellos sent Indicates the number of point-to-point hellos that the router sent.

level 1/level 2 LSPs rcvd Indicates the number of L1 and L2 link state packets (LSPs) that the router received.

level 1/level 2 LSPs sent Indicates the number of L1 and L2 LSPs that the router sent.

level 1/level 2 CSNPs rcvd Indicates the number of L1 and L2 complete sequence number PDUs (CSNPs) that the router received.

level 1/level 2 CSNPs sent Indicates the number of L1 and L2 CSNPs that the router sent.

level 1/level 2 PSNPs rcvd Indicates the number of L1 and L2 partial sequence number PDUs (PSNPs) that the router received.

level 1/level 2 PSNPs sent Indicates the number of L1 and L2 PSNPs that the router sent.

L1-Routes M

Displays all the level 1 routes that are in the L1 routing database.

Syntax: l1-routes

Example:

```
l1-routes
Level 1 Routes
Destination System ID  Cost   Source     Next Hop
0000-9300-0047          0     LOC-Area      *
AA00-0400-080C         10     ES-IS      AA00-0400-0C04, Ifc 7
0800-2B92-6BD1         20     IS-IS      AA00-0400-B814, Ifc 0
                                         0800-2B92-6B98, Ifc 0
0000-9378-288D         10     IS-IS      0800-2B26-31FB, Ifc 1
                                         0800-2B26-31FB, Ifc 1
7777-7777-7777         40     IS-IS      3455-6537-2215
```

Destination System ID Indicates the system ID of the destination host.

Cost Indicates the cost of this route.

Source Indicates the one of three sources where the router learned of the route: LOC-AREA, ES-IS, or IS-IS.

Next Hop Indicates the next hop a packet takes on its route. An asterisk (*) designation refers to the router itself as the packet’s destination. An address with an interface number is either the MAC address of a directly connected ES or the DTE address if the next hop is an X.25 switch, or a DLCI if the next hop is a Frame Relay switch. A system ID (*345565372215)* refers to the next hop to destination.

Note: Multiple Next Hop entries against one destination system indicate that there are multiple equal-cost routes to that system. If the addresses are identical, the duplicate routes occur beyond the next hop system.

L2-Routes M

Displays all the level 2 routes in the L2 database.

Syntax: l2-routes

Example:

```
l2-routes
Level 2 Routes
Destination    Cost     Type        Next Hop
4700-0500-01     0      LOC-AREA        *
4900-02          20     AREA        0000-9310-04C9
```

Destination Indicates the system ID of the destination area or reachable address.

Cost Indicates the cost of this route.

Type Indicates the four types of routes: LOC-area (local), LOC-prefix, area, prefix/I, and prefix/E. LOC-area is a directly connected area; LOC-prefix is a prefix that this router advertises; and prefix/I and prefix/E are routes that require another hop to reach their destination.

Next Hop

Indicates the next hop a packet would take on its route. An ***** designation or a "*direct*" designation refers to a directly connected host off the router. A system ID refers to the next router the packet must pass through to reach its destination.

Note: Multiple Next Hop entries against one destination system indicate that there are multiple equal-cost routes to that system. If the addresses are identical, the duplicate routes occur beyond the next hop system.

L1-Summary M

Displays a summary of the level 1 link state database.

Syntax: l1-summary

Example:

```
l1-summary
Link State Database Summary - Level One

LSP ID              Lifetime   Sequence #   Checksum   Flags   Cost

0000-9300-40B0-0000   0           0            0        0     1024
0000-93E0-107A-0000   384         CE          3CC9                0
AA00-0400-0504-0000   298         8E          40F1      B        20
AA00-0400-0504-0100   4           B8          A812      3        20

Total Checksum 25CC
```

LSP ID

This represents the system ID of the source of the link state PDU, plus two additional bytes. The first additional byte designates the type of update. 0 represents a nonpseudonode update. 1-*FF* represents a pseudonode update for that circuit number. The second byte represents the LSP number. This number is attached to the packet when the data is contained in more than one packet.

Lifetime

Indicates the amount of time (in seconds), that the router maintains the LSP.

Sequence #

Indicates the sequence number of the LSP.

Checksum Indicates the checksum value of the LSP.

Flags Indicates a one octet value that reflects the flag field of the LSP. The eight bits are broken down as follows:

- **Bit 8** – Indicates the *P* flag. When set (1), the issuing IS supports the optional Partition Repair function.
- **Bits 7-4** – Indicate the *ATT* flag. When set (1), the issuing IS is attached to other areas using one of the following: the Default Metric (bit 4), the Delay Metric (bit 5), the Expense Metric (bit 6), or the Error Metric (bit 7).
- **Bit 3 –** Indicates the *LSPDBOL* flag. When set (1), an LSP database overload has occurred. An LSP with this bit set is not used by the decision process to calculate routes to another I through the originating system.
- **Bits 2-1** – Indicates the *IS Type* flag. When set to the following values, designates the type of IS router, level 1 or level 2.

Value	**Description**
0	Unused.
1	Bit 1 set. Level 1 IS.
2	Unused.
3	Bits 1 and 2 set. Level 2 IS.

Cost Indicates the cost of routing to that neighbor, in the range 0 to 1024:

0 – Local system (zero cost link).

1 to 1023 – Cost of routing to the neighbor.

1024 – Link not available.

L2-Summary M

Displays a summary of the level 2 link state database.

Syntax: l2-summary

Example:

```
l2-summary
Link State Database Summary - Level Two

LSP ID              Lifetime   Sequence #   Checksum   Flags   Cost
0000-9310-04F0-0000    33E     12             EF19       3       0
0000-5000-FB06-0000    455      4             2BB1       3      20
0000-5000-FB06-0100    469     12             DE32       3      20

Total Checksum 0
```

Description of the **l2-summary** output is the same as the **l1-summary** command listed on the previous page.

L1-Update M

Displays a link state update for the specified level 1 IS.

Syntax: l1-update

Example:

```
l1-update
LSP ID []? 000931004F0000

Link State Update For ID 0000931004F00000

Area Addresses

470005001

Intermediate System Neighbors Metric     Two Way

0000931004F002                  20          N
0000931004F001                  20          Y

End System Neighbors          Metric

00009310004F0                  *
```

LSP ID Indicates the system ID of the source of the link state PDU, plus two additional bytes. The first byte designates the type of update. A 0 represents a nonpseudonode update. A 1-*FF* represents a pseudonode update. The second byte represents the LSP number. This number is attached to the packet when the data is contained in more than one packet.

Area Addresses Indicates the area addresses in which this router is configured to route packets.

Intermediate System Neighbors Indicates adjacent neighbor ISs.

Metric Indicates the cost to the neighbor IS.

Two Way Indicates whether the router is receiving updates from its neighbor.

End System Neighbors Indicates any directly connected ESs.

L2-Update M

Displays the link state update for the specified level 2 IS.

Syntax: l2-update

Example:

```
l2-update
LSP ID []? 000931004F0000

Link State Update For ID 0000931004F00000

INTERMEDIATE SYSTEM NEIGHBORS           METRIC   TWO WAY
   0000931004F002                          20        N
   0000931004F001                          20        N
   55002000182000                          20        N
```

Intermediate System Neighbors Indicates other directly connected ISs.

Metric Indicates the cost to the IS.

Two Way Indicates whether the router is receiving updates from its neighbor.

11.9 OSI Configuration and Console Commands

List C

Displays the current configuration of the OSI protocol.

Syntax: list

adjacency
algorithm
alias
filter
globals
integrated-isis
permitted-neighbors
phaseivpfx
prefix-addresses
routing-circuits
subnet
summary-address
templates
timers
virtual-circuit

adjacency

Displays all statically configured ES adjacencies.

Example:

```
list adjacency
Ifc     Area Address     System ID        MAC Address
 0                       0001-0203-0405   0001-0203-0405
 1                       0002-4000-0000   0000-0019-3004
```

Ifc Indicates the interface number that connects to the adjacency.

Area Address Indicates the area address of this ES adjacency.

System ID Indicates the portion of the NET that identifies the adjacency.

MAC Address Indicates the MAC address (SNPA) of the adjacency.

11.9 OSI Configuration and Console Commands

algorithm

Displays the routing algorithm that is configured in SRAM for the DNA V protocol. If you are running the OSI protocol only, this parameter is unsupported.

Example:

```
list algorithm
Level 1 algorithm - LINK_STATE
Level 2 algorithm - DISTANCE_VECTOR
```

Level 1 Algorithm Indicates the current configuration of the routing algorithm for level 1, Link State (default) or Distance Vector.

Level 2 Algorithm Indicates the current configuration of the routing algorithm for level 2, Link State or Distance Vector (default).

Note: Depending on whether or not DNA IV is enabled or disabled, the routing algorithm displayed here may be different from what is actually running on the router.

alias

Displays the configured aliases and their corresponding address segments.

Example:

```
list aliases
Alias       Segment                       Offset

joplin      AA0004000104                     1
moon        0000931004F0                     1
trane       000093E0107A                     1
```

filter

Displays the defined filters for X.25 routing circuits.

Example:

```
list filters
Rout Cir Name  Filter Name   DTE Addr     Pri  Call Data
routeCir2      filter1       25           5    81
```

11.9 OSI Configuration and Console Commands

globals

Displays the router’s current NET, area addresses, switch settings, global parameters, and timer configuration.

Example:

```
list globals
OSI State: Enabled*            Network Entity Title:
4700050001:0000931004F0
DNAV State: Enabled*

Area Addresses:
1. 4700050001   2. 7700050011


Switches:
ESIS Checksum = On             ES-IS Init Option = Off
Authentication = Off


Globals:
IS Type = L2                   System ID Length = 6
L1 LSP Size = 1492 bytes       L2 LSP Size = 1492 bytes
Max IS Adjs = 50               Max ES Adjs = 200
Max Areas = 50                 Max ESs per Area = 50
Max Int Prefix Adds = 100      Max Ext Prefix Adds = 100
Max Synonymous Areas = 3
```

OSI State or *DNAV State* Indicates if the OSI or DNA V protocol is running on the router.

Network Entity Title: Indicates the area address and system ID that make up the router’s NET.

Area Addresses: Indicates the areas that the router operates within. The first area address reflects the router’s configured NET area address. Additional area addresses were added with the **add area** command.

Globals: Indicates the currently configured global parameters:

IS Type The router’s designation in the OSI environment L1 or L2.

System ID Length The size (in bytes) of the system ID portion of the NET.

Note: All routers throughout the domain must agree on the length of the system ID.

L1 LSP Size / L2 LSP Size — Displays the L1 and L2 maximum LSP buffer size.

Max IS Adjacencies/ Max ES Adjacencies — Displays the maximum number of ES and IS adjacencies this allowed for all circuits.

Max Areas — Displays the maximum number of areas in the routing domain.

Max ESs per Area — Displays the maximum number ESs allowed in one area.

Max Int Prefix Adds — Displays the maximum number of internal prefix addresses.

Max Ext Prefix Adds — Displays the maximum number of external prefix addresses.

Max Synonymous Areas — Displays the maximum number of level 1 areas serviced by this router.

integrated-isis

Displays the configured settings and status for integrated IS-IS.

Example:

```
list integrated-isis
Integrated ISIS:
State = DISABLED
Import RIP Routes = OFF
Import IGP Routes = OFF
Import BGP Routes = OFF
Compare Other IP Routes to Metric Type = EXTERNAL
Originate Default Route = OFF
Originated Default Route Metric Type = EXTERNAL
Originated Default Route Metric = 20
```

State — Indicates whether Integrated IS-IS protocol is running.

Import RIP Routes — Indicates whether routes learned via RIP are imported.

Import IGP Routes — Indicates whether routes learned via IGP are imported.

Import EGP Routes — Indicates whether routes learned via EGP are imported.

Import BGP Routes — Indicates whether routes learned via BGP are imported.

Compare Other IP Routes To Metric Type — Indicates whether imported routes are treated as equivalent to IS-IS internal or external routes.

Originate Default Route Indicates whether default IS-IS routes are announced or not.

Originated Default Route Metric Type If the Originate Default Route is set ON, then this indicates whether the default route is announced as internal or external.

Originated Default Route Metric If the Originate Default Route is set ON, then this value is used as the metric for the default route.

permitted-neighbors

Displays the system-level security verifiers that have been configured.

Example:

```
list permitted-neighbors
Name         Verifier                      Node ID
router1      534543524554                  AA 0 4 0 734
```

Name Indicates the name of the permitted router entry.

Verifier Indicates the value of the verifier in hexadecimal.

Node ID Displays the system ID of the system that has to provide this verifier.

phaseivpfx

Displays the configured DNA phase IV address-prefix that the OSI protocol is using to route packets to a connected DNA IV network.

Example:

```
list phaseivpfx
Local Phase IV Prefix: 49
```

prefix-addresses

Displays all the SNPAs for statically configured routes. Short format prefix addresses are displayed as a table:

Example:

```
list prefix-addresses
Ifc  Type  Metric  State  Address Prefix  Dest Phys Address
 0    INT    20     On        470006        302198112233
 1    EXT    50     OFF       470006        302198223344
```

The fields displayed for extended format prefix addresses depend on the subnet and configuration:

Example:

```
list prefix-addresses
Interface          = 0
Address Prefix     = 490038
Type               = OUTBOUND
Metric Type        = INT
Metric             = 20
Mapping            = MANUAL
MAC Address        = 112233445566
Data Foramt        = 5(V)
State              = ON
```

Interface (Ifc) — Indicates the interface number where the address can be reached.

Address prefix — Indicates the destination NSAP prefix. This prefix may be 20 bytes long.

Type — Indicates whether the address prefix is Inbound or Outbound.

Metric Type — Indicates the type of metric, internal (INT) or external (EXT).

Note: In the old-style display the column heading for this field is *Type*.

Metric — Indicates the cost of the reachable address.

Mapping — Indicates how the destination physical address is determined. It is either MANUAL or X.121.

MAC Address — Indicates the destination MAC address if the interface corresponds to a LAN subnet.

Dest Phys Address — Indicates the physical address of the destination in the form appropriate for the interface type.

For example, it will be the MAC address on a LAN interface or the destination DTE address if this interface is X.25 and the configured mapping is manual.

Data Format — Specifies whether the data format for the circuit is DECnet Phase IV or DECnet Phase V.

State — Indicates whether the prefix address is advertised to other L2 routers (state ON) or not (OFF).

11.9 OSI Configuration and Console Commands

routing-circuits

Displays a summary of all routing circuits or details of each routing circuit.

Example:

```
list routing circuits
Summary or Detailed [Summary]? Summary
 Ifc   Name        Type          Enabled
   0   routecir1   STATIC-OUT    YES
   0   routecir2   STATIC-IN     YES
   0   routecir3   DA            YES
```

Example:

```
list routing circuits
Summary or Detailed [Summary]? Detailed
Routing Circuit Name []?  routecir2

  Interface #:                     0
   Enabled:                        YES
   Type:                           STATIC
   Direction:                      Incoming
   Initial Minimum Timer:          55
   Enable IS-IS:                   YES
   L2 Only:                        NO
   External Domain:                NO
   Metric:                         20
   IS-IS Hello Timer:              3
   DECnetV Link Initialization:    YES
   Receive Verifier:
   Transmit Verifier:
   Explicit Receive Verification: TRUE
```

Interface # / Ifc Logical X.25 interface for this routing-circuit.

Name Alphanumeric name of this routing-circuit record.

Type STATIC-OUT, STATIC-IN, or DA (dynamically assigned).

Enabled State of the routing circuit: YES for enabled, NO for disabled.

Direction How the router establishes a static routing circuit: by an incoming call request (IN) or by an outgoing call request (OUT).

In either case, the SVC is initially established upon operator action, but the circuit is not fully enabled until both ends of the circuit have initialized successfully.

Initial Minimum Timer — Amount of time (in seconds) that a static-out circuit waits for a link to be initialized (reception of either an ESH or an ISH) after the call request is accepted. If the initial minimum timer expires before the link is fully initialized, the SVC is cleared and an event is generated that indicates initialization failure.

Enable IS-IS — Whether the IS-IS protocol is enabled on this routing circuit.

L2 only — Whether this routing circuit is used for level 2 routing only.

External Domain — Whether the router transmits and receives messages to and from a domain outside its IS-IS routing domain.

Metric — Cost of this address.

ISIS Hello Timer — Time interval between transmission of IS-IS hellos.

DECnetV Link Initialization — Whether DEC-style link initialization for this circuit is enabled, yes or no.

Receive Verifier — Verification data to be checked against on receiving an XID when verifying by circuit.

Transmit Verifier — Verification data to be included in the XID.

Explicit Receive Verification — Whether verification is by circuit or by system. TRUE specifies verification by circuit, and FALSE specifies by system.

11.9 OSI Configuration and Console Commands

subnet ***subnet.reprt intfc#***

Displays subnet information.

- *Subnet.reprt* has two options, Summary and Detailed.
  - The summary option displays information for all configured subnets.
  - The detailed option displays information for LAN subnets only.
- *Intfc#* is the interface that connects to the subnet.

Example:

```
list subnet summary
Ifc  State  Type  ESIS  ISIS  L2 Only  Ext Dom  Metric  EIH (sec) IIH(sec)
 0   ON     LAN   ON    ON    NO       NO       20      10        3
 1   OFF    SL    OFF   OFF   YES      YES      50      10        3
 2   ON     X25
 3   ON     FRL
```

Ifc Indicates the interface number of the subnet.

State Indicates the state of the interface, ON or OFF.

Type Indicates the type of subnet: LAN, X25, and Serial Line (SL).

ESIS Indicates the state of the ES-IS protocol, ON or OFF.

ISIS Indicates the state of the IS-IS protocol, ON or OFF.

L2 Only Indicates if the router is operating at level 2 only: yes (true) or no (false).

Ext Dom Indicates if the router is operating outside the IS-IS routing domain (external domain), yes (true) or no (false).

Metric Indicates the cost of using this subnet.

EIH Indicates the interval that ES hello messages are sent out over the subnet.

IIH Indicates the interval that IS hello message are sent out over the subnet.

Example:

```
list subnet detailed
Interface Number [0]?

Detailed information for subnet 0:
   ISIS Level 1 Multicast: 018002B000014
   ISIS Level 2 Multicast: 018002B000015
   All ISs Multicast:      009002B000005
   All ESs Multicast:      009002B000004
   Level 1 Priority: 64
   Level 2 Priority: 64
```

ISIS Level 1 Multicast Indicates the multicast address to use when transmitting and receiving L1 IS-IS PDUs.

ISIS Level 2 Multicast Indicates the multicast address to use when transmitting and receiving L2 IS-IS PDUs.

All ISs Multicast Indicates the multicast address to use when receiving ES hellos.

All ESs Multicast Indicates the multicast address to use when transmitting IS hellos.

Level 1 Priority/Level 2 Priority Indicates the router's priority for becoming the designated router on the LAN.

summary-addresses

Displays summary address information. In this example, all subnet addresses in the range 17.133.0.1 to 17.133.255.254 are abbreviated to 17.133.0.0 with a metric of 1.

Example:

```
list summary-addresses
Summary Address             Mask          Metric
17.133.0.0                  255.255.0.0   1
```

Summary Address Specifies the IP address that will be used to summarize a range of subnet addresses for the local DECnet/OSI area.

Mask Indicates the mask used to match subnets to the summary address.

Metric — Indicates the default metric that will be used when abbreviating the route at level 2.

template

Displays a list of templates defined on this router.

Example:

`list template`

```
Rout Cir Name  Template Name  DTE Addr  Compression  Call Data
routetest2     temptest2      25        ENABLED      81
```

timers

Displays the OSI/DNA V timer configuration.

```
Timers:*
Complete SNP (sec) = 10     Partial SNP (sec) = 2
Min LSP Gen (sec) = 30      Max LSP Gen (sec) = 900
Min LSP Xmt (sec) = 30      Min Br LSP Xmt (msec) = 33
Waiting Time (sec) = 60     DR ISIS Hello (sec) = 1
ES Config Timer (sec) = 10
```

* This output reflects what is actually running on the router: OSI or DNA V.

Timers: — Indicates the configuration of the OSI timers excluding any per-circuit timers.

Complete SNP — The interval between generation of complete SNPs.

Partial SNP — The minimum interval between sending partial SNPs.

Min LSP Generation/Max LSP Generation — The minimum and maximum intervals between generations of LSPs.

Min LSP Transmission — The minimum interval between LSP retransmissions.

Min Broadcast LSP Transmission — The minimum interval between LSP retransmissions on a broadcast circuit.

Waiting Time — The time the update process must delay before entering the ON state.

DR ISIS Hello The interval between generations of IS-IS hello PDUs if this router is a designated router.

ES Config Timer The minimum interval between that an ES must send a hello packet each time an interface comes up.

virtual-circuit

Displays all the configured X.25 SVCs or all the Frame Relay configured virtual circuits.

Example:

```
list virtual-circuit
Ifc State ISIS  L2 Only  Ext Dom  Metric IIH   Dest DTE
 0   On    Ena   False    False     20    3   1238765742
```

Ifc Indicates the interface number over which the configured virtual circuit runs.

State When set to ON, OSI can operate over this circuit.

ISIS Indicates if the IS-IS protocol is running (enabled) over the interface.

L2 Only Indicates if the circuit is operating at level 2 only: yes (true) or no (false).

Ext Dom Indicates if the circuit is operating outside the IS-IS routing domain (external domain).

Metric Indicates the cost of the virtual circuit.

IIH Indicates the interval at which IS-IS hellos are sent out.

Dest DTE Indicates the DTE address of the X.25 network.

11.9 OSI Configuration and Console Commands

Route M

Displays the next hop a packet takes to a specified destination (*dest-nsap*).

Syntax: route *dest-nsap*

Example:

```
route 490002aa0004000e08
Destination System: 0000-9310-04C9
Destination MAC Address: AA00-0400-1408
Interface: 0
```

Destination System	Indicates the system ID of the next hop IS. For a directly connected ES, this is blank.
Destination MAC Address	Indicates the MAC address of the next hop IS or the directly connected ES.
Interface	Indicates the interface that a packet goes out over to reach the the next hop IS or the directly connected ES.

Send (Echo Packet) M

Encode an echo request message in the CLNP packet to the specified destination NSAP. During this command, the system does not interact with the OSI console. To verify that the echo request was sent and that an echo reply was received, check the ELS (Event Logging System).

Note: You cannot send an echo packet to yourself. If you try, you receive a CLNP.004 ELS message.

Syntax: send

Example:

```
send
Destination NSAP: []?
```

11.9 OSI Configuration and Console Commands

Set C

Configures the router to run the OSI protocol.

Syntax: set

adjacency
algorithm
globals
network-entity-title
phaseivpfx
subnet
switches
timers
transmit-password
virtual-circuit

adjacency

Adds or changes an ES adjacency. Add an ES adjacency for all LAN ESs that do not run the ES-IS protocol.

Example:

```
set adjacency
Interface Number [0]?
Area Address []?
System ID []?
MAC Address []?
```

Interface Number Indicates the interface number that connects to the adjacency.

Area Address Indicates the area where the adjacency is located.

System ID Indicates the system ID portion of the NET that is used to identify the adjacency.

MAC Address Indicates the MAC address (SNPA) of the adjacency.

11.9 OSI Configuration and Console Commands

algorithm

Note: This is a DNA phase V command. This command works only if the DNA phase V protocol is included in the software load.

This command allows you to select the type of routing algorithm that you are using for the DNA routing protocol, link state (DNA V), or distance vector (DNA IV).

Example:

```
set algorithm
Level 1 Algorithm [link_state]?
Level 2 Algorithm [distance_vector]?
```

Level 1 Algorithm Selects the type of routing algorithm, link_state (for DNA V networks), or distance_vector (for DNA IV networks).

Level 2 Algorithm Selects the type of routing algorithm, link_state (for DNA V networks), or distance_vector (for DNA IV networks).

globals

Configures the global parameters required by the OSI protocol.

Example:

```
set globals
IS Type [L2]?
Domain ID Length [6 bytes]?
Max Synonymous Areas [3]?
L1 LSP Buffer Size [1492 bytes]?
L2 LSP Buffer Size [1492 bytes]?
Max IS Adjacencies [50]?
Max ES Adjacencies [200]?
Max Areas [50]?
Max ESs per Area [500]?
Max Internal Prefix Addresses [100]?
Max External Prefix Addresses [100]?
Max Link State Updates [100]?
```

IS Type (L1 or L2) Selects the level of the router, level 1 or level 2.

Domain ID Length Selects the length of the domain ID portion of the NET. This length must be the same for all routers in the same domain.

Max Synonymous Areas Selects the maximum number of level 1 areas that are serviced by this router.

L1 LSP Buffer Size Selects the buffer size of the level 1 LSPs and SNPs originated by the router. Range is 512 to 1492. If the interface packet size is less than what you configured here, OSI does not run and the router generates the ELS message ISIS.053.

L2 LSP Buffer Size Selects the buffer size of the level 2 LSPs and SNPs originated by the router. Range is 512 to 1492. If the interface packet size is less than what you configured here, OSI does not run and the router generates the ELS message ISIS.053.

Max IS Adjacencies Selects the total number of IS adjacencies allowed for all circuits. This number is used to size the IS adjacency free pool.

Max ES Adjacencies Selects the total number of ES adjacencies allowed for all circuits. This number is used to size the ES adjacency free pool.

Max Areas Selects the total number of areas in the routing domain. This number is used to size the L2 routing table.

Max ESs per Area Selects the total number ESs in any one area. This number is used to size the L1 routing table.

Max Internal Prefix Addresses Selects the number you are using to size the internal metric routing table.

Max External Prefix Addresses Selects the number you are using to size the external metric routing table.

Max Link State Updates Selects the number you are using to size the link state database.

11.9 OSI Configuration and Console Commands

network-entity-title

Configures the router’s NET. The NET consists of the router’s system ID and area address.

Note: You cannot use this command if DNA IV routing is enabled. The DNA IV compatible NET is set automatically for you when OSI is enabled.

Example:

```
set network-entity-title
Area Address []?
System-ID []?
```

Area Address Indicates one of area address portions of the router’s NET. It is included as the first address in the router’s set of manual area addresses. Each area address may be a maximum of 19 bytes.

System-ID Defines the portion of the NSAP that identifies this specific router. The system ID can be a maximum of 19 bytes, but the length must agree with the domain ID length that you configured with the **set globals** command.

phaseivpfx

Configures the prefix address to allow the OSI protocol to route packets to the attached DNA IV network. The default is 49 (hexadecimal).

Example:

```
set phaseivpfx
Local Phase IV prefix [49]?
```

If DNA IV routing is enabled when you use this command, the NET is set to the default Phase IV compatible value using the Phase IV prefix and the DNA IV address. In the example below, DNA IV is enabled, with address 13.1.

Example:

```
set phaseivpfx
Local Phase IV prefix [49]? 471212
Phase IV routing is enabled
Defaulting NET to Phase IV compatible [471212000D:AA0004000134]
```

subnet

Adds or changes a subnet. This parameter prompts you for different information depending on the type of subnet that you are configuring: X.25, serial line (SL), or LAN.

Example:

`set subnet`

X.25 subnet:

```
Interface number [0]?1
Subnet Type:  X25
```

Serial Line subnet:

```
Interface number [0]?1
Subnet Type:  SL
Enable IS-IS [YES]?
Level 2 Only [NO]?
External Domain [NO]?
Default Metric [20]?
ISIS Hello Timer [3]?
Modify Transmit password [NO]?
Modify the set of receive passwords [NO]?
```

LAN subnet:

```
Interface number [0]?
Subnet Type: LAN
Enable ES-IS [YES}?
Enable IS-IS [YES]?
Level 2 Only [NO]?
External Domain [NO]?
Default Metric [20]?
ESIS IS Hello Timer [10]?
ISIS Hello Timer [3]?
Modify Transmit password [NO]?
Modify the set of receive passwords [NO]?
L1 Priority [64]?
L2 Priority [64]?
All ESs [09002B000004]?
All ISs [09002B000005]?
All L1 ISs [0180C2000014]?
All L2 ISs [0180C2000015]?
```

Frame Relay subnet:

```
Interface number [0]?1
Subnet Type:  FRL
```

11.9 OSI Configuration and Console Commands

Interface number Binds the subnet to the specified interface.

Subnet Type Indicates the type of subnet: LAN, Serial Line (SL), X.25, and Frame Relay (FRL). LAN includes Ethernet, Token Ring, and FDDI.

Use SL (serial line) for fixed serial lines such as PPP leased lines or PPP-FR circuits (PVCs and SVCs) and for dial circuits (V.25 *bis* or ISDN).

Enable ES-IS Indicates whether the ES-IS protocol is going to run over the interface: yes (Y) or no (N).

Enable IS-IS Indicates whether the IS-IS protocol is going to run over the interface: yes (Y) or no (N).

Level 2 Only Indicates whether the subnet runs at level 2 only: yes (Y) or no (N). A no designation allows the router to route over that subnet at both level 1 and level 2.

External Domain Indicates whether the circuit is operating outside the IS-IS routing domain.

Default Metric Indicates the cost of the subnet. Cost range 1-63.

IS Hello Timer Indicates the period between transmissions of IS hello PDUs.

ISIS Hello Timer Indicates the period between transmissions of L1 and L2 IS-IS hello PDUs.

Modify Transmit password Removes or changes a circuit transmit password. When you select yes, this option prompts you with the following message:

```
Delete or change the transmit password  [change]?
```

Modify the set of receive passwords Removes all or adds one circuit receive password. When you select yes, this option prompts you with the following message:

```
Delete all or add 1 receive password [add]?
```

L1 Priority/L2 Priority Indicates the router priority for becoming the designated router on the LAN.

All ESs — Indicates the multicast address to use when transmitting IS hellos. The default address reflects the Ethernet/802.3 multicast address. If you are connecting to an 802.5 LAN, use C00000004000.

All ISs — Indicates the multicast address to use when receiving ES hellos. The default address reflects the Ethernet/802.3 multicast address. If you are connecting to an 802.5 LAN, use C00000008000.

All L1 ISs — Indicates the multicast address to use when transmitting and receiving L1 IS-IS PDUs. The default address reflects the Ethernet/802.3 multicast address. If you are connecting to an 802.5 LAN, use C00000008000.

All L2 ISs — Indicates the multicast address to use when transmitting and receiving L2 IS-IS PDUs. The default address reflects the Ethernet/802.3 multicast address. If you are connecting to an 802.5 LAN, use C00000008000.

switches

Turns the OSI options on or off.

Example:

```
set switches
ES-IS Checksum Option [OFF]?
ES-IS Init Option [OFF]?
Authentication [OFF]?
```

ES-IS Checksum Option — When switched on, the router generates checksums for all sourced ES-IS packets.

ES-IS Init Option — When switched on, the router sends a directed IS Hello to a new ES neighbor.

Authentication — If switched on, each IS-IS packet includes the transmit password configured for the domain, area, and circuits. No checking of receive passwords is done.

11.9 OSI Configuration and Console Commands

timers

Configures the OSI timers, excluding any circuit timers.

Example:

```
set timers
Complete SNP [10 sec]?
Partial SNP [2 sec]?
Min LSP Generation [30 sec]?
Max LSP Generation [900 sec]?
Min LSP Transmission [5 sec]?
Min Broadcast LSP Transmission [33 msec]?
Waiting Time [60 sec]?
Designated Router ISIS Hello [1 sec]?
Suggested ES Configuration Timer (sec) [10]?
```

Complete SNP Selects the interval between the generation of complete sequence number PDUs (SNP) by the designated router on a broadcast circuit.

Partial SNP Selects the minimum interval between sending partial sequence number PDUs (SNP).

Min LSP Generation Selects the minimum interval between successive generations of Link State Packets (LSPs) with the same LSP ID generated by the router.

Max LSP Generation Selects the maximum interval between LSPs generated by the router.

Min LSP Transmission Selects the minimum interval between retransmissions of a LSP.

Min Broadcast LSP Transmission Selects the minimum transmission, in milliseconds, between transmission of LSPs on a broadcast circuit.

Waiting Time Selects the number of seconds the update process delays in the waiting state before entering the ON state.

Designated Router ISIS Hello Selects the interval between the generation of IS-IS hello PDUs by the router if the router is the designated router on a LAN.

Suggested ES Configuration Timer Sets the option field of the IS hello message that instructs the ES to change the rate at which it sends ES hellos.

transmit-password

Sets or changes a transmit password.

Example:

```
set transmit-password
Password type [Domain]?
Password []?
Renter password:
```

Password type

Selects the type of password: domain or area.

- Domain passwords are used with L2 LSPs and SNPs.
- Area passwords are used with L1 LSPs and SNPs.

Password

Indicates the character string that you are using for authentication. Maximum allowable string can be 16 characters.

virtual-circuit

Configures a X.25 SVC or a Frame Relay virtual circuit.

Example:

```
set virtual-circuit
Interface Number [0]?
DTE Address []?
Enable IS-IS (Y or N) [Y]?
L2 only (Y or N) [N]?
External Domain (Y or N) [N]?
Default Metric [20]?
ISIS Hello Timer [3 sec]?
Modify transmit password (y or n) [N]?
Modify the set of receive passwords [No]?
```

Interface Number

Indicates the X.25 or Frame Relay interface over which the virtual circuit is configured.

DTE Address

Indicates the destination DTE address for X.25 or the DLCI (Data Link Control Identifier) for Frame Relay. This address must be the same as the one defined for the virtual circuit in the X.25 configuration or the Frame Relay configuration.

Default Metric Indicates the cost of the circuit.

Enable IS-IS Indicates whether the IS-IS protocol is going to run over the interface: yes (Y) or no (N).

L2 only Indicates whether the circuit runs at level 2 only: yes (Y) or no (N). A no designation allows the router to route at both level 1 and level 2.

External Domain Indicates whether the circuit is operating outside the IS-IS routing domain.

Show routing circuits M

Displays the state of routing circuits for the specified interface. Applies only when the router is configured as a DEC-style router.

Syntax: show routing circuits

Example:

```
show routing circuits
Interface Number [0]?
Name              Type        State
routeCir1         DA          INIT
routeCir2         STATIC-IN   UP
```

Subnets M

Displays information on all operational subnets. Subnets that are down or disabled are not listed.

Syntax: subnets

Example:

```
subnets
                     L2
 Hdw   Int #  Circ   Only  ES-IS IS-IS  L1DR  LPri  L2DR  L2pri  Cost  Ext
 SL /2   2      3    N     N     Y
 Eth /0  0      1    N     Y     Y      Y     64    N     64     20    N
FDDI/1   1      2    N     Y     Y      N     64    N     64     20    N
```

Hdw Indicates the type and instance of the network that connects to the subnet.

Int # Indicates the router’s interface number that connects to the subnet.

Circ Indicates the circuit-assigned ID for the IS-IS protocol.

L2 only Indicates whether this router is a level 2 router only: Y (yes) or N (no).

ES-IS Indicates if ES-IS protocol is enabled on the subnet: Y or N.

IS-IS Indicates if the IS-IS protocol is enabled on the subnet: Y or N.

L1DR Indicates if this router is the level 1 designated router for this subnet: Y or N.

L2Pri Indicates the subnet’s level 1 priority for becoming the designated router.

L2DR Indicates if this router is the level 2 designated router for this subnet: Y or N.

LPri Indicates the LAN subnet’s level 2 priority for becoming the designated router.

Cost Indicates the cost of the circuit.

Ext Indicates whether the subnet is operating outside the IS-IS routing domain (external).

Toggle (Alias /No Alias) M

Enables or disables the NSAP alias display function for the OSI protocol.

Syntax: toggle

Example:

```
toggle
Alias substitution is ON
```

Traceroute M

Tracks the path an OSI packet takes to a destination.

Note: You cannot do a traceroute to yourself or you will receive the following error message:

```
Sorry, can’t traceroute to this router.
```

Syntax: traceroute *address*

Example:

```
traceroute 490002aa0004000e08
Successful trace:

TRACEROUTE 470007: 56 databytes

1          490002aa0004000e08    32ms     5 ms     5ms

Destination unreachable response:

Destination unreachable

No response:

1 * * *
2 * * *
```

TRACEROUTE Displays the destination area address and the size of the packet being sent to that address.

1 The first trace showing the destination’s NSAP and the amount of time it took the packet to arrive at the destination. The packet is traced three times.

Destination unreachable Indicates that no route to the destination is available.

*1 * * **
*2 * * ** Indicates that the router is expecting some form of response from the destination, but the destination is not responding. The router waits 32 hops before timing out. Go to the ELS and turn on OSI CLNP messages to determine why the host is not responding.

Virtual-circuits M

Displays information about all X.25 virtual circuits.

Note: This applies only when the router is configured as a Proteon-style router.

Example:

```
virtual-circuits
                         L2    Ext                DLCI     Dest
Hdw  Ifc State ISIS  Only  Dom  Metric  IIH  Address  Address
X25  0         ON    NO    NO   02      3
```

Exit C M

Returns to the previous prompt level.

Syntax: exit

Example:

```
exit
```

12

Configuring and Monitoring OSPF

This chapter describes how to configure the OSPF protocol using the OSPF configuration commands and how to monitor the OSPF protocol using the OSPF console commands.

For more information about OSPF, refer to the *Routing Protocols Reference Guide*.

12.1 Accessing the OSPF Configuration and Console Environments

For information about accessing the OSPF configuration and console environments, see Chapter 1.

12.2 Basic Configuration Procedures

The following sections present information about how to initially configure the OSPF protocol. This information outlines the tasks required to set up and run the OSPF protocol. Information about how to make further configuration changes is explained in the command sections of this chapter. A complete configuration example appears at the end of this section.

12.2.1 Before You Begin

Before your router can run the OSPF protocol, you must perform the following steps:

1. Enable the OSPF protocol. In doing so, you must estimate the final size of the OSPF routing domain.
2. Define OSPF areas attached to the router. If no OSPF areas are defined, a single backbone area is assumed.
3. Define the router’s OSPF network interfaces. The cost of sending a packet through each interface must be set, along with a collection of the OSPF operating parameters.

4. If you want to forward IP multicasts (IP Class D addresses), enable IP multicast routing capability.
5. If the router interfaces to non-broadcast networks, you must also set the non-broadcast network parameters. This consists of a list of the other OSPF routers that are connected to the non-broadcast network.
6. If you want the router to import routes learned from other routing protocols (EGP, RIP, or statically configured routes), you must enable AS boundary routing. In addition, you must define whether routes are imported as Type 2 or Type 1 externals.
7. If you want to boot a neighboring router through an attached point-to-point interface, the neighbor’s IP address must be configured. This is done by defining non-broadcast parameters for the point-to-point interface.

12.2.2 Enabling the OSPF Protocol

When enabling the OSPF routing protocol, you must supply the following two values to estimate the size of the OSPF link state database. These two values are configured identically in all of your OSPF routers.

- Total number of AS external routes that are imported into the OSPF routing domain. A single destination may lead to multiple external routes when it is imported by separate AS boundary routers. For example, if the OSPF routing domain has two AS boundary routers, both importing routes to the same 100 destinations, the number of AS external routes is set to 200.
- Total number of OSPF routers in the routing domain.

To enable the OSPF routing protocol, use the **enable** command as shown in the following example:

Example:

```
OSPF Config> enable ospf
Estimated # external routes[0]? 200
Estimated # OSPF routers [0]? 60
```

12.2.3 Defining Attached OSPF Areas

The next step in the configuration process is setting the parameters that define the OSPF areas that are directly attached to the router. If no areas are defined, the router software assumes that all the router's directly attached networks belong to the backbone area (area ID 0.0.0.0).

To set the parameters for an OSPF area, use the **set area** command and respond to the following prompts:

Example:

```
OSPF Config> set area
Area number [0.0.0.0]? 0.0.0.1
Authentication type [1]? 1
Is this a stub area? (Yes or No): no
```

- Area number is the OSPF area address. An OSPF area is a contiguous group of networks that is defined by a list of address ranges, each indicated by a combination of the IP address and an address mask. A network belongs to an area if its address is in the list.
- Authentication type (security scheme) to be used in the area. The choices for authentication types are 1, which indicates a simple password; or 0, which indicates that no authentication is necessary for the exchange.
- Stub area designation. If you designate YES:
  - The area does not receive any AS external link advertisements, reducing the size of the area's OSPF database and decreasing memory usage for external routers in the stub area.
  - You cannot configure virtual links through a stub area.
  - You cannot configure a router within the stub area as an AS boundary router.
  - You get two further prompts:

    ```
    Stub default cost [0]?
    Import summaries? [yes]
    ```

  - Answering **yes** to import summaries (inter-area routes) allows summary routes to be flooded into the router.

External Routing in Stub Areas. You cannot configure the backbone as a stub area. External routing in stub areas is based on a default route. Each border area router attaching to a stub area originates a default route for this purpose. The cost of this default route is also configurable in the OSPF **set area** command.

12.2 Basic Configuration Procedures

12.2.4 Setting OSPF Interfaces

To set the OSPF parameters for the router’s network interfaces, use the **set interface** command.

When responding to the prompts, supply the interface’s IP address for each interface in the router and answer the questions that follow. For the parameters listed below, you must enter the same value for all routers attached to a common network segment:

- Hello interval
- Dead router interval
- Authentication key (if an authentication type of 1 (simple password) is used)

The first prompt asks for the OSPF area to which the interface attaches. In the following example, suppose that the interface address mask is 255.255.255.0, indicating that the interface attaches to a subnet (128.185.138.0) of network 128.185.0.0. All other OSPF routers attached to subnet 128.185.138.0 must also have their *hello interval* set to 10, *dead router interval* set to 40, and their interface *authentication key* set to xyz_q.

Example:

```
OSPF Config> set interface
Interface IP address [0.0.0.0]? 16.24.11.251
Attaches to area [0.0.0.0]?
Retransmission Interval (in seconds) [5]?
Transmission Delay (in seconds) [1]?
Router Priority [1]?
Hello Interval (in seconds) [10]?
Dead Router Interval (in seconds) [40]?
Type Of Service 0 cost [1]? 2
Authentication Key []?
Retype Auth. Key []?
```

12.2.5 Setting Non-Broadcast Network Interface Parameters

If the router is connected to a non-broadcast, multi-access network, such as an X.25 PDN, you have to configure the parameters below to help the router discover its OSPF neighbors. This configuration is only necessary if the router is eligible to become the designated router of the non-broadcast network.

First configure the OSPF poll interval with the following command:

Example:

```
OSPF Config> set non-broadcast
Interface IP address [0.0.0.0]? 128.185.138.19
Poll Interval [120]?
```

Then configure the IP addresses of all other OSPF routers that are attached to the non-broadcast network. For each router configured, you must also specify its eligibility to become the designated router.

Example:

```
OSPF Config> add neighbor
Interface IP address [0.0.0.0]? 128.185.138.19
IP Address of Neighbor [0.0.0.0]? 128.185.138.21
Can that router become Designated Router [Yes]?
```

12.2.6 Enabling IP Multicast Routing

To enable the routing of IP multicast (class D) datagrams, you must invoke the **enable multicast** command in the OSPF configuration console. This command is described below. When enabling multicast routing, you are also prompted as to whether you want the router to forward multicasts between OSPF areas as well as whether you want the router to forward multicasts between Autonomous Systems.

Example:

```
OSPF Config>enable multicast
Inter-area multicasting enabled(Yes or No): yes
Inter-AS multicasting enabled(Yes or No): yes
```

When the above command is first invoked to enable multicast forwarding, multicast is enabled on all OSPF interfaces with default parameters. The interface parameters can be later modified using the OSPF **set interface** command.

Unless the **enable multicast** command is invoked, forwarding of IP multicast datagrams is disabled. In other words, by default the router does not forward IP class D datagrams.

12.2.7 Enabling AS Boundary Routing

To import routes learned from other protocols (EGP, RIP, and statically configured information) into the OSPF domain, enable AS boundary routing. You must do this even if the only route you want to import is the default route (destination 0.0.0.0).

When enabling AS boundary routing, you are asked which external routes you want to import. You can choose to import, or not to import, routes belonging to several categories. The categories are as follows:

- RIP routes
- EGP routes
- BGP routes
- Static routes
- Direct routes

For example, you can choose to import EGP and direct routes, but not RIP or static routes. When you choose to import EGP routes, only the routes that appear in the EGP input exchange tables are actually imported. All routes are imported with cost equal to their routing table cost. They are all imported as either type 1 or type 2 external routes, depending on the routing protocol comparison.

Independent of the external categories, you can also configure whether or not to import subnet routes into the OSPF domain. This configuration item defaults to OFF (subnets not imported).

The metric type used in importing routes determines how the imported cost is viewed by the OSPF domain. When comparing two type 2 metrics, only the external cost is considered in picking the best route. When comparing two type 1 metrics, the external and internal costs of the route are combined before making the comparison.

You are asked whether or not you want to originate an OSPF default route. You can answer always, never, or only if you have EGP/BGP routes. If originating a default route when EGP/BGP routes are available, you can also choose to originate the default only if EGP/BGP routes are received from a particular Autonomous System or if a particular route is received through the EGP.

Combinations of these options are possible. For example, you can set the router so that its default is originated only if a route to 10.0.0.0 is received from AS number 12. Setting the AS number to 0 means “from any AS.” Setting the network number to 0.0.0.0 means “any routes received.”

The syntax of the **enable** command is as follows:

Example:

```
OSPF Config>enable as boundary
Import EGP routes? [No]: y
Import BGP routes? [No]:
Import RIP routes? [No]:
Import static routes? [No]:
Import direct routes? [No]: y
Import subnet routes? [No]:
Always originate default route? [No]:
Originate default if EGP/BGP routes available? [No]: y
   From AS number [0]? 12
   To network number [0.0.0.0]? 10.0.0.0
Originate as type 1 or 2 [2]?
Default route cost [1]?
Default forwarding address [0.0.0.0]?
```

12.2.7.1 Forwarding Address for OSPF Default Route

Normally, if you configure a router to originate a default route, then other routers on the network will send it packets that they cannot route themselves.

If you enter a *Default Forwarding Address*, the routers receiving the default route will not send their unroutable packets to this router. Instead they will send their packets to the router identified by the *Default Forwarding Address*.

You would typically do this if the router is:

- Running both OSPF and BGP, and
- Learning all its BGP routes—that is, routes in other autonomous systems—from a BGP-only router

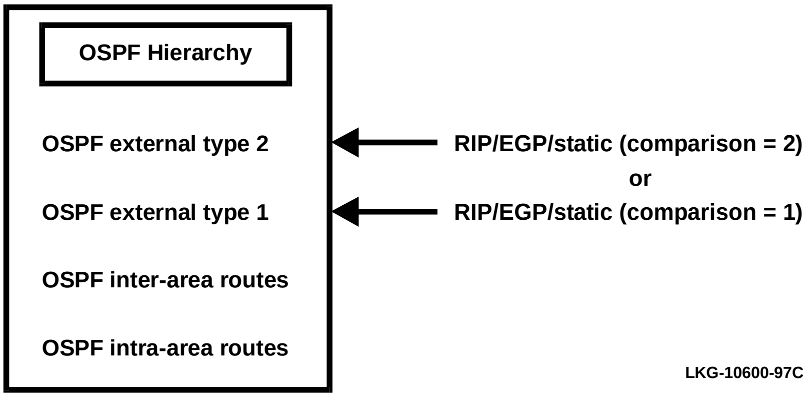
In this case, the router is broadcasting the default route to the other OSPF routers to tell those routers where to send packets whose destination is in another autonomous system. It is therefore advisable to inform OSPF routers to send such packets directly to the BGP router from which you are learning the BGP routes.

12.2.8 Configuring for Routing Protocol Comparisons

If you use a routing protocol in addition to OSPF, or when you change your routing protocol to OSPF, you must set the Routing Protocol Comparison.

OSPF has a 4-level routing hierarchy (see Figure 12–1). The **set comparison** command tells the router where the EGP/RIP/static routes fit in the OSPF hierarchy. The two lower levels consist of the OSPF internal routes. OSPF intra-area and inter-area routes take precedence over information obtained from any other sources, all of which are located on a single level.

Figure 12–1 OSPF Routing Hierarchy



To put the EGP/RIP/static routes on the same level as OSPF external type 1 routes, set the comparison to 1. To put the EGP/RIP/static routes on the same level as OSPF external type 2 routes, set the comparison to 2. The default setting is 2.

For example, suppose the comparison is set to 2. In this case, when RIP routes are imported into the OSPF domain, they are imported as type 2 externals. All OSPF external type 1 routes override received RIP routes, regardless of metric. However, if the RIP routes have a smaller cost, the RIP routes override OSPF external type 2 routes. The comparison values for all of your OSPF routers must match. If the comparison values set for the routers are inconsistent, your routing does not function properly.

The syntax of the **set comparison** command is as follows:

Example:

```
OSPF Config> set comparison
Compare to type 1 or 2 externals [2]?
```

12.2.9 Setting Virtual Links

To maintain backbone connectivity, you must have all of your backbone routers interconnected either by *permanent* or *virtual* links. Virtual links may be configured between any two area border routers that share a common non-backbone and non-stub area. Virtual links are considered to be separate router interfaces connecting to the backbone area. Therefore, you are asked to also specify many of the interface parameters when configuring a virtual link.

Note: If the router is an ABR that does not directly attach to the backbone area (0.0.0.0), you must create a virtual link. Logically, virtual links belong to area 0.0.0.0 and you must configure ABR to set area 0.0.0.0.

The example below illustrates the configuration of a virtual link. Virtual links must be configured in each of the link's two endpoints. Note that OSPF router IDs are entered in the same form as IP addresses.

Example:

```
OSPF Config> set virtual
Virtual endpt.  (Router ID) [0.0.0.0]? 128.185.138.21
Link’s transit area [0.0.0.1]? .0.0.1
Retransmission Interval (in seconds) [10]?
Transmission Delay (in seconds) [5]?
Hello Interval (in seconds) [30]?
Dead Router Interval (in seconds) [180]?
Authentication Key []? 3-14159
```

12.2.10 OSPF Router IDs

Every router in an OSPF routing domain must be assigned a 32-bit router ID. The current OSPF implementation sets the OSPF router ID to be the address of the first OSPF interface appearing in the router's configuration.

The OSPF router ID can also be explicitly set by the IP **set router id** command. In this case, the router ID must still be one of the router's IP interface addresses.

12.2.11 Converting from RIP to OSPF

To convert your Autonomous System from RIP to OSPF, install OSPF one router at a time, leaving RIP running. Gradually, all your internal routes shift from being learned through RIP to being learned by OSPF (OSPF routes have precedence over RIP routes). If you want to have your routes look exactly as they did under RIP (in order to check that the conversion is working properly), use hop count as your OSPF metric. This is done by assigning the cost of each OSPF interface to 1.

12.2 Basic Configuration Procedures

Remember that the size of your OSPF system must be estimated when the protocol is enabled. This size estimate should reflect the final size of the OSPF routing domain.

After installing OSPF on your routers, turn on AS boundary routing in all those routers that still need to learn routes through other protocols (EGP, RIP, and statically configured routes). Keep the number of these AS boundary routers to a minimum.

Finally, you can disable the receiving of RIP information about all those routers that are not AS boundary routers.

12.2.12 Dynamically Changing Interface Costs

The cost of an OSPF interface can be dynamically changed from the router’s console interface. This new cost is flooded quickly throughout the OSPF routing domain, and modifies the routing immediately.

When the router restarts/reloads, the cost of the interface reverts to the value that was configured in SRAM. In future router releases, you are also able to set the cost of an OSPF interface using the SNMP protocol.

12.2.12.1 Sample Configuration Procedure for OSPF

The following procedure demonstrates the configuration of OSPF on a RouteAbout Access EI bridging router.

In this procedure, each parameter that has a default value is given that value. For parameters that have no default (for example, addresses), the procedure uses an arbitrary value.

Note: You should check all values carefully and change them as necessary for your individual implementation.

The following steps summarize the procedure you can use the to configure the OSPF protocol on router interfaces:

1. Talk to the router and access OSPF.
2. Enable the OSPF protocol.
3. Define the attached OSPF areas.
4. Set OSPF interfaces.
5. Set non-broadcast interface parameters.
6. Enable IP multicast routing.
7. Enable AS boundary routing.

8. Configure for routing protocol comparisons.
9. Set up virtual links.
10. Set the OSPF router ID.

To configure OSPF on the router, peform the following steps. You may have to change the values or parameters given here according to your network setup.

Note: You must restart the router to put these values in effect.

1. Talk to the router and access OSPF.

```
* t 6
Config> protocol OSPF

Open SPF-Based Routing Protocol configuration console

OSPF Config>
```

2. Enable the OSPF protocol.

```
OSPF Config> enable ospf
Estimated # external routes [0]? 200
Estimated # OSPF routers [0]? 60
OSPF Config>
```

3. Define the attached OSPF areas.

```
OSPF Config> set area
Area number [0.0.0.0]? 1.1.2.2
Authentication Type [0]? 1
Is this a stub area [No]? no
OSPF Config>
```

4. Set OSPF interfaces.

```
OSPF Config> set interface
Interface IP address [0.0.0.0]? 16.24.11.251
Attaches to area [0.0.0.0]? 0.0.0.0
Retransmission Interval (in seconds) [5]? 5
Transmission Delay (in Seconds) [1]? 1
Router Priority [1]? 1
Hello Interval (in seconds) [10]? 10
Dead Router Interval (in seconds) [40]? 40
Type of Service 0 cost [1]? 1
Authentication key []? auth_key
Retype Auth. Key []? auth_key
OSPF Config>
```

12.2 Basic Configuration Procedures

5. Set non-broadcast interface parameters.

```
OSPF Config> set non-broadcast
Interface IP address [0.0.0.0]? 128.185.138.19
Poll interval [120]? 120
OSPF Config>
```

Add the neighbor(s):

```
OSPF Config> add neighbor
Interface IP Address [0.0.0.0]? 128.185.138.19
IP Address of Neighbor [0.0.0.0]? 128.185.138.21
Can the router become Designated Router on this net [yes]? yes
OSPF Config>
```

6. Enable IP multicast routing.

```
OSPF Config> enable multicast
Inter-area multicasting enabled [No]? yes
OSPF Config>
```

7. Enable AS boundary routing.

```
OSPF Config> enable AS boundary
Import EGP routes? [No]: yes
Import BGP routes? [No]: no
Import RIP routes? [No]: no
Import static routes? [No]: no
Import direct routes? [No]: yes
Import subnet routes? [No]: no
Always originate default route? [No]: no
Originate default in EGP/BGP routes available? [No]: yes
  From AS number [0]? 12
  To network number [0.0.0.0]? 10.0.0.0
Originate as type 1 or 2 [2]? 2
Default route cost [1]? 1
Default forwarding address [0.0.0.0]? 0.0.0.0
OSPF Config>
```

8. Configure for routing protocol comparisons.

```
OSPF Config> set comparison
Compare to type 1 or 2 externals [2]? 2
OSPF Config>
```

9. Set up virtual links.

```
OSPF Config> set virtual
Virtual endpoint (Router ID) [0.0.0.0]? 128.185.138.21
Link’s transit area [0.0.0.1]? 0.0.0.1
Retransmission Interval (in seconds) [10]? 10
Transmission Delay (in seconds) [5]? 5
Hello Interval (in Seconds) [30]? 30
Dead Router Interval (in seconds) [180]? 180
Authentication Key []? 3-14159
OSPF Config>
```

10. Set the OSPF router ID.

```
OSPF Config> exit
Config> protocol IP

Internet protocol user configuration

IP Config> set router ID
Router-ID [0.0.0.0]? 128.185.138.21
IP Config> exit
Config>
```

12.3 OSPF Configuration and Console Commands

The following section summarizes and explains the OSPF configuration and monitoring commands. You enter the configuration commands at the `OSPF config>` prompt. Enter the OSPF monitoring (console) commands at the `OSPF>` prompt.

Table 12–1 lists the OSPF configuraton and console commands.

12.3 OSPF Configuration and Console Commands

Table 12–1 OSPF Configuration and Console Commands Summary

Command	Tasks	Function
? (Help)	Configuring and Monitoring	Lists the OSPF configuration commands or lists the options associated with specific commands.
Add	Configuring	Adds to already existent OSPF information. You can add ranges to areas, and neighbors to non-broadcast networks.
Advertisement	Monitoring	Displays a link state advertisement belonging to the OSPF database.
Area summary	Monitoring	Displays OSPF area statistics and parameters.
AS-external	Monitoring	Lists the AS external advertisements belonging to the OSPF link state database.
Database summary	Monitoring	Displays the advertisements belonging to an OSPF area's link state database.
Delete	Configuring	Deletes OSPF information from SRAM.
Disable	Configuring	Disables the entire OSPF protocol, AS boundary routing capability, or IP multicast routing.
Dump routing tables	Monitoring	Displays the OSPF routes contained in the routing table.
Enable	Configuring	Enables the entire OSPF protocol, AS boundary routing capability, or IP multicast routing.
Interface Summary	Monitoring	Displays OSPF interface statistics and parameters.
Join	Configuring and Monitoring	Configures the router to belong to one or more multicast groups.
Leave	Configuring and Monitoring	Removes the router from membership in multicast groups.
List	Configuring	Displays OSPF configuration.
Mcache	Monitoring	Displays a list of currently active multicast forwarding cache entries.
Mgroups	Monitoring	Displays the group membership of the router's attached interfaces.
Mstats	Monitoring	Displays various multicast routing statistics.

Table 12–1 OSPF Configuration and Console Commands Summary (Continued)

Command	Tasks	Function
Neighbor summary	Monitoring	Displays OSPF neighbor statistics and parameters.
Routers	Monitoring	Displays the reachable OSPF area-border routers and AS-boundary routers.
Set	Configuring	Establishes or changes the configuration information concerning OSPF areas, interfaces, non-broadcast networks, or virtual links. This command also allows you to set the way in which OSPF routes are compared to information gained from other routing protocols.
Size	Monitoring	Displays the number of LSAs currently in the link state database, categorized by type.
Statistics	Monitoring	Displays OSPF statistics detailing memory and network usage.
Weight	Monitoring	Dynamically changes the cost of an OSPF interface.
Exit	Configuring and Monitoring	Exits the OSPF configuration or console process.

? (Help) C M

Lists the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example:

?

```
ADD
DELETE
DISABLE
ENABLE
EXIT
JOIN
LEAVE
LIST
SET
```

12.3 OSPF Configuration and Console Commands

Add C

Adds more information to already existing OSPF information. With this command, you can add ranges to areas as well as neighbors to non-broadcast networks.

Syntax: add

range . . .
neighbor . .

range ***area# IP-address IP-address-mask***

Adds ranges to OSPF areas. OSPF areas are defined in terms of address ranges. External to the area, a single route is advertised for each address range. For example, if an OSPF area consists of all subnets of the class B network 128.185.0.0, it is defined as consisting of a single address range. The address range is specified as an address of 128.185.0.0 together with a mask of 255.255.0.0. Outside of the area, the entire subnetted network is advertised as a single route to network 128.185.0.0.

Example:

```
add range 0.0.0.2 128.185.0.0 255.255.0.0
```

neighbor

Adds neighbors to non-broadcast networks. If the router is connected to a non-broadcast, multi-access network, such as an X.25 PDN, you have to use this command to help the router discover its OSPF neighbors. This configuration is only necessary if the router is eligible to become the designated router of the non-broadcast network. Configure the IP addresses of all other OSPF routers that are attached to the non-broadcast network. For each router configured, you must also specify its eligibility to become designated router.

Example:

```
add neighbor
Interface IP address [0.0.0.0]? 128.185.138.19
IP Address of Neighbor [0.0.0.0]? 128.185.138.21
Can that router become Designated Router [Yes]?
```

12.3 OSPF Configuration and Console Commands

Advertisement Expansion M

Prints the contents of a link state advertisement contained in the OSPF database. For a summary of the router's advertisements, use the **database** command.

A link state advertisement is defined by its link state type, link state ID, and its advertising router. There is a separate link state database for each OSPF area. Providing an *area-id* on the command line tells the software which database you want to search. Listed below are different kinds of advertisements that depend on the value given for link-state-type:

- **Router links** – Contain descriptions of a single router's interface.
- **Network links** – Contain the list of routers attached to a particular interface.
- **Summary nets** – Contain descriptions of a single inter-area route.
- **Summary AS boundary routers** – Contain descriptions of the route to an AS boundary router in another area.
- **AS external nets** – Contain descriptions of a single route.
- **Multicast group memberships** – Contain descriptions of a particular group's membership in the neighborhood of the advertising router.

Note: Link state IDs, advertising routers (specified by their router IDs), and area IDs take the same format as IP addresses. For example, the backbone area can be entered as 0.0.0.0.

The example that follows shows an expansion of a router links advertisement. The router's ID is 128.185.184.11. It is an AS boundary router and has three interfaces to the backbone area (all of cost 1). Multicast routing was enabled. Detailed field descriptions are provided with the example shown below.

This command was also enhanced in two ways. First, when displaying router-LSAs and network-LSAs, the reverse cost of each router-to-router link and router-to-transit-network link is displayed, as well as the previously displayed forward cost. This is done because routing of multicast datagrams whose source lies in different areas/ASs is based on reverse cost instead of forward cost. In those cases where there is no reverse link (which means that the link is never used by the Dijkstra), the reverse cost is shown as ‘‘1-way”.

In addition, the LSA's OSPF options are displayed in the same manner as they were displayed in the detailed OSPF **neighbor** command.

New group-membership-LSAs can also be displayed. An example follows. The ‘‘LS destination” of each group-membership-LSA is a group address. A router originates a group-membership-LSA for each group that has members on one or more of the router’s attached networks. The group-membership-LSA for the group lists those attached transit networks having group members (the type ‘‘2” vertices), and when there are members belonging to one or more attached stub networks, or if the router itself is a member of the multicast group, a type ‘‘1” vertex whose ID is the router’s OSPF router ID is included.

Syntax: advertisement *ls-type link-state-id [advertising-router] [area-id]*

Example:

```
advertisement 1 128.185.184.11 0.0.0.0
LS age:     173
LS options:  E,MC
LS type:     1
LS destination (ID): 128.185.184.11
LS originator:   128.185.184.11
LS sequence no:  0x80000047
LS checksum:     0x122
LS length:       60
Router type:  ASBR,W
# router ifcs:   3
         Link ID:          128.185.177.31
         Link Data:        128.185.177.11
         Interface type:   2
                  No. of metrics: 0
                  TOS 0 metric:   3 (0)
         Link ID:          128.185.142.40
         Link Data:        128.185.142.11
         Interface type:   2
                  No. of metrics: 0
                  TOS 0 metric:   4 (0)
         Link ID:          128.185.184.0
         Link Data:        255.255.255.0
         Interface type:   3
                   No. of metrics: 0
                   TOS 0 metric:   1
```

LS age Indicates the age of the advertisement in seconds.

LS options Indicates the optional OSPF capabilities supported by the piece of the routing domain described by the advertisement. These capabilities are denoted by E (processes type 5 externals. When this is not set to the area to which the advertisement belongs was configured as a stub), T (can route based on TOS) and MC (can forward IP multicast datagrams).

LS type Classifies the advertisement and dictates its contents: 1 (router links advertisement), 2 (network link advertisement), 3 (summary link advertisement), 4 (summary ASBR advertisement), 5 (AS external link) and 6 (group-membership advertisement).

LS destination Identifies what is being described by the advertisement. Depends on the advertisement type. For router links and ASBR summaries, it is the OSPF router ID. For network links, it is the IP address of the network's designated router. For summary links and AS external links, it is a network/subnet number. For group-membership advertisements, it is a particular multicast group.

LS originator OSPF router ID of the originating router.

LS sequence number Used to distinguish separate instances of the same advertisement. Is looked at as a signed 32-bit integer. Starts at 0x80000001, and increments by one each time the advertisement is updated.

LS checksum A checksum of advertisement contents, used to detect data corruption.

LS length The size of the advertisement in bytes.

Router type Indicates the level of functionality of the router. ASBR means that the router is an AS boundary router, ABR that the router is an area border router, and W that the router is a wildcard multicast receiver.

# Router ifcs The number of router interface described in the advertisement.

Link ID Indicates what the interface connects to. Depends on Interface type. For interfaces to routers (point-to-point links), the Link ID is the neighbor's router ID. For interfaces to transit networks, it is the IP address of the network designated router. For interfaces to stub networks, it is the network's network/subnet number.

Link Data 4 bytes of extra information concerning the link, it is either the IP address of the interface (for interfaces to point-to-point networks and transit networks), or the subnet mask (for interfaces to stub networks).

Interface type One of the following: 1 (point-to-point connection to another router), 2 (connection to transit network), 3 (connection to stub network) or 4 (virtual link).

No. of metrics The number of non-zero TOS values for which metrics are provided for this interface.

TOS 0 metric The cost of the interface. In parentheses, the reverse cost of the link is given (derived from another advertisement). If there is no reverse link, “1-way” is displayed.

The *LS age, LS options, LS type, LS destination, LS originator, LS sequence no, LS checksum* and *LS length* fields are common to all advertisements. The *Router type* and *# router ifcs* are seen only in router links advertisements. Each link in the router advertisement is described by the *Link ID, Link Data,* and *Interface type* fields. Each link can also be assigned a separate cost for each IP Type of Service (TOS). This is described by the *No. of metrics* and *TOS 0 metric fields* (the router currently does not route based on TOS, and only looks at the TOS 0 cost).

The next example shows an expansion of a group-membership advertisement. A group-membership advertisement for a given group/advertising router combination lists those networks directly attached to the advertising router that have group members. It also lists whether the router itself is a member of the specified group. The example below shows that network 128.185.184.0 has members of group 224.0.1.1.

Example:

```
adv 6 224.0.1.1 128.185.184.114
For which area [0.0.0.0]?
              LS age:     168
              LS options:  E
              LS type:     6
              LS destination (ID): 224.0.1.1
              LS originator:   128.185.184.114
              LS sequence no:  0x80000001
              LS checksum:     0x7A3
              LS length:       28
              Vertex type: 2
              Vertex ID:   128.185.184.114
```

Vertex type Describes the object having group members, one of: 1 (the router itself, or stub networks attached to the router) or 2 (a transit network).

Vertex ID When the vertex type is 1, always the advertising router’s ID. When the vertex type is 2, the IP address of the transit network’s designated router.

12.3 OSPF Configuration and Console Commands

Area Summary M

Display the statistics and parameters for all OSPF areas attached to the router.

In the example below, the router attaches to a single area (the backbone area). A simple password scheme is being used for the area’s authentication. The router has three interfaces attaching to the area, and has found 4 transit networks, 7 routers and no area border routers when doing the SPF tree calculation for the backbone.

Syntax: area

Example:

```
area
Area ID       Authentication   #ifcs  #nets  #rtrs  #brdrs
0.0.0.0        Simple-pass       3      4      7      0
```

# ifcs Indicates the number of router interfaces attached to the particular area. These interfaces are not necessarily functional.

# nets Indicates the number of transit networks that were found while doing the SPF tree calculation for this area.

# rtrs Indicates the number of routers that were found when doing the SPF tree calculation for this area.

# brdrs Indicates the number of area border routers that were found when doing the SPF tree calculation for this area.

12.3 OSPF Configuration and Console Commands

AS-external Advertisements, M

Lists the AS external advertisements belonging to the OSPF routing domain. One line is printed for each advertisement. Each advertisement is defined by the following three parameters: its link state type (always 5 for AS external advertisements), its link state ID (called the LS destination), and the advertising router (called the LS originator).

Syntax: as-external

Example:

```
as-external
Type LS destination LS originator   Seqno      Age   Xsum
  5  0.0.0.0       128.185.123.22  0x80000084 430   0x41C7
  5  128.185.131.0 128.185.123.22  0x80000080 450  0x71DC
  5  128.185.132.0 128.185.123.22  0x80000080 450  0x66E6
  5  128.185.144.0 128.185.123.22  0x80000002 329  0xF2CA
  5  128.185.178.0 128.185.123.22  0x80000081 450  0x72AA
  5  128.185.178.0 128.185.129.40  0x80000080 382  0xDD28
  5  129.9.0.0     128.185.123.22  0x80000082 451  0x4F30
  5  129.9.0.0     128.185.126.24  0x80000080 676  0x324A
  5  134.216.0.0   128.185.123.22  0x80000082 451  0x505A
  5  134.216.0.0   128.185.126.24  0x80000080 676  0x3374
  5  192.9.3.0     128.185.123.22  0x80000082 451  0xF745
  5  192.9.3.0     128.185.126.24  0x80000080 677  0xDA5F
  5  192.9.12.0    128.185.123.22  0x80000082 452  0x949F
  5  192.9.12.0    128.185.128.41  0x80000080 679  0x31B2
  5  192.26.100.0  128.185.123.22  0x80000081 452  0xFDCD
  5  192.26.100.0  128.185.126.24  0x80000080  21  0xDEE8
                              etc.
               # advertisements:             133
               Checksum total:               0x43CC41
```

Type Always 5 for AS external advertisements.

LS destination Indicates an IP network/subnet number. These network numbers belong to other Autonomous Systems.

LS originator Advertising router.

Seqno, Age, Xsum It is possible for several instances of an advertisement to be present in the OSPF routing domain at any one time. However, only the most recent instance is kept in the OSPF link state database (and printed by this command). The LS sequence number (Seqno), LS age (Age) and LS checksum fields (Xsum) are compared to see which instance is most recent. The LS age field is expressed in seconds. Its maximum value is 3600.

At the end of the display, the total number of AS external advertisements is printed, along with a checksum total over all of their contents. The checksum total is simply the 32-bit sum (carries discarded) of the individual advertisement's LS checksum fields. This information can be used to quickly determine whether two OSPF routers have synchronized databases.

Database Summary M

Displays a description of the contents of a particular OSPF area's link state database. AS external advertisements are omitted from the display. A single line is printed for each advertisement. Each advertisement is defined by the following three parameters: its link state type (called Type), its link state ID (called the LS destination) and the advertising router (called the LS originator).

Syntax: database *area-id*

Example:

```
database 0.0.0.0
Type LS destination LS originator    Seqno     Age  Xsum
  1  128.185.123.22 128.185.123.22 0x80000084  442 0xCE2D
  1  128.185.125.38 128.185.125.38 0x80000082  470 0x344D
  1  128.185.126.24 128.185.126.24 0x80000088 1394 0xCC47
  1  128.185.128.41 128.185.128.41 0x80000082  471 0x16A2
  1  128.185.129.25 128.185.129.25 0x8000008D 1624 0x8B64
  1  128.185.129.40 128.185.129.40 0x8000008A 1623 0xABBE
  1  128.185.136.39 128.185.136.39 0x80000082  469 0x5045
  2  128.185.125.40 128.185.129.40 0x80000049  457 0xA31
  2  128.185.126.25 128.185.129.25 0x80000002 1394 0x56B8
  2  128.185.127.24 128.185.126.24 0x8000007F 1031 0x592D
  2  128.185.129.25 128.185.129.25 0x8000005F 2295 0x8219
  2  128.185.129.40 128.185.129.40 0x80000001 1623 0x12C9
  6  224.0.2.6      128.185.142.9  0x8000003D  232 0x513F
  6  224.0.2.6      128.185.184.11 0x80000003  376 0x2250

                    # advertisements:        14
                      Checksum total:        0x4BBC2
```

12.3 OSPF Configuration and Console Commands

Type Separate LS types are numerically displayed: type 1 (router links advertisements), type 2 (network links advertisements), type 3 (network summaries), type 4 (AS boundary router summaries), and type 6 (group-membership-LSAs).

LS destination Indicates what is being described by the advertisement.

LS originator Advertising router.

Seqno, Age, Xsum It is possible for several instances of an advertisement to be presenting the OSPF routing domain at any one time. However, only the most recent instance is kept in the OSPF link state database (and printed by this command). The LS sequence number (Seqno), LS age (Age) and LS checksum fields (Xsum) are compared to see which instance is most recent. The LS age field is expressed in seconds. Its maximum value is 3600.

At the end of the display, the total number of advertisements in the area database is printed, along with a checksum total over all of their contents. The checksum total is simply the 32-bit sum (carries discarded) of the individual advertisement's LS checksum fields. This information can be used to quickly determine whether two OSPF routers have synchronized databases.

Note: When comparing multicast-capable to non-multicast routers, the above database checksum (and also # advertisements) will not necessarily match, because non-multicast routers do not handle or store group-membership-LSAs.

12.3 OSPF Configuration and Console Commands

Delete C

Deletes OSPF information from SRAM.

Syntax: delete

range . . .
area . . .
interface . . .
neighbor . . .
non-broadcast . . .
virtual link

range ***area# IP-address***

Deletes ranges from OSPF areas.

Example:

```
delete range 128.185.0.0 255.255.0.0
```

area ***area#***

Deletes OSPF areas from the current OSPF configuration.

Example:

```
delete area 0.0.0.1
```

interface ***interface-IP-address***

Deletes an interface from the current OSPF configuration.

Example:

```
delete interface 128.185.138.19
```

neighbor

Deletes neighbors on non-broadcast networks from the current OSPF configuration.

Example:

```
delete neighbor
Interface IP address [0.0.0.0]? 128.185.138.19
IP Address of Neighbor [0.0.0.0]? 128.185.138.21
```

non-broadcast ***interface-IP-address***

Deletes non-broadcast network information from the current OSPF configuration.

Example:

```
delete non-broadcast 128.185.133.21
```

virtual-link

Deletes a virtual link. Virtual links can be configured between any two backbone routers that have an interface to a common non-backbone area. Virtual links are used to maintain backbone connectivity and must be configured at both endpoints.

Example:

```
delete virtual-link
Virtual endpoint (Router ID) [0.0.0.0]?
Link's transit area [0.0.0.1]?
```

Disable C

Disables either the entire OSPF protocol or just the AS boundary routing capability.

Syntax: disable

as boundary routing
multicast
OSPF routing protocol

as boundary routing

Disables the AS boundary routing capability. When disabled, the router does not import external information into the OSPF domain.

Example:

```
disable as boundary routing
```

multicast

Disables IP multicast routing on all interfaces. When disabled, the router does not forward IP multicast (Class D) datagrams.

Example:

```
disable multicast
```

OSPF routing protocol

Disables the entire OSPF protocol.

Example:

```
disable OSPF routing protocol
```

Dump Routing Tables M

Displays all the routes that were calculated by OSPF and are now present in the routing table. Its output is similar in format to the IP console's **dump routing tables** command.

Syntax: dump

Example:

```
dump
Type   Dest net         Mask     Cost Age  Next hop(s)

Sbnt   16.0.0.0         FF000000  1    0    None
 SPF   16.24.8.0        FFFFFF00  2    2    20.24.12.230
 SPF   16.24.10.0       FFFFFF00  4    4    20.24.12.230
 SPF   16.24.11.0       FFFFFF00  2    2    20.24.12.230
Sbnt   17.0.0.0         FF000000  1    0    None
 SPF   17.1.1.0         FFFFFF00  4    4    20.24.12.230
Sbnt   18.0.0.0         FF000000  1    0    None
 SPF   18.1.1.0         FFFFFF00  4    4    20.24.12.230
Sbnt   20.0.0.0         FF000000  1    0    None
 SPF*  20.24.12.0       FFFFFF00  1    1    Eth/1
Sbnt   21.0.0.0         FF000000  1    0    None
 SPF*  21.24.16.0       FFFFFF00  1    1    Eth/5
 Dir*  21.24.166.0      FFFFFF00  1    0    Eth/5
 Dir*  21.24.167.0      FFFFFF00  1    0    Eth/5
 Dir*  21.24.168.0      FFFFFF00  1    0    Eth/5
 Dir*  21.24.169.0      FFFFFF00  1    0    Eth/5
 Dir*  21.24.170.0      FFFFFF00  1    0    Eth/5
Sbnt   25.0.0.0         FF000000  1    0    None
 SPF   25.24.13.0       FFFFFF00  4    4    20.24.12.230
Sbnt   135.24.0.0       FFFF0000  1    0    None
 SPF   135.24.10.0      FFFFFF00  4    4    20.24.12.230

Routing table size: 768 nets (55296 bytes), 21 nets known
```

Type Indicates destination type. Net indicates that the destination is a network. All other destinations are covered by the OSPF **routers** command.

- **Sbnt** – Indicates that the network is subnetted; such an entry is a placeholder only.
- **Dir** – Indicates a directly connected network or subnet.
- **RIP** – Indicates the route was learned through the RIP protocol.
- **Del** – Indicates the route was deleted.
- **Stat** – Indicates a statically configured route.
- **EGP** – Indicates routes learned through the EGP protocol.
- **EGPR** – Indicates routes learned through the EGP protocol that are readvertised by OSPF and RIP.
- **Fltr** – Indicates a routing filter.
- **SPF** – Indicates that the route is an OSPF intra-area route.
- **SPIA** – Indicates that it is an OSPF inter-area routes.
- **SPE1, SPE2** – Indicates OSPF external routes (types 1 and 2 respectively).
- **Rnge** – Indicates a route type that is an active OSPF area address range and is not used in forwarding packets.

Dest net Destination host or network.

Mask Displays the entry’s subnet mask.

Cost Age Displays the route cost.

Next hop(s) Address of the next router on the path toward the destination host. A number in parentheses at the end of the column indicates the number of equal-cost routes to the destination. The first hops belonging to these routes can be displayed with the IP console’s **route** command.

12.3 OSPF Configuration and Console Commands

Enable C

Enables either the entire OSPF protocol or just the AS boundary routing capability.

Syntax: enable

as boundary routing
multicast
ospf

as boundary routing

Enables the AS boundary routing capability that allows you to import routes learned from other protocols (EGP, BGP RIP, and statically configured information) into the OSPF domain. See Section 12.2 for more information about using this command.

Example:

```
enable as boundary routing
Import EGP routes? [No]: y
Import BGP routes? [No]:
Import RIP routes? [No]:
Import static routes? [No]:
Import direct routes? [No]: y
Import subnet routes? [No]:
Always originate default route? [No]:
Originate default if EGP/BGP routes available? [No]: y
   From AS number [0]? 12
   To network number [0.0.0.0]? 10.0.0.0
Originate as type 1 or 2 [2]?
Default route cost [1]?
Default forwarding address [0.0.0.0]?
```

multicast

Enables the forwarding of IP multicast (Class D) datagrams. When enabling multicast routing, you are also prompted whether you want to forward IP multicast datagrams between OSPF areas and between Autonomous Systems. To run MOSPF (OSPF with multicast extensions), a router currently running OSPF needs only to use this command. You do not need to re-enter its configuration information.

Example:

```
enable multicast
Inter-area multicasting enabled (Yes or No): yes
Inter-AS multicasting enabled (Yes or No): yes
```

12.3 OSPF Configuration and Console Commands

OSPF routing protocol

Enables the entire OSPF protocol. When enabling the OSPF routing protocol, you must supply the following two values that are used to estimate the size of the OSPF link state database:

- Total number of AS external routes imported into the OSPF routing domain. A single destination may lead to multiple external routes when it is imported by separate AS boundary routers. For example, if the OSPF routing domain has two AS boundary routers, both importing routes to the same 100 destinations, the number of AS external routes is set to 200.
- Total number of OSPF routers in the routing domain.

Example:

```
enable OSPF routing protocol
Estimated # external routes[0]? 200
Estimated # OSPF routers [0]? 60
```

Interface Summary M

Displays statistics and parameters related to OSPF interfaces. If no arguments are given, a single line is printed summarizing each interface. If an interface's IP address is given, detailed statistics for that interface are displayed.

Syntax: interface *interface-ip-address*

Example:

```
interface
Ifc Address      Phys    assoc. Area      Type   State   #nbrs   #adjs
16.24.8.251      Eth/1   0.0.0.0         Brdcst   64        1        1
16.24.11.251     Eth/1   0.0.0.0         Brdcst   64        1        1
17.1.1.251       Eth/2   0.0.0.0         Brdcst   64        0        0
25.24.13.251     Eth/3   0.0.0.0         Brdcst   64        0        0
18.1.1.251       Eth/4   0.0.0.0         Brdcst   64        0        0
16.24.10.251     Eth/0   0.0.0.0         Brdcst   64        0        0
135.24.10.251    Eth/0   0.0.0.0         Brdcst   64        0        0
```

Ifc Address Interface IP address.

Phys The physical interface.

Assoc Area Attached area ID.

Type Can be either Brdcst (broadcast, for example, an Ethernet interface), P-P (a point-to-point network, for example, a synchronous serial line), Multi (non-broadcast, multi-access, for example, an X.25 connection) and VLink (an OSPF virtual link).

State Can be one of the following: 1 (down), 2 (looped back), 4 (waiting), 8 (point-to-point), 16 (DR other), 32 (backup DR) or 64 (designated router).

#nbrs Number of neighbors. This is the number of routers whose hellos were received, plus those that were configured.

#adjs Number of adjacencies. This is the number of neighbors in state Exchange or greater. These are the neighbors with whom the router has synchronized or is in the process of synchronization.

Example:

```
interface 128.185.125.22
Interface address:      128.185.125.22
Attached area:          0.0.0.1
Physical interface:     Eth /1
Interface mask:         255.255.255.0
Interface type:         Brdcst
State:                  32
Designated Router:      128.185.184.34
Backup DR:              128.185.184.11

DR Priority:       1  Hello interval:   10  Rxmt interval:     5
Dead interval:    60  TX delay:          1  Poll interval:     0
Max pkt size:   1500  TOS 0 cost:        1

# Neighbors:       2  # Adjacencies:     2  # Full adjs.:      2
# Mcast floods: 1714  # Mcast acks:    856

MC forwarding:    on  DL unicast:      off  IGMP monitor:     on
# MC data in:  14444  # MC data acc: 13379  # MC data out: 16254
IGMP polls snt: 1316  IGMP polls rcv: 1009  Unexp polls:       9
IGMP reports:   2000  Nbr node: type:    2  ID:   128.185.184.34
```

Interface Address Interface IP address.

Attached Area Attached area ID.

Physical interface Displays physical interface type and number.

Interface Mask Displays interface subnet mask.

12.3 OSPF Configuration and Console Commands

Interface type Can be either Brdcst (broadcast, for example, an Ethernet interface), P-P (a point-to-point network, for example, a synchronous serial line), Multi (non-broadcast, multi-access, for example, an X.25 connection) and VLink (an OSPF virtual link).

State Can be one of the following: 1 (Down), 2 (Attempt), 4 (Init), 8 (2-Way), 16 (ExStart), 32 (Exchange), 64 (Loading) or 128 (Full).

Designated Router IP address of the designated router.

Backup DR IP address of the backup designated router.

DR Priority Displays priority assigned to designated router.

Hello interval Displays the current hello interval value.

Rxmt interval Displays the current retransmission interval value.

Dead interval Displays the current dead interval value.

TX delay Displays the current transmission delay value.

Poll interval Displays the current poll interval value.

Max pkt size Displays the maximum size for an OSPF packet sent out this interface.

TOS 0 cost Displays the interface’s TOS 0 cost.

# Neighbors Number of neighbors. This is the number of routers whose hellos were received, plus those that were configured.

# Adjacencies Number of adjacencies. This is the number of neighbors in state Exchange or greater.

# Full adj Number of full adjacencies. The number of full adjacencies is the number of neighbors whose state is Full (and therefore, with which the router has synchronized databases).

# Mcast Floods Number of link state updates flooded out the interface (not counting retransmissions).

# Mcast acks Number of link state acknowledgements flooded out the interface (not counting retransmissions).

MC forwarding Displays whether multicast forwarding was enabled for the interface.

DL unicast Displays whether multicast datagrams are to be forwarded as data-link multicasts or as data-link unicasts.

IGMP monitor Displays whether IGMP is enabled on the interface.

# MC data in Displays the number of multicast datagrams that were received on this interface and then successfully forwarded.

# MC data acc Displays the number of multicast datagrams that were successfully forwarded.

# MC data out Displays the number of datagrams that were forwarded out the interface (either as data-link multicasts or data-link unicasts).

IGMP polls sent Displays the number of IGMP Host Membership Queries that were sent out the interface.

IGMP polls rcv Displays the number of IGMP Host Membership Queries that were received on the interface.

Unexp polls Displays the number of unexpected IGMP Host Membership Queries that were received on the interface (received when the router itself was sending them).

IGMP reports Displays the number of IGMP Host Membership Reports received on the interface.

Nbr node: type and ID Displays the identity of the upstream node if the router is supposed to receive datagrams on this interface. *Type* here is an integer from 1 to 3, with 1 indicating router, 2 indicating transit net, and 3 indicating stub net.

Join C M

C Configures the router as a member of a multicast group. When the router is the member of a multicast group, it responds to pings and SNMP queries sent to the group address.

Group membership can also be obtained in a more temporary (and more immediate) way through the **join** command in the OSPF monitoring console.

M Establishes the router as a member of a multicast group.

This command is similar to the **join** command in the OSPF configuration console with two differences:

- The effect on group membership is immediate when the commands are given from the OSPF monitor (a restart/reload is not required).
- The command keeps track of the number of times a particular group is ‘‘joined.”

When the router is the member of a multicast group, it responds to pings and SNMP queries sent to the group address.

Syntax: join *multicast-group-address*

Example:

```
join  224.185.0.0
```

Leave C M

C Remove a router’s membership in a multicast group. This prevents the router from responding to pings and SNMP queries sent to the group address.

Group membership can also be deleted in a more temporary (and more immediate) fashion through the **leave** command in the OSPF monitoring console.

M Remove a router’s membership in a multicast group. This keeps the router from responding to pings and SNMP queries sent to the group address.

This command is similar to the **leave** command in the OSPF configuration console with two differences:

- The effect on group membership is immediate when the commands are given from the OSPF monitor (a restart/reload is not required).
- The command does not delete group membership until the “leaves” executed equals the number of “joins” previously executed.

Syntax: leave *multicast-group-address*

Example:

```
leave 224.185.0.0
```

List C

Displays OSPF configuration information.

Syntax: list

all
areas
interfaces
non-broadcast
virtual-links

all

Lists all OSPF related configuration information.

Example:

```
list all
                        --Global configuration--
                OSPF Protocol:          Enabled
                # AS ext. routes:       100
                Estimated # routers:    20
                External comparison:    Type 2
                AS boundary capability: Enabled
                Import external routes: RIP
                Orig. default route:    No (0,0.0.0.0)
                Default route cost:     (1, Type 2)
                Default forward. addr.: 0.0.0.0
                Multicast forwarding:   Disabled

                        --Area configuration--
Area ID          AuType          Stub? Default-cost Import-summaries?
0.0.0.0          0=None           No         N/A           N/A

                        --Interface configuration--
IP address        Area              Cost  Rtrns  TrnsDly  Pri  Hello  Dead
16.24.8.251       0.0.0.0              2      5       1     1     10    40
16.24.11.251      0.0.0.0              2      5       1     1     10    40
17.1.1.251        0.0.0.0              2      5       1     1     10    60
```

```
    18.1.1.251        0.0.0.0              2      5       1     1     10    60
    16.24.10.251      0.0.0.0              2      5       1     1     10    40
    135.24.10.251     0.0.0.0              2      5       1     1     10    40
    192.24.18.251     0.0.0.0              2      5       1     1     10    40
    25.1.1.251        0.0.0.0              2      5       1     1     10    60
```

OSPF protocol Displays whether OSPF is enabled or disabled.

# AS ext. routes Displays the estimated number of Autonomous System external routes. The router cannot accept more than this number of AS external routes.

Estimated # routers Displays the estimated number of routers found in the OSPF configuration.

External comparison Displays the external route type used by OSPF when importing external information into the OSPF domain and when comparing OSPF external routes to RIP/EGP routes.

AS boundary capability Displays whether the router imports external routes into the OSPF domain.

Import external Displays which routes are imported.

Orig default route Displays whether the router imports a default into the OSPF domain. When the value is “YES”, a non-zero network number is displayed in parentheses. This indicates that the default route originates if and only if a route to that network is available.

Default route cost Displays the cost and type that are used in the imported default route.

Default forward addr Displays the forwarding address that is used in the imported default route.

Multicast forwarding Displays whether IP multicast datagrams is forwarded.

Area-ID Displays the attached area ID (area summary information)

AuType Displays the method used for area authentication. “Simple-pass” means a simple password scheme is being used for the area’s authentication.

Stub area — Displays whether or not the area being summarized is a stub area. Stub areas do not carry an external route, resulting in a smaller routing database. However, stub areas cannot contain AS boundary routers, nor can they support configured virtual links.

Interface Configuration — For each interface, its IP address is printed, together with configured parameters. “Area” is the OSPF area to which the interface attaches. “Cost” indicates the TOS 0 cost (or metric) associated with the interface. “Rtrns” is the retransmission interval, which is the number of seconds between retransmissions of unacknowledged routing information. “TrnsDly” is the transmission delay, which is an estimate of the number of seconds it takes to transmit routing information over the interface (it must be greater than 0). “Pri” is the interface’s Router Priority, which is used when selecting the designated router. “Hello” is the number of seconds between Hello Packets sent over the interface. “Dead” is the number of seconds after Hellos cease to be heard that the router is declared down.

Virtual links — Lists all virtual links that were configured with this router as endpoint. “Virtual endpoint” indicates the OSPF Router ID of the other endpoint. “Transit area” indicates the non-backbone area through which the virtual link is configured. Virtual links are considered treated by the OSPF protocol similarly to point-to-point networks. The other parameters listed in the command (“Rtrns,” “TrnsDly,” “Hello,” and “Dead”) are maintained for all interfaces. See the OSPF **list interfaces** command for more information.

areas

Lists all information concerning configured OSPF areas.

Example:

```
list areas
Area ID:        0.0.0.0
Authentication: 1 (Simple-pass)
Stub area?:     FALSE

Area ID:        0.0.0.1
Authentication: 0 (None)
Stub area?:     FALSE
```

Area-ID Displays the attached area ID (area summary information).

Authentication Displays the method used for area authentication. “Simple-pass” means a simple password scheme is being used for the area’s authentication.

Stub area Displays whether or not the area being summarized is a stub area.

interfaces

For each interface, its IP address is printed, together with configured parameters. “Area” is the OSPF area to which the interface attaches. “Cost” indicates the TOS 0 cost (or metric) associated with the interface. “Rtrns” is the retransmission interval, which is the number of seconds between retransmissions of unacknowledged routing information. “TrnsDly” is the transmission delay, which is an estimate of the number of seconds it takes to transmit routing information over the interface (it must be greater than 0). “Pri” is the interface’s router priority, which is used when selecting the designated router. “Hello” is the number of seconds between Hello Packets sent out the interface. “Dead” is the number of seconds after Hellos cease to be heard that the router is declared down.

Example:

```
list interfaces
  OSPF interfaces:
  IP address        Area     Cost  Rtrns  TrnsDly  Pri Hello Dead
  128.185.177.11   0.0.0.0   3      5       1     0     10    60
  128.185.142.11   0.0.0.0   4      5       1     1     10    60
  128.185.184.11   0.0.0.0   1      5       1     1     10    60
```

non-broadcast

Lists all information related to interfaces connected to non-broadcast networks. For each non-broadcast interface, as long as the router is eligible to become designated router on the attached network, the polling interval is displayed together with a list of the router’s neighbors on the non-broadcast network.

Example:

```
list non-broadcast
Interface Addr      Poll Interval
128.185.235.34      120
```

virtual-links

Lists all virtual links that were configured with this router as endpoint. “Virtual endpoint” indicated the OSPF router ID of the other endpoint. “Transit area” indicates the non-backbone area through which the virtual link is configured. Virtual links are considered treated by the OSPF protocol similarly to point-to-point networks. The other parameters listed in the command (“Rtrns,” “TrnsDly,” “Hello,” and “Dead”) are maintained for all interfaces. See the OSPF **list interfaces** command for more information.

Example:

```
list virtual-links
Virtual links:
Virtual endpoint   Transit area   Rtrns  TrnsDly Hello  Dead
0.0.0.0            0.0.0.1         10        5     30    180
```

Mcache M

Displays a list of currently active multicast cache entries. Multicast cache entries are built on demand, whenever the first matching multicast datagram is received. There is a separate cache entry (and therefore a separate route) for each datagram source network and destination group combination.

Cache entries are cleared on topology changes (for example, a point-to-point line in the MOSPF system going up or down), and on group membership changes.

Syntax: mcache

Example:

```
mcache
                  MOSPF forwarding cache
Source          Destination     Upstream        #Down     Usage
128.185.142.0   224.0.1.1       128.185.142.11    0       184
```

Source Source network/subnet of matching datagrams.

Destination Destination group of matching datagrams.

Upstream Displays the neighboring network/router from which the datagram is received in order to be forwarded. When this reads as “none,” the datagram is never forwarded.

Down Displays the total number of downstream interfaces/neighbors to which the datagram is forwarded. When this is 0, the datagram is not forwarded.

Usage Displays the number of received datagrams that matched the cache entry.

There is more information in a multicast forwarding cache entry. A cache entry can be displayed in detail by providing the source and destination of a matching datagram on the command line. If a matching cache entry is not found, one is built. A sample of this command is shown below:

Example:

```
mcache 128.185.182.9  224.0.1.2
source Net:     128.185.182.0
Destination:    224.0.1.2
Use Count:      472
Upstream Type:  Transit Net
Upstream ID:    128.185.184.114
Downstream:     128.185.177.11 (TTL = 2)
```

In addition to the information shown in the short form of the **mcache** command, the following fields are displayed:

Upstream Type Indicates the type of node from which the datagram must be received in order to be forwarded. Possible values for this field are “none” (indicating that the datagram is not forwarded), “router” (indicating that the datagram must be received over a point-to-point connection), “transit network,” “stub network,” and “external” (indicating that the datagram is expected to be received from another Autonomous System).

Down-stream Prints a separate line for each interface or neighbor to which the datagram is sent. A TTL value is also given, indicating that datagrams forwarded out of or to this interface must have at least the specified TTL value in their IP header. When the router is itself a member of the multicast group, a line specifying “internal Application” appears as one of the downstream interfaces/neighbors.

Mgroups M

Displays the group membership of the router’s attached interfaces. Only the group membership for those interfaces on which the router is either designated router or backup designated router are displayed.

Syntax: mgroups

Example:

```
mgroups
                Local Group Database

  Group              Interface                  Lifetime (secs)

 224.0.1.1         128.185.184.11 (Eth /1)         176
 224.0.1.2         128.185.184.11 (Eth /1)         170
 224.1.1.1         Internal                        1
```

Group Displays the group address as it was reported (through IGMP) on a particular interface.

Interface Displays the interface address to which the group address was reported (through IGMP).

The router’s internal group membership is indicated by a value of “internal.” For these entries, the lifetime field (see below) indicates the number of applications that requested membership in the particular group.

Lifetime Displays the number of seconds that the entry persists if Membership Reports cease to be heard on the interface for the given group.

12.3 OSPF Configuration and Console Commands

Mstat M

Displays various multicast routing statistics. The command indicates whether multicast routing is enabled and whether the router is an inter-area and/or inter-AS multicast forwarder.

Syntax: mstats

Example: **`mstats`**

```
                MOSPF forwarding:               Disabled
                Inter-area forwarding:          Disabled
                DVMRP forwarding:               Disabled

    Datagrams received:            0  Datagrams (ext source):           0
    Datagrams fwd (multicast):     0  Datagrams fwd (unicast):          0
    Locally delivered:             0  No matching rcv interface:        0
    Unreachable source:            0  Unallocated cache entries:        0
    Off multicast tree:            0  Unexpected DL multicast:          0
    Buffer alloc failure:          0  TTL scoping:                      0

    # DVMRP routing entries:       0  # DVMRP entries freed:            0
    # fwd cache alloc:             0  # fwd cache freed:                0
    # fwd cache GC:                0  # local group DB alloc:           0
    # local group DB free:         0
```

MOSPF forwarding Displays whether the MOSPF is running (*Enabled*) or not (*Disabled*).

Inter-area forwarding Displays whether the router forwards IP multicast datagrams between areas.

DVMRP forwarding Displays whether the DVMRP is running (*Enabled*) or not (*Disabled*).

Datagrams received Displays the number of multicast datagrams received by the router (datagrams whose destination group lies in the range 224.0.0.1 - 224.0.0.255 are not included in this total).

Datagrams (ext source) Displays the number of datagrams received whose source is outside the AS.

Datagrams fwd (multicast) Displays the number of datagrams forwarded as data-link multicasts. (This includes packet replications, when necessary, so this count can be greater than the number received.)

Datagrams fwd (unicast) Displays the number of datagrams forwarded as data-link unicasts.

Locally delivered Displays the number of datagrams forwarded to internal applications.

No matching rcv interface Displays the count of those datagrams received by a non-inter-AS multicast forwarder on a non-MOSPF interface.

Unreachable source Displays a count of those datagrams whose source address was unreachable.

Unallocated cache entries Displays a count of those datagrams whose cache entries were not created due to resource shortages.

Off multicast tree Displays a count of those datagrams not forwarded either because there was no upstream neighbor or no downstream interfaces/neighbors in the matching cache entry.

Unexpected DL multicast Displays a count of those datagrams that were received as data-link multicasts on those interfaces that were configured for data-link unicast.

Buffer alloc failure Displays a count of those datagrams that could not be replicated because of buffer shortages.

TTL scoping Indicates those datagrams that were not forwarded because their TTL indicated that they could never reach a group member.

#DVMRP routing entries Indicates the number of DVMRP routing table entries currently in use.

#DVMRP entries freed Indicates the number of DVMRP routing table entries that have been freed.

# fwd cache alloc Indicates the number of cache entries allocated. The current forwarding cache size is the number of entries allocated ("# fwd cache alloc") minus the number of cache entries freed ("# fwd cache freed").

# fwd cache freed Indicates the number of cache entries freed. The current forwarding cache size is the number of entries allocated ("# fwd cache alloc") minus the number of cache entries freed ("# fwd cache freed").

# fwd cache GC Indicates the number of cache entries were cleared because they were not recently used and the cache overflowed.

# local group DB alloc Indicates the number of local group database entries allocated. The number allocated (“# local group DB alloc”) minus the number freed (“# local group DB free”) equals the current size of the local group database.

# local group DB free Indicates the number of local group database entries freed. The number allocated (“# local group DB alloc”) minus the number freed (”# local group DB free”) equals the current size of the local group database.

The number of cache hits can be calculated as the number of datagrams received (“Datagrams received”) minus the total of datagrams discarded due to “No matching rcv interface,” “Unreachable source” and “Unallocated cache entries,” and minus “# local group DB alloc.” The number of cache misses is simply “# local group DB alloc.”

Neighbor Summary M

Displays statistics and parameters related to OSPF neighbors. If no arguments are given, a single line is printed summarizing each neighbor. If a neighbor’s IP address is given, detailed statistics for that neighbor are displayed.

Syntax: neighbor *neighbor-ip-address*

Example:

```
neighbor
Neighbor addr   Neighbor ID     State  LSrxl  DBsum  LSreq  Ifc
128.185.125.39  128.185.136.39  128      0     0       0    Eth/1
```

Neighbor addr Displays the neighbor address.

Neighbor ID Displays the neighbor’s OSPF router ID.

Neighbor State Can be one of the following: 1 (Down), 2 (Attempt), 4 (Init), 8 (2-Way), 16 (ExStart), 32 (Exchange), 64 (Loading) or 128 (Full).

LSrxl Displays the size of the current link state retransmission list for this neighbor.

DBsum Displays the size of the database summary list waiting to be sent to the neighbor.

LSreq Displays the number of more recent advertisements that are being requested from the neighbor.

Ifc Displays the interface shared by the router and the neighbor.

Example:

`neighbor 128.185.138.39`

The meaning of most of the displayed fields is given in Section 10 of the OSPF specification (RFC 1131).

```
Neighbor IP address:    128.185.184.34
OSPF Router ID:         128.185.207.34
Neighbor State:         128
Physical interface:     Eth /1
DR choice:              128.185.184.34
Backup choice:          128.185.184.11
DR Priority:            1
Nbr options:            E,MC

DB summ qlen:      0  LS rxmt qlen:      0  LS req qlen:       0
Last hello:        7
# LS rxmits:     108  # Direct acks:    13  # Dup LS rcvd:   572
# Old LS rcvd:     2  # Dup acks rcv:  111  # Nbr losses:     29
# Adj. resets:    30
```

Neighbor IP addr Neighbor IP address.

OSPF router ID Neighbor’s OSPF router ID.

Neighbor State Can be one of the following: 1 (Down), 2 (Attempt), 4 (Init), 8 (2-Way), 16 (ExStart), 32 (Exchange), 64 (Loading), or 128 (Full).

Physical interface Displays physical interface type and number of the router and neighbor’s common network.

DR choice, backup choice, DR priority Indicate the values seen in the the last hello received from the neighbor.

Nbr options Indicates the optional OSPF capabilities supported by the neighbor. These capabilities are denoted by E (processes type 5 externals. When this is not set, the area to which the common network belongs was configured as a stub), T (can route based on TOS) and MC (can forward IP multicast datagrams). This field is valid only for those neighbors in state Exchng or greater.

DBsumm qlen Indicates the number of advertisements waiting to be summarized in Database Description packets. It is zero except when the neighbor is in state Exchange.

LS rxmt qlen Indicates the number of advertisements that were flooded to the neighbor, but not yet acknowledged.

LS req qlen Indicates the number of advertisements that are being requested from the neighbor in state Loading.

Last hello Indicates the number of seconds since a hello was received from the neighbor.

# LS rxmits Indicates the number of retransmissions that occurred during flooding.

# direct acks Indicates responses to duplicate link state advertisements.

# Dup LS rcvd Indicates the number of duplicate retransmissions that occurred during flooding.

# Old LS rcvd Indicates the number of old advertisements received during flooding.

# Dup acks rcvd Indicates the number of duplicate acknowledgements received.

# Nbr losses Indicates the number of times the neighbor transitioned to Down state.

# Adj. resets Counts entries to state ExStart.

Routers M

Displays all router routes that were calculated by OSPF and are now present in the routing table. With the **dump routing tables** command, the *Net* field indicates that the destination is a network. The **routers** command covers all other destinations.

Syntax: routers

Example:

```
routers
DType RType Destination       Area          Cost       Next hop(s)

  BR  SPF   20.24.12.230     0.0.0.0         1         20.24.12.230
Fadd  SPF   20.24.12.230     0.0.0.0         1         0.0.0.2
  BR  SPF   16.24.8.251      0.0.0.0         2         20.24.12.230
ASBR  SPIA  19.24.9.252      0.0.0.0         3         20.24.12.230
```

DType Indicates destination type. Net indicates that the destination is a network, ASBR indicates that the destination is an AS boundary router, and BR indicates that the destination is an area border router, and Fadd indicates a forwarding address (for external routes).

RType Indicates route type and how the route was derived. SPF indicates that the route is an intra-area route (comes from the Dijkstra calculation), SPIA indicates that it is an inter-area route (comes from considering summary link advertisements).

Destination Destination router's OSPF ID. For Type D entries, one of the router's IP addresses is displayed (which corresponds to a router in another AS).

Area Always displayed as 0.0.0.0.

Cost Displays the route cost.

Next hop Address of the next router on the path toward the destination host. A number in parentheses at the end of the column indicates the number of equal-cost routes to the destination.

12.3 OSPF Configuration and Console Commands

Set C

Displays or changes the configuration information concerning OSPF areas, interfaces, non-broadcast networks, or virtual links. This command also allows you to set the way in which OSPF routes are compared to information obtained from other routing protocols.

Syntax: set

area
comparison
interface
non-broadcast
virtual-links

area

Sets the parameters for an OSPF area. If no areas are defined, the router software assumes that all the router's directly attached networks belong to the backbone area (area ID 0.0.0.0).

Example:

```
set area
Area number [0.0.0.0]? 0.0.0.1
Authentication type [1]? 1
Is this a stub area? (Yes or No): no
```

- **Area number is the OSPF area address.** An OSPF area is a contiguous group of networks that is defined by a list of address ranges, each indicated by a combination of the IP address and an address mask. A network belongs to an area if its address is in the list.
- **Authentication type (security scheme) to be used in the area.** The choices for authentication types are 1, which indicates a simple password; or 0, which indicates that no authentication is necessary to pass packets.
- **Stub area designation.** If you designate YES:
  - The area does not receive any AS external link advertisements, reducing the size of your database and decreasing memory usage for routers in the stub area.
  - You cannot configure virtual links through a stub area.

- – You cannot configure a router within the stub area as an AS boundary routers.
- – You get two further prompts:

  ```
  Stub default cost [0]?
  Import summaries? [yes]
  ```

- – Answering **yes** to import summaries (inter-area routes) allows summary routes to be flooded into the router.

External Routing in Stub Areas. You cannot configure the backbone as a stub area. External routing in stub areas is based on a default route. Each border area router attaching to a stub area originates a default route for this purpose. The cost of this default route is also configurable with the `OSPF config>` **set area** command.

comparison

Tells the router where the EGP/RIP/static routes fit in the OSPF hierarchy. The two lower levels consist of the OSPF internal routes. OSPF internal routes take precedence over information gained from any other sources, all of which are located on a single level.

Example:

```
OSPF Config> set comparison
Compare to type 1 or 2 externals [2]?
```

interface

Sets the OSPF parameters for the router’s network interfaces.

Example:

```
OSPF Config> set interface
Attaches to area [0.0.0.0]? 0.0.0.1
Interface IP address [0.0.0.0]? 128.185.138.19
Retransmission Interval (in seconds) [5]?
Transmission Delay (in seconds) [1]? 1
Router Priority [1]? 1
Hello Interval (in seconds) [10]? 10
Dead Router Interval (in seconds) [60]? 40
Type Of Service 0 cost [1]? 5
Authentication Key []? xyz_q
Retype Auth.  Key []? xyz_q
Forward multicast datagrams (Yes or No)? Yes
Forward as data-link unicasts (Yes or No)? No
IGMP polling interval (in seconds) [60]? 60
IGMP timeout (in seconds) [180]? 180
```

12.3 OSPF Configuration and Console Commands

When responding to the prompts, supply the IP address for each interface in the router and answer the questions that follow. For the parameters listed, you must enter the same value for all routers attached to a common network.

- Hello interval
- Dead router interval
- Authentication key (if an authentication of 1 is used)

The first prompt asks for the OSPF area to which the interface attaches. For example, suppose that the interface address mask is 255.255.255.0, indicating that the interface attaches to a subnet (128.185.138.0) of network 128.185.0.0. All other OSPF routers attached to subnet 128.185.138.0 must also have their *hello interval* set to 10, *dead router interval* set to 40, and their interface *authentication key* set to xyz_q.

In a multicast routing configuration (multicast was enabled), the MOSPF parameters for each OSPF interface are set to their default values. This means the following:

- Multicast forwarding is enabled.
- Multicast datagrams are forwarded as data-link multicasts.
- IGMP Host Membership is sent out the interface every 60 seconds.
- Local group database entries are removed 180 seconds after IGMP Host Membership reports for the group cease to be received by the interface.

If you want to change the MOSPF parameters, use the **set interface** command. You are queried for multicast parameters (the last five parameters shown in the output display below) only if you first enable multicast forwarding.

On networks that lie on the edge of an Autonomous System, where multiple multicast routing protocols (or multiple instances of a single multicast routing protocol) may exist, you may need to configure forwarding as data-link unicasts to avoid unwanted datagram replication. In any case, for all routers attached to a common network, the interface parameters *forward multicast datagrams* and *forward as data-link unicasts* are configured identically.

12.3 OSPF Configuration and Console Commands

non-broadcast

Helps the router discover its OSPF neighbors. This configuration is only necessary if the router is eligible to be the designated router of the non-broadcast network. After using this command you must then configure the IP addresses of all other OSPF routers that are attached to the non-broadcast network. See the **add neighbor** command for more information.

Example:

```
set non-broadcast
Interface IP address [0.0.0.0]? 128.185.138.19
Poll Interval [120]?
```

virtual-link

Configures virtual links between any two area border routers. To maintain backbone connectivity, you must have all of your backbone routers interconnected either by permanent or virtual links. Virtual links are used to connect an area border router to the backbone area through an intermediate area if no direct connection is possible. Therefore, you are asked to also specify many of the interface parameters when configuring a virtual link.

Note: If you use the **set virtual-link** command to create a virtual connection to the backbone area, you must also use the **set area 0.0.0.0** command to configure the backbone area.

Example:

```
set virtual link
Virtual endpt.  (Router ID) [0.0.0.0]? 128.185.138.21
Link’s transit area [0.0.0.1]? .0.0.1
Retransmission Interval (in seconds) [10]?
Transmission Delay (in seconds) [5]?
Hello Interval (in seconds) [30]?
Dead Router Interval (in seconds) [180]?
Authentication Key []? 3-14159
```

Size M

Displays the number of LSAs currently in the link state database, categorized by type.

Syntax: size

Example:

```
size
# Router-LSAs:           7
# Network-LSAs:          6
# Summary-LSAs:          14
# Summary Router-LSAs:   2
# AS External-LSAs:      44
# Group-membership-LSAs: 21
```

Statistics M

Displays statistics generated by the OSPF routing protocol. The statistics indicate how well the implementation is performing, including its memory and network utilization. Many of the fields displayed are confirmation of the OSPF configuration.

Syntax: statistics

Example:

```
statistics
S/W version:            2.1
OSPF Router ID:         128.185.184.11
External comparison:    Type 2
AS boundary capability: Yes
Import external routes: EGP RIP STA DIR SUB
Orig. default route:    No (0,0.0.0.0)
Default route cost:     (1, Type 2)
Default forward. addr:  0.0.0.0

Attached areas:                 2  Estimated # external routes:    300
Estimated # OSPF routers:     100  Estimated heap usage:         76000
OSPF packets rcvd:          60822  OSPF packets rcvd w/ errs:    28305
Transit nodes allocated:     1728  Transit nodes freed:           1715
LS adv. allocated:           7394  LS adv. freed:                 7313
Queue headers alloc:          224  Queue headers avail:            224

# Dijkstra runs:              391  Incremental summ. updates:        0
Incremental VL updates:         0  Buffer alloc failures:            0
Multicast pkts sent:        49487  Unicast pkts sent:              557
LS adv. aged out:               0  LS adv. flushed:                521
```

S/W version — Displays the current OSPF software revision level.

OSPF Router ID — Displays the router’s OSPF ID.

External comparison — Displays the external route type used by the router when importing external routes.

AS boundary capability — Displays whether external routes are imported.

Import external routes — Displays which external routes are imported.

Orig default route — Displays whether the router will advertise an OSPF default route. If the value is Yes and a non-zero number is displayed in parentheses, then a default route is advertised only when a route to the network exists.

Default route cost — Displays the cost and type of the default route (if advertised).

Default forward addr — Displays the forwarding address specified in the default route (if advertised).

Attached areas — Indicates the number of areas that the router has active interfaces to.

Estimated heap usage — Rough indication of the size of the OSPF link state database (in bytes).

Transit nodes — Allocated to store router links and network links advertisements.

LS adv. — Allocated to store summary link and AS external link advertisements.

Queue headers — Form lists of link state advertisements. These lists are used in the flooding and database exchange processes; if the number of queue headers allocated is not equal to the number freed, database synchronization with some neighbor is in progress.

# Dijkstra runs — Indicates how many times the OSPF routing table was calculated from scratch.

Incremental summ updates, incremental VL updates — Indicate that new summary link advertisements caused the routing table to be partially rebuilt.

Buffer alloc failures. — Indicate buffer allocation failures. The OSPF system recovers from temporary lack of packet buffers.

Multicast pkts sent — Covers OSPF hello packets and packets sent during the flooding procedure.

Unicast pkts sent — Covers OSPF packet retransmissions and the Database Exchange procedure.

LS adv. aged out — Counts the number of advertisements that have hit 60 minutes. Link state advertisements are aged out after 60 minutes. Usually, they are refreshed before this time.

LS adv. flushed — Indicates number of advertisements removed (and not replaced) from the link state database.

Incremental ext. updates. — Displays number of changes to external destinations that are incrementally installed in the routing table.

Weight M

Changes the cost of one of the routers OSPF interfaces. This new cost is immediately flooded throughout the OSPF routing domain, causing routes to be updated accordingly.

The cost of the interface will revert to its configured cost whenever the router is restarted or reloaded. To make the cost change permanent, you must reconfigure the appropriate OSPF interface after invoking the **weight** command. This command causes a new router links advertisement to originate, unless the cost of the interface does not change.

Syntax: weight *ip-interface-address* *new-cost*

Example:

```
weight 128.185.124.22 2
```

Exit C M

Returns to the previous prompt level.

Syntax: exit

Example:

```
exit
```

13

Configuring and Monitoring PIM

This chapter describes how to configure and monitor Protocol Independent Multicast (PIM) in the router.

For more information about Protocol Independent Multicast, refer to the *Routing Protocols Reference Guide.*

13.1 Configuring for Protocol Independent Multicasting

13.1.1 Before You Begin

Before you configure PIM on your router, you must have the following:

- The IP addresses of all the interfaces on which you will run PIM. These interfaces must be configured for IP, using an appropriate unicasting protocol (for example, OSPF).
- The IP addresses of all groups that will be multicast using Sparse Mode PIM and the IP addresses for all of their respective Rendezvous Point (RP) routers.
- The timer intervals, timeouts, and threshold data rates that will apply throughout the PIM network.

Note: It is important that the RP address lists and PIM timer parameters are configured to be the same throughout the network; otherwise, the effects on PIM routing will be unpredictable.

13.1.2 Dense Mode PIM Configuration Procedure

This section describes the steps required to configure Dense Mode PIM on one router. This procedure must be repeated at every PIM router in the network.

13.1 Configuring for Protocol Independent Multicasting

The following steps assume that the interfaces that either connect to another PIM router or are connected to host systems (directly or over a LAN) have already been configured as IP interfaces:

1. At the `Config>` prompt, enter **protocol PIM** to display the `PIM Config>` prompt.

   ```
   Config>protocol pim
   PIM Config>
   ```

2. Use the **set interface** command to set the PIM query interval and query timeout values for each PIM interface, and to set the IGMP polling interval and timeout for interfaces connected to a LAN. The interface is identified by its *IP-address.* Repeat this command for each interface.

   It is recommended that you use the **default** values for these timers.

   ```
   PIM Config>set interface
   Interface IP address [0.0.0.0]? IP-address
   Query interval (seconds) [30]?
   Query timeout (seconds) [90]?
   Prune/Join interval (seconds) [60]?
   Prune/Join timeout (seconds) [180]?
   IGMP polling interval (seconds) [60]?
   IGMP timeout (seconds) [180]?
   ```

3. Optionally, set the (S,G) entry timeout value. Use the **set pim** command if you want to change this value from the default. Only the SG entry timeout value is used in PIM Dense Mode. All the other parameters relate to PIM Sparse Mode and can be ignored.

   It is recommended that you use the **default** value for the (S,G) timeout interval.

   ```
   PIM Config>set pim
   SPT switchover threshold (pkt/s) [-1]?
   Encapsulation termination threshold (pkt/s) [-1]?
   RP-reachability interval (seconds) [90]?
   RP-reachability timeout (seconds) [270]?
   RP wait timeout (seconds) [180]?
   RP poll interval (seconds) [60]?
   SG entry timeout (seconds) [180]? 240
   ```

4. Use the **enable pim protocol** command to allocate sufficient memory for the routing tables and enable PIM.

It is recommended that you use the **default** values for the system estimates unless you know you will require more groups or shortest path trees.

```
PIM Config>enable pim protocol
Estimated number of groups [500]?
Estimated number of shortest path trees [50]?
Average number of OIFs per SG [8]
```

13.1.3 Sparse Mode PIM Configuration Procedure

This section describes the steps required to configure Sparse Mode PIM on one router. This procedure must be repeated at every PIM router in the network.

The following steps assume that the interfaces that either connect to another PIM router or are connected to host systems (directly or over a LAN) have already been configured as IP interfaces:

1. At the `Config>` prompt, enter **protocol PIM** to display the `PIM Config>` prompt.

   ```
   Config>protocol pim
   PIM Config>
   ```

2. Use the **set group** command to specify which groups will be multicast using Sparse Mode PIM, and to define the RP list for each group. Repeat this command for each group or range of groups. Ensure that these groups and RP lists are identical at every PIM router in the network.

   ```
   PIM Config>set group
   Group address [0.0.0.0]: group-address
   Mask length [32]: group-mask-length
   RP address [0.0.0.0]: ip-address
   RP address [0.0.0.0]:
   ```

3. Use the **set interface** command to set the PIM query and Prune/Join intervals and their timeout values for each PIM interface, and to set the IGMP polling interval and timeout for interfaces connected to a LAN. Repeat this command for each interface.

   It is recommended that you use the **default** values for these timers.

   ```
   PIM Config>set interface
   Interface IP address [0.0.0.0]? IP-address
   Query interval (seconds) [30]?
   Query timeout (seconds) [90]?
   Prune/Join interval (seconds) [60]?
   Prune/Join timeout (seconds) [180]?
   IGMP polling interval (seconds) [60]?
   IGMP timeout (seconds) [180]?
   ```

4. Optionally, set the source tree switchover thresholds, RP timers and (S,G) entry timeout value. Use the **set pim** command if you want to change these value from the default settings

   It is recommended that you use the **default** values for all of these parameters throughout the network.

```
PIM Config>set pim
SPT switchover threshold (pkt/s) [-1]?
Encapsulation termination threshold (pkt/s) [-1]?
RP-reachability interval (seconds) [90]?
RP-reachability timeout (seconds) [270]?
RP wait timeout (seconds) [180]?
RP poll interval (seconds) [60]?
SG entry timeout (seconds) [180]?
```

5. Use the **enable pim protocol** command to allocate sufficient memory for the routing tables and enable PIM.

   It is recommended that you use the **default** values for the system estimates unless you know you will require more groups or shortest path trees.

```
PIM Config>enable pim protocol
Estimated number of groups [500]?
Estimated number of shortest path trees [50]?
Average number of OIFs per SG [8]
```

13.2 Accessing the Interface Configuration Process

You can access the PIM configuration commands by typing **protocol PIM** at the `Config>` prompt, for example:

Example:

```
Config> protocol PIM
```

For more information about accessing the PIM configuration environment, see Chapter 1.

Note: After you access the configuration process, you may begin entering configuration commands. Whenever you make a change to a user-configurable parameter, you must restart the router for this change to take effect.

13.3 Accessing the PIM Console Environment

You can access the PIM montioring commands by typing **protocol PIM** at the + prompt, for example:

Example:

```
+ protocol PIM
PIM>
```

For more information about accessing the PIM console environment, see Chapter 1.

13.4 PIM Configuration and Console Commands

Table 13–1 summarizes the PIM configuration and console commands. The sections that follow explain these commands. Enter the configuration commands at the `PIM Config>` prompt.

The monitoring capabilities include the following: configured parameters such as interface addresses and RP lists can be viewed, the current state of the PIM routing tables can be displayed and PIM statistics such as the number of packets sent and received, and the memory used can be listed. Enter the console commands at the `PIM>` prompt.

Table 13–1 PIM Configuration and Console Command Summary

Command	Tasks	Function
? (Help)	Configuring and Monitoring	Displays all the PIM protocol configuration commands or lists the options for specific commands.
Delete	Configuring	Deletes all PIM group and interface configuration parameters.
Disable	Configuring	Disables PIM protocol.
Dump	Monitoring	Displays the (S,G) entries in the PIM routing table.
Enable	Configuring	Enables PIM protocol.
Interface	Monitoring	Displays the PIM interface statistics and parameters.
Join	Configuring and Monitoring	Issues a request for the router to join a multicast group. This is used for testing purposes only.
Leave	Configuring and Monitoring	Issues a request for the router to leave a multicast group. This is used for testing purposes only.
List	Configuring	Lists all information related to the PIM protocol parameters and options.

Table 13–1 PIM Configuration and Console Command Summary (Continued)

Command	Tasks	Function
Mgroups	Monitoring	Displays the group membership of the router's attached interfaces.
Neighbor	Monitoring	Displays the current state of all PIM neighbors.
Ping	Monitoring	Sends ICMP Echo Requests to another host once a second and watches for a response. This command can be used to isolate trouble in an internetworking environment.
Routes	Monitoring	Lists the contents of the IP (unicast) routing table. This command is equivalent to the **dump** command in the IP console environment.
Set	Configuring	Sets PIM parameters.
Stat	Monitoring	Displays various PIM routing statistics.
Traceroute	Monitoring	Displays the complete path (hop by hop) to a particular destination.
Exit	Configuring and Monitoring	Exits the PIM configuration process and returns to the `Config>` prompt.

Help (?) C M

Lists the commands that are available from the current prompt level. You can also enter **?** after a specific command name to list its options.

Syntax: ?

Example:

```
? C
DELETE
DISABLE
ENABLE
EXIT
JOIN
LEAVE
LIST
SET
```

***Example:*?** M

```
DUMP source group
EXIT
INTERFACE summary
JOIN
LEAVE
MGROUPS
NEIGHBOR summary
PING address
ROUTES summary
STAT
TRACEROUTE address
```

Example:

```
list ?
ALL
```

Delete C

Deletes a group or an interface definition.

Syntax: delete

group . . .
interface . . .

group

Deletes a group entry from the group configuration table. This only applies to Sparse Mode PIM.

Example:

```
delete group
Group address [0.0.0.0]? 226.0.0.0
Mask length [32]?
```

interface

Deletes an interface entry from the interface configuration table.

Example:

```
delete interface
Interface IP address [0.0.0.0]? 16.36.16.122
```

13.4 PIM Configuration and Console Commands

Disable C

Disables the PIM protocol globally at system startup.

Syntax: disable
pim

Example:

```
disable pim
```

Dump M

Displays the (S,G) entries in the routing table. The **dump** command can be used to give a summary listing of the PIM routing table or a detailed display of selected entries.

Syntax: dump *source-address group-address*

If there are no arguments to the **dump** command, it displays a summary of all of the (S,G) entries in the routing table.

Example:

```
dump
SG Database summary, States are (G)ood, (B)ad and (N)egative,
                     Tree flags are (!)Register, (*)RP-Annotated.

State/Source         Group              IIF     Tree   OIF(s)
G 17.4.4.191         224.1.1.1          Eth/0   DM     PPP/1,PPP/0
G 17.4.4.192         224.1.1.1          PPP/0   DM     PPP/1,PPP/0
G 16.36.16.122       225.1.1.1          Eth/1   RP!    Int,PPP/0,PPP/1

Total of 3 SG entries, in 2 groups; RP for 0 groups
.
```

State Displays the current state of this (S,G) tree:

- **G**(ood) – The tree is usable.
- **B**(ad) – The RP router is not reachable (PIM-SM only).
- **N**(egative) – Packets will be discarded. Used to prune the branch of a source tree (PIM-SM only)

Source Identifies the source of the multicast datagrams. In a Shared Tree this field will be an asterisk (*) wildcard, otherwise it is the IP address of the Host system (S), which is multicasting to this group.

Group

Displays the IP address of the multicast group (G), in the range 224.0.0.0 to 239.255.255.255

Note: The Source and Group fields together uniquely identify the (S,G) or (*,G) entry in the routing table.

IIF

The network interface identifier of input interface (IIF). This is the interface that the (S,G) datagrams are expected to arrive on.

Tree

Indicates the type of tree that this (S,G) entry is part of.

- DM, Dense Mode source tree
- RP, Sparse Mode shared tree
- SPT, Sparse Mode source tree

Flags

Flag characters next to the tree field are used in PIM-SM to indicate the following conditions:

- **!** Register. Encapsulate incoming datagrams and unicast the register packets to the RP.
- ***** RP Annotated. Indicates that the RP has joined the source tree for this (S,G), so there is no need to unicast the datagrams.

These flags are only used in DRs, which forward multicast datagrams from source Hosts to the RP.

OIF(s)

Displays all the network interfaces in the output interface (OIF) list for the (S,G) entry.

The *Int* interface is an internal pseudo-interface created by the **join** command to indicate that the packet is to be delivered internally.

Total

SG entries

Displays the total number of (S,G) entries in the routing table.

groups

Displays the total number of unique groups in the routing table. There may be multiple sources for any multicast group.

RP for

Indicates the number of groups for which this router is currently the Rendezvous Point (PIM-SM only).

13.4 PIM Configuration and Console Commands

source-address group-address

Displays detailed information about a particular (S,G) entry. The information displayed depends on the mode and status of the (S,G) entry.

This is a typical Dense Mode entry:

Example:

```
dump 17.4.4.192 224.1.1.2
     Detailed information for SG entry  17.4.4.192,224.1.1.2

     Tree:            DM
     Input IF:        Eth/0
     Output IF(s):    Int(0),PPP/1(125)
     Pruned IF(s):    PPP/0(55)
     Lifetime:        112 (seconds)
     NH address:      17.6.6.196
```

This is a typical Sparse Mode entry:

Example:

```
dump * 239.0.0.30
     Detailed information for SG entry  *,239,0,0,30

     Tree:           RP
     Input IF:       PPP/1
     Output IF(s):   Eth/0(10)
     Lifetime:       170 (seconds)
     Join time:      93 (seconds)
     RP list:        *4.4.4.44,3.3.3.33,2.2.2.22
     NH address:     30.30.40.2
     NH PJ due:      7 (seconds)
     Group flags:    SM
     SG flags:       WC,RP,Join
```

Tree Indicates the type of tree that this (S,G) entry is part of:

- DM, Dense Mode source tree
- RP, Rendezvous Point Sparse Mode shared tree
- SPT, Sparse Mode source tree

Input IF Indicates the network interface identifier of the input interface (IIF). This is the interface that the (S,G) datagrams are expected to arrive on.

Output IF(s)

Displays all the active network interfaces in the output interface (OIF) list for the (S,G) entry. The number in brackets after each OIF is the number of seconds remaining before that interface times out.

The timeout value on the Int interface is zero, indicating that it will expire as soon as the physical interfaces on the OIF list expire. If PIM was configured on this router with a permanent **join** command then the timer will *never* timeout.

Pruned IF(s)

Dense Mode only. Displays all the network interfaces in the output interface (OIF) list that have been pruned. The number in brackets is the number of seconds remaining before the interface is reinstated.

Lifetime

Displays the number of seconds remaining before this (S,G) entry times out and is deleted. It is based on the (S,G) entry timeout interval, and only starts counting down when all of the OIF entries have been deleted.

Join time

Sparse mode only. Indicates the time left (in seconds) before this router will send packets onto the LAN.

If there are multiple routers connected to LAN, only the router with the highest IP address will issue Join requests. The others will suppress sending Join requests, and reset this countdown timer to 60 seconds each time they detect a Join from a router with a higher IP address. If that router becomes inactive, then this router will resume issuing Join request packets when this timer expires.

A value of 0 indicates that this router has not suppressed Join requests.

RP lifetime

Sparse mode only. Indicates the time left (in seconds) before this router assumes that the current RP router is not reachable. This value is reset when an RP reachability message is received.

RP list

Sparse mode only. The list of Rendezvous Point IP addresses for this (S,G) group. The entry marked by an asterisk (*) is the active RP.

NH address

Displays the IP address of the PIM router at the next hop (NH) towards the source of the tree. This is the upstream neighbor.

NH PJ due — Sparse mode only. Indicates the time interval (in seconds) before the next Join request packet is sent.

Group flags — Indicates the current status of the (S,G) group as follows:

- *Deleted* — Indicates that this (S,G) entry is in the process of being deleted.
- *SM* — Indicates that this is a Sparse Mode Group. This flag is omitted for Dense Mode groups.
- *RPlocal* — Indicates that this router is the Rendezvous Point for this (S,G). (Sparse Mode only)
- *RPseen* — Indicates that the RP reachability messages have been received from the RP for this group. (Sparse Mode only).

SG flags — Sparse mode only. Indicates the current status of the (S,G) entry, as follows:

- *WC* — Indicates that this is a Wild Card (*,G) entry, shared by all sources for this multicast group.
- *RP* — Indicates that this (S,G) entry is on the RP shared tree.
- *Join* — If set, indicates that this router is the one that is issuing Join Requests. This is the usual state. If this flag is not set it indicates Join requests are suppressed.
- *SPT* — Indicates that this entry is on the Shortest Path Tree (SPT) for the source.
- *Sel* — Indicates that this entry is scheduled for deletion. There is a short delay before actually deleting the entry in case it is reactiviated.
- *Ann* — RP Annotated. Indicates that the Rendezvous Point has joined the Shortest Path Tree to the source for this group.

  If set, it indicates that this router does not need to encapsulate data to register with the router, since the most efficient route is over the Shortest Path Tree.
- *Reg* — Registering. Indicates that this router is encapsulating data and registering it with the RP.

Internal

Indicates that there is an INTERNAL interface for this group (either from a permanent **Join** command, or required for another protocol on the router).

This means that the (S,G) entry will be kept open for internal delivery of the data, even if all external connections are closed.

Enable C

Enables the PIM protocol at system startup and sets memory requirements for the PIM routing tables based on the estimated number of groups, shortest path trees, and output interfaces that will be used.

Do not underestimate any of these requirements or the router may run out of memory.

Syntax: enable pim

Example:

```
enable pim
Estimated number of groups [500]?
Estimated number of shortest path trees [50]?
Average number of OIFs per SG [8]
```

Estimated number of groups

Specifies the maximum number of multicasts groups that may be active at any one time.

Range: 0 to 65535
Default: 500

Estimated number of shortest path trees

Specifies the maximum number of shortest path trees that may be active at any one time. In Dense Mode PIM, you must allow for at least one tree per multicast group.

In Sparse Mode PIM you must allow for the number of **source-based** trees that this router may switch to.

Range: 0 to 65535
Default: 50

Average number of OIFs per SG Specifies the average number of output interfaces that will appear in the OIF list of the (S,G) entries in the shortest path trees.

This number cannot exceed the number of interfaces configured on the router.

Range: 0 to 65535
Default: 8

Interface M

Displays the definition and status of the PIM interfaces. You can either display a summary listing of all of the interfaces or a detailed listing of a selected interface.

Syntax: interface

ip-address
physical-identifier

To get a summary listing of the interfaces, use the **interface** command with no parameters.

Example:

```
interface
If Address       Phys    State  DR               #nbrs
17.0.1.193       Eth/0   Up      17.0.1.196       1
17.1.1.193       PPP/0   Up      17.1.1.194       1
17.5.5.193       PPP/1   Up     *17.5.5.193       1
```

If Address Displays the IP address of the network interface.

Phys Displays the interface name and its instance number.

State Displays the state of the interface (Up or Down).

DR Displays the IP address of the Designated Router on this interface. The DR is the PIM router with the highest IP address on this . If the DR is this router, then the address is marked with an asterisk (*).

#nbrs Displays the number of neighboring PIM routers on this interface.

ip-address

To get a detailed listing of a particular interface, specify the IP address or the physical network identifier for the interface.

Example:

```
interface 17.5.5.193
             Physical interface:     PPP/1
             Interface address:      17.5.5.193
             Designated Router:      17.5.5.193

Query interval:     30   Query timeout:     90   IGMP interval:      60
IGMP timeout:      180

# Neighbors:         1   PIM pkts snt:   63288   PIM pkts rcv:    63572
Mcast pkts fwd:  50874   Mcast pkts rcv:   262

IGMP polls snt:  31643   IGMP polls rcv:     0   Unexp polls:         0
IGMP reports:        0
```

Physical interface Displays the interface name and its instance number.

Interface Address Displays the IP address of the network interface.

Designated Router Displays the IP address of the Designated Router on this interface. The DR is the interface with the highest IP address.

Query interval Displays the interval, in seconds, at which PIM Query messages are sent on this interface.

Query timeout Displays the timeout period if no response is received to a PIM Query message. If this timer expires it indicates that there is no neighboring PIM router available on this interface.

IGMP interval Displays the interval at which IGMP Host-Query messages are sent on this interface.

IGMP timeout Displays the timeout period if no response is received to an IGMP Host-Query message. If this timer expires it indicates that there are no group members on this interface.

# Neighbors Displays the number of neighboring PIM routers that have responded to PIM Query messages on this interface.

PIM pkts snt Displays the number of PIM protocol packets that have been sent on this interface since system startup.

13.4 PIM Configuration and Console Commands

PIM pkts rcv Displays the number of PIM protocol packets that have been received on this interface since system start-up.

Mcast pkts fwd Displays the number of multicast datagram packets that have been forwarded over this interface.

Mcast pkts rcv Displays the number of multicast datagram packets that have been received over this interface.

IGMP polls snt Displays the number of IGMP Host-Query packets that have been sent on this interface.

IGMP polls rcv Displays the number of IGMP Host-Query packets that have been received over this interface.

Unexp polls Displays the number of unexpected polls that have been received over this interface.

Only the DR should send polling messages. Sometimes during transitional state changes a router may send polls, but should stop as soon as it detects the elected DR. This counter should be very low.

IGMP reports Displays the number of IGMP Host-Report messages received over this interface.

Join C M

C Sets permanent group membership for the router at system startup. This membership is not affected by host systems joining or leaving the group.

This command is used for testing purposes only.

Syntax: join

Example:

```
join
Group address [0.0.0.0]? 226.1.1.3
```

M Requests the router to join a multicast group. It adds the group to the group membership table. For PIM-SM groups it will also send a join request to the RP. When you use the **join** command a pseudo-interface called Internal, or Int, is used as the output interface for the group member.

This command is used for testing purposes only. For example, you can use it to force group membership at one PIM router. You can then use the **ping** command at another PIM router to ping the multicast group. This generates multicast data traffic for the group and you can then examine the results at the PIM routers in the network using the PIM monitoring commands.

Syntax: join *group-address*

Example:

```
join
Group address [0.0.0.0]? 224.1.1.3
```

Leave C M

C Removes permanent group membership previously configured using the **join** command.

Syntax: leave

Example:

```
leave
Enter the address to be deleted [0.0.0.0]? 226.1.1.3
```

M Forces the router to leave a multicast group. It sends a Prune request upstream and deletes the group from the group membership table.

This command is used for testing purposes only. It can force a branch of a shared path tree to be pruned.

Syntax: leave

Example:

```
leave
Enter the address to be deleted [0.0.0.0]? 226.1.1.3
```

List C

Displays information related to PIM parameters and options.

Syntax: list all

Lists all options and parameters related to the PIM protocol.

Example:

```
list all
```

```
                        --Global configuration--
                PIM Protocol:             Enabled
                Estimated max SPTs:       50
                Estimated max groups:     500
                Average OIF per SG:       8
                SPT switchover:           10 (pkt/s)
                Encap termination:        10 (pkt/s)
                RP-reachability interval: 90
                RP-reachability timeout:  270
                RP poll interval:         60
                SG entry timeout:         180

                        --Groups configuration--
Group range          RP-List (4 max)
224.0.0.0/ 4         17.4.4.192      17.4.4.191

                        --Interface configuration--
IP address        QPoll     QTmo      PJPoll    PJTmo     IPoll    ITmo
16.36.16.122      30        90        60        180       60       180
17.6.6.122        30        90        60        180       60       180

                        --Group membership--
        224.0.0.0
```

Note: This example shows all possible options and parameters.

The following section explains the information displayed by the **list all** command:

Global configuration

PIM Protocol Displays whether the PIM protocol has been **enabled** or **disabled**.

Estimated max SPTs Indicates the maximum number of source-based shortest path trees (SPTs) that can be used by this router at any one time.

Estimated max groups Indicates the maximum number of multicast groups that can be active at any one time.

Average OIF per SG Indicates the average number of interfaces in the output interface (OIF) list for each (S,G) entry in the routing table.

SPT switchover Indicates the threshold data rate, in packets per second, at which the router will switch from using a Shared Tree to a Source Tree.

This applies to Sparse Mode PIM only.

Encap termination Indicates the threshold data rate, in packets per second, for a Rendezvous Point (RP) router to initiate the switch from receiving unicast encapsulated data to using source tree multicasting for a particular source.

This applies to Sparse Mode PIM only.

RP-reachability interval Indicates the interval, in seconds, that RP-Reachability Messages are sent out down the shared tree if this router becomes an RP.

This applies to Sparse Mode PIM only.

RP-reachability timeout Indicates the timeout interval, in seconds, for receiving an RP-Reachability Message.

This applies to Sparse Mode PIM only.

RP poll interval Indicates the interval, in seconds, at which an alternate RP sends PIM Poll messages to the network to see if a preferred RP router becomes available.

This applies to Sparse Mode PIM only.

SG entry timeout Indicates the interval, in seconds, before an (S,G) entry in the routing table is deleted.

Groups configuration Indicates the Rendezvous Point (RP) list for all multicast groups that will use Sparse Mode PIM.

Group range Displays the IP address of the multicast group and mask length if a range of addresses is specified. The two fields are separated by a ’/’ character.

RP-List Displays the list of IP addresses of the primary RP and up to three alternative RPs. The addresses are in order of preference.

Interface configuration Indicates the PIM polling timers and timeouts for each interface to a PIM router or host system.

IP address Displays the IP address of the network interface.

QPoll Indicates the interval, in seconds, at which to broadcast a PIM Query message on every PIM interface.

QTmo Indicates the timeout interval, in seconds, for receiving a PIM Query Message from a neighboring PIM router.

PJPoll Indicates the interval, in seconds, at which to broadcast Prune/ Join messages as Join requests to upstream sources for each (S,G) or (*,G) entry in its routing table.

PJTmo Indicates the timeout interval, in seconds, for receiving Prune/ Join requests from a downstream source.

IPoll Indicates the interval, in seconds, at which a DR sends IGMP Host-Query packets to directly connected hosts.

ITmo Indicates the timeout interval, in seconds, for a DR wait to receive IGMP Host-Report messages in response to an IGMP Host-Query message.

Group membership Indicates the list of multicast groups for which the router has hosts that are group members.

Mgroups M

Displays the group membership list for this router.

Syntax: mgroups

Example:

mgroups

```
           Local Group Database
Group            Interface                Lifetime (secs)
224.1.1.1        Internal                    1
224.1.1.1        Eth/0                      55
```

Local Group Database

Group Displays the IP address of the multicast group.

Interface Displays the interface which has members of this group.

If you force group membership using the **join** command it will use *Internal* as the pseudo interface for the group.

Lifetime (secs) Displays the number of seconds remaining before the group membership on this interface is due to expire. This is refreshed when an IGMP Host-Report is received on this interface for this group.

Neighbor M

Displays a summary of all of the neighboring PIM routers.

Syntax: neighbor

Example:

```
neighbor
 Nbr Address      Life   If
*17.0.1.196       70     Eth/0
*17.1.1.194       74     PPP/0
 17.5.5.192       75     PPP/1
```

Nbr Address Displays the IP address of the neighboring PIM router. The address is flagged with an asterisk (*) if the neighbor is the Designated Router on this interface.

Life Displays the number of seconds remaining for the neighboring PIM router to respond to the last PIM Query message. If this timer expires, the neighbor is assumed to have gone down.

If Displays the local interface that connects to the neighbor.

Ping M

Has the router send ICMP Echo Requests to a given destination once a second and watch for the response.

When a multicast address is given as the destination, there may be multiple responses printed for each packet sent, one for each group member. Each returned response is displayed with the source address of the responder.

Note: This command is identical to the **ping** command in the IP console. It is included in the PIM console for convenience when monitoring PIM routers. For more information about the **ping** command, see Chapter 9.

Syntax: ping *destination-address*

Example:

```
ping 224.1.1.2
PING 224.1.1.2: 56 data bytes
64 bytes from 17.6.6.122: icmp_seq=0. time=0. ms
64 bytes from 16.36.16.191: icmp_seq=0. time=16. ms
64 bytes from 17.5.5.192: icmp_seq=0. time=16. ms
64 bytes from 17.4.4.191: icmp_seq=0. time=16. ms
64 bytes from 16.36.16.103: icmp_seq=0. time=33. ms
64 bytes from 16.36.16.203: icmp_seq=0. time=33. ms

----224.1.1.2 PING Statistics----
1 packets transmitted, 6 packets received
```

Routes M

Displays the IP routing table. A separate entry is printed for each reachable IP network/subnet.

Note: This command is equivalent to the **dump** command in the IP console. It is included in the PIM console for convenience when monitoring PIM routers. For more information about this command, see the description of the **dump** command in Chapter 9.

Syntax: routes

Example:

```
routes
Type   Dest net         Mask       Cost     Age   Next hop(s)
SPE2   0.0.0.0          00000000   5        6     17.4.4.192
Sbnt   16.0.0.0         FF000000   1        0     None
SPE2   16.36.0.0        FFFFFF00   5        6     17.4.4.192
SPE2   16.36.0.1        FFFFFFFF   5        6     17.4.4.192
Sbnt   17.0.0.0         FF000000   1        0     17.4.4.192
 SPF   17.0.1.0         FFFFFF00   4        4     17.4.4.192
                          .
                          .

Default gateway in use.
Type Cost Age  Next hop
SPE2 5    6    17.4.4.192

Routing table size: 768 nets (55296 bytes), 130 nets known
```

13.4 PIM Configuration and Console Commands

Set C

Sets PIM parameters for this router. Every neighboring PIM router must be configured with identical timers and thresholds. In Sparse Mode PIM, the group configurations must be identical throughout the network.

Note: Values immediately following the command option prompts reflect the current setting of that option. They are not always the default values.

Syntax: set

```
                group . . .
                interface . . .
                pim
```

group

Maps multicast groups to the list of Rendezvous Point routers for PIM Sparse Mode.

Example:

```
set group
Group address [0.0.0.0]: 224.0.0.0
Mask length [32]: 32
RP address [0.0.0.0]: 17.4.4.194
RP address [0.0.0.0]: 17.4.4.193
RP address [0.0.0.0]: 17.4.4.192
RP address [0.0.0.0]: 17.4.4.191
```

Group address Displays the IP address of the first multicast group in the range of groups.

IP addresses in the range 224.0.0.0 to 239.255.255.255 are reserved for multicast groups.

Mask length Specifies the length of the mask to be applied to the group address to determine the range of groups being mapped. A full-length IP mask (32) maps just one group.

Range: 3 to 32. Default: 32.

RP address

Displays the list of IP addresses for the RPs mapped to this range of multicast groups. There must be at least one RP address, and no more than four RPs in the list. The addresses must be in order of preference, with the primary RP first.

If there are fewer than four RPs in the list, a null entry will terminate the list (press the Return key).

interface

Configures the polling timers and timeout intervals for the interfaces to neighboring PIM routers or to host systems on the local LAN.

Example:

```
set interface
Interface IP address [0.0.0.0]? 16.36.16.122
Query interval (seconds) [30]?
Query timeout (seconds) [90]?
Prune/Join interval (seconds) [60]?
Prune/Join timeout (seconds) [180]?
IGMP polling interval (seconds) [60]?
IGMP timeout (seconds) [180]?
```

Interface IP address

Specifies the IP address of the neighboring PIM router. The IP address must already have been configured on one of the network interfaces. Specifying the IP address identifies the interface.

Query interval

Specifies the interval, in seconds, at which to broadcast a PIM Query message on every PIM interface. The PIM Query message is used to tell other PIM routers that this is a PIM router.

PIM Query messages establish the PIM router neighbors. If there are multiple PIM routers on a LAN, the PIM Query polls are used to determine which is the Designated Router (DR) for the LAN. The DR is the active PIM router with the highest IP address.

Range: 0 to 65535. Default: 30.

Query timeout

Specifies the timeout interval, in seconds, for receiving a PIM Query Message from a neighboring PIM router.

There is a separate timer for each neighboring PIM router, which is refreshed by receiving a PIM Query message from that router. If the timeout expires, the neighboring router is considered to be down.

If a DR goes down, neighboring routers on the LAN will elect a new DR when the associated timer expires.

Range: 0 to 65535.
Minimum: twice the Query interval.
Default: three times the Query interval.

Prune/Join interval

Specifies the interval, in seconds, at which to broadcast Prune/Join messages as Join requests to upstream sources for each (S,G) or (*,G) entry in its routing table.

Sparse Mode PIM uses regular Join requests to repeatedly assert group membership, refreshing the link towards the source and keeping the (S,G) shortest path tree alive.

Range: 0 to 65535.
Default: 60.

Prune/Join Timeout

Specifies the timeout interval in seconds for receiving Prune/Join requests from a downstream source.

This timer is used by Sparse Mode PIM to automatically prune branches if group members stop sending join requests.

There is a separate Prune/Join timeout timer for each OIF entry in the OIF list for every (S,G) or (*,G) entry in the routing table.

If the timeout for an OIF expires, the router will delete the OIF from the entry in its routing table and stop forwarding multicast packets to that destination.

Range: 0 to 65535.
Minimum: twice the Prune/Join interval.
Default: three times the Prune/Join interval.

IGMP polling interval

Specifies the interval, in seconds, at which a DR sends IGMP Host-Query packets to directly connected hosts.

An IGMP Host-Query packet is transmitted periodically by DRs to ask host systems to report which multicast groups they are members of. An IGMP Host-Report packet is transmitted by hosts in response to these queries, indicating their group membership. Hosts only respond if they are members of at least one multicast group.

Range: 0 to 65535.
Default 60.

IGMP timeout

Specifies the timeout interval, in seconds, for a DR to receive IGMP Host-Report messages in response to an IGMP Host-Query message. If the timeout expires and the DR has not received any IGMP Host-Reports, then it assumes that it has no multicast group members and can delete all group information.

Range: 0 to 65535.
Minimum: twice the IGMP polling interval.
Default: three times the IGMP polling interval.

pim

Sets global timers and thresholds for PIM Sparse Mode, and sets (S,G) entry timer for both Sparse and Dense Modes.

Example:

```
set pim
SPT switchover threshold (pkt/s) [-1]? 10
Encapsulation termination threshold (pkt/s) [-1]? 10
RP-reachability interval (seconds) [90]?
RP-reachability timeout (seconds) [270]?
RP poll interval (seconds) [60]?
SG entry timeout (seconds) [180]?
```

SPT switchover

Specifies the threshold data rate, in packets per second, at which the router will switch from using a Shared Tree to a Source Tree.

Range: -1 to 65535.

-1 means **never** switch over.

0 means **always** switch over.

1 to 65535 threshold data rate.

Default: -1.

Note: SPT switchover is **disabled** by default. If you want to enable SPT switchover to Source Tree multicasting, you must define an appropriate threshold data rate.

Use of Source Trees may improve data throughput in certain circumstances, but maintaining those trees increases the protocol traffic.

This applies to Sparse Mode PIM only.

Encap termination

Specifies the threshold data rate, in packets per second, for a Rendezvous Point (RP) router to initiate the switch from receiving unicast encapsulated data to using Source Tree multicasting for a particular source. If the threshold is exceeded, the RP sends a Join request to the source DR.

Range: -1 to 65535.

-1 means **never** switch over.

0 means **always** switch over.

1 to 65535 threshold data rate.

Default: -1.

Note: Encap termination is **disabled** by default. If you want to enable switchover from unicasting to the RP to multicasting, you must define an appropriate threshold data rate.

Use of Source Trees may improve data throughput in certain circumstances, but maintaining those trees increases the protocol traffic.

This applies to Sparse Mode PIM only.

13.4 PIM Configuration and Console Commands

RP-reachability interval Specifies the interval, in seconds, that RP-Reachability Messages are sent out down the Shared Tree if this router becomes an RP.

Range: 1 to 65535
Default: 90

This applies to Sparse Mode PIM only.

RP-reachability timeout Specifies the timeout interval, in seconds, for receiving an RP-Reachability Message. This timer is refreshed when an RP-Reachability Message or multicast datagram is received from the RP. If the timeout expires, the router will attempt to join the next RP in the RP list.

Range: 2 to 65535.
Minimum: twice the RP-reachability polling interval.
Default: three times the RP-reachability polling interval.

This applies to Sparse Mode PIM only.

RP poll interval Specifies the interval, in seconds, at which an alternate RP sends PIM Poll messages to the network to determine when a preferred RP router becomes available.

Range: 1 to 65535
Default: 60

This applies to Sparse Mode PIM only.

SG entry timeout Specifies the interval, in seconds, before an (S,G) entry in the routing table is deleted if there is not activity on that tree.

In Dense Mode PIM, this timer is refreshed by data arriving at the router for this (S,G) entry.

In Sparse Mode PIM this timer specifies the interval that the (S,G) entry is kept in a Source Tree after all of the output interfaces have been pruned off. OIFs are pruned when downstream members either leave the group or revert to the Shared Tree.

Keeping the (S,G) entry alive for this period may reduce the amount of routing changes necessary.

Range: 1 to 65535
Default: 180

Stat M

Displays statistics of memory usage, data throughput, and error rates.

Example:

```
stat

             Protocol version:     2
             State:                Enabled
             SPT switchover:       10 (pkt/s)
             Encap termination:    10 (pkt/s)
             Garbage Collector:    Dormant

Estimated # groups:             500  Estimated # SPTs:              50
GE in use:                        1  GE free:                      499
SG in use:                        1  SG free:                      549
OIF in use:                       1  OIF free:                    4399
Dynamic heap usage:             452  Total memory usage:        218252
Interfaces:                       2  Neighbors:                      2
Groups:                           1  SG entries:                     1
Next hops:                        1  RP lists:                       0

PIM pkts sent:               112226  PIM pkt send fails:             0
PIM pkts rcvd:               111506  PIM pkts rcvd with errors:      0
Multicast pkts fwd:            1147  Multicast pkts rcvd:        25393
Multicast pkts encap:             0  Multicast pkts decap:           0
Multicast pkt/s:                  0  Max multicast pkt/s:            2

Total errors:                     0
```

Protocol version Displays the version of PIM protocol that has been implemented. This is currently PIM V2.

State Displays the current state of PIM, either **enabled** or **disabled**.

SPT switchover Displays the configured threshold for switching from shared tree to source tree for multicast routing.

Encap termination Displays the configured threshold for this router to switch from receiving encapsulated, registered data unicast from the source DR, to receiving the data using Source Tree multicasting.

Garbage Collector Displays the current state of the garbage collector housekeeping routine. The garbage collector is used to recover memory allocated to deleted routing information.

Estimated # groups
Displays the configured estimate of the maximum number of groups that will be used.

Estimated # SPTs
Displays the configured estimate of the maximum number of shortest path trees that will be used.

GE in use
Displays the total number of group entries that are currently in use.

GE free
Displays the total number of group entries that are not allocated. This is the difference between the *Estimated # groups* and the *GE in use* values.

SG in use
Displays the total number of (S,G) entries that are currently in use.

SG free
Displays the total number of (S,G) entries that are not allocated.

The total number of (S,G) entries is *Estimated # groups* plus *Estimated # SPTs*. The number of free (S,G) entries is this total less the *SG in use* value.

OIF in use
Displays the total number of output interfaces (OIFs) currently allocated in the (S,G) entries.

OIF free
Displays the number of OIF slots available.

There are eight OIF slots reserved for each (S,G) entry (*Estimated #groups* plus *Estimated # SPTs*).

Dynamic heap usage
Displays memory reserved for internal use.

Total memory usage
Total memory reserved for PIM routing tables and information.

Interfaces
Displays the number of interfaces configured for PIM usage.

Neighbors
Displays the total number of neighboring PIM routers.

Groups
Displays the number of groups currently in use.

SG entries
Displays the number of (S,G) entries that currently exist in the routing tables.

Next hops
Displays the number of unique PIM routers upstream on the active Routing Trees. If a PIM router is the source for several groups, then it counts as one Next Hop router.

RP lists Displays the number of RP mapping entries, indicating the number of groups that are using Sparse Mode PIM.

PIM pkts sent Displays the total number of PIM protocol packets sent by this router.

PIM pkt send fails Displays the total number ofPIM protocol packets that the router was unable to send.

PIM pkts rcvd Diplays the total numer of PIM protocol packets received by this router.

PIM pkts rcvd with errors Displays the total number of PIM protocol packets received with checksum errors.

Multicast pkts fwd Displays the total number of multicast datagrams that have been forwarded by this router.

Multicast pkts rcvd Displays the total number of multicast datagrams that have been received by this router.

Multicast pkts encap Displays the total number of multicast datagrams that have been encapsulated in registration packets and unicast to the corresponding RP.

This only applies to the DR routers near the Source Host when using Sparse Mode PIM.

Multicast pkts decap Displays the total number of encapsulated datagrams that have been received at this router.

This only applies to the RP routers in Sparse Mode PIM.

Multicast pkt/s Displays the number of multicast datagrams processed in the previous second.

Max multicast pkt/s Displays the maximum number of multicast datagrams processed in a second since system startup.

Total errors The total number of errors that have been observed since system startup.

If the *Total errors* counter is nonzero, then one or more of the following error counters will also be displayed:

SG allocation failures Unable to allocate memory for a new (S,G) entry.

Use the `PIM Config>` **set** command to increase the *Estimated number of SPT* entries.

GE allocation failures Unable to allocate memory for a new group entry.

Use the `PIM Config>` **set** command to increase the *Estimated number of groups*.

OIF allocation failures Unable to allocate memory for a new output interface.

Use the `PIM Config>` **set** command to increase the *Average number of OIFs***.**

NH allocation failures Unable to allocate dynamic memory for a new next hop router.

You must free up memory by reducing allocations either for PIM routing tables or for other configurations.

IF allocation failures Unable to allocate dynamic memory for a new PIM interface.

You must free up memory by reducing allocations either for PIM routing tables or for other configurations.

PDU allocation failures Unable to allocate dynamic memory for a new packet.

You must free up memory by reducing allocations either for PIM routing tables or for other configurations.

NBR allocation failures Unable to allocate dynamic memory for a new neighboring router.

You must free up memory by reducing allocations either for PIM routing tables or for other configurations.

RP allocation failures Unable to allocate memory for a new Rendezvous Point router.

You must free up memory by reducing allocations either for PIM routing tables or for other configurations.

RP list mismatches When a neighboring router issues a Join request packet, it includes a list of the RP routers it is configured for.

Check the RP lists on this and all neighboring routers if this counter increments.

Traceroute M

Displays the entire path to a given destination, hop by hop.

Note: This command is identical to the **traceroute** command in the IP console. It is included in the PIM console for convenience when monitoring PIM routers. For more information about the **traceroute** command, see Chapter 9.

Syntax: traceroute *interface-address*

Example:

```
traceroute 17.4.4.191
TRACEROUTE 17.4.4.191: 56 data bytes
 1 17.6.6.196 0 ms 0 ms 0 ms
 2 17.0.1.193 16 ms 0 ms 0 ms
 3 17.5.5.192 16 ms 16 ms 16 ms
 4 17.4.4.191 16 ms 16 ms 16 ms
```

Exit C M

Returns to the previous level prompt.

Syntax: exit

Example:

```
exit
```

14

Configuring and Monitoring SNMP

This chapter describes the SNMP configuration and console commands.

For more information about SNMP, refer to the *Routing Protocols Reference Guide.*

14.1 Accessing the SNMP Configuration Environment

SNMP has been implemented as a protocol. For information about accessing the SNMP protocol configuration and console environments, see Chapter 1.

14.2 SNMP Configuration and Console Commands

This section summarizes and explains the SNMP configuration and console commands.

Table 14–1 lists the SNMP configuration and console commands. Table 14–2 separately lists the configuring commands and their options.

The SNMP configuration commands allow you to specify network parameters for router interfaces that transmit SNMP packets. The information you specify takes effect immediately with the exception of the **set trap** command. Enter the SNMP configuration commands at the `SNMP config>` prompt.

The SNMP console commands enable you to view the parameters and statistics for the router interfaces. Enter the SNMP console commands from the `SNMP>` prompt.

14.2 SNMP Configuration and Console Commands

Table 14–1 SNMP Configuration and Console Commands Summary

Command	Tasks	Function
? (Help)	Configuring and Monitoring	Lists all the SNMP configuration or console commands or lists the options associated with specific commands.
Add	Configuring	Adds a community to the list of SNMP communities, an IP address with mask to a community, or a subtree to a MIB view.
Delete	Configuring	Removes a community from the list of SNMP communities, an IP address with mask from a community, or a subtree from a MIB view.
Enable/Disable	Configuring	Enables/disables SNMP protocol and standard traps associated with named communities.
List	Configuring and Monitoring	Displays the current communities with their associated access modes, enabled traps, IP addresses, and views. Also displays all views and their associated MIB subtrees.
Set	Configuring	Sets a community's access mode or view. A community's access modes is one of the following: • Read and trap generation • Read, write and trap generation • Trap generation only Also allows setting of trap UDP port.
Statistics	Monitoring	Displays statistics about the number of defined variables and the size of the MIB.
Exit	Configuring and Monitoring	Exits the SNMP configuration or console process and returns to the previous environment.

14.2 SNMP Configuration and Console Commands

Table 14–2 SNMP Configuration Commands Options Summary

Command	Param 1	Param 2	Param 3	Param 4	Default
add	community	<name>			None
	address	<comm_name>	<ipAddress>	<ipMask>	
	sub_tree	<view_text_name>	<oid>		
delete	community	<name>			
	address	<comm_name>	<ipAddress>		
	sub_tree	<view_text_name>	<oid>		
disable	snmp				
	trap	all	<comm_name>		
		cold_start	<comm_name>		
		warm_start	<comm_name>		
		link_down	<comm_name>		
		link_up	<comm_name>		
		auth_fail	<comm_name>		
		egp	<comm_name>		
		enterprise	<comm_name>		
enable	snmp				
	trap	all	<comm_name>		
		cold_start	<comm_name>		
		warm_start	<comm_name>		
		link_down	<comm_name>		
		link_up	<comm_name>		
		auth_fail	<comm_name>		
		egp	<comm_name>		

14.2 SNMP Configuration and Console Commands

Table 14–2 SNMP Configuration Commands Options Summary (Continued)

Command	Param 1	Param 2	Param 3	Param 4	Default
		enterprise	<comm_name>		
list	all				
	community	access			
		traps			
		address			
		view			
	views				
set	community	access	read_trap	<comm_name>	
			write_read_trap	<comm_name>	
			trap_only	<comm_name>	
		view	<comm_name>	all	
				<view>	
	trap_port	<udpPort#>			
exit					

14.2 SNMP Configuration and Console Commands

? (Help) C M

Lists the commands that are available from the current prompt level. You can also enter **?** after a specific command name to list its options.

Syntax: ?

Example:

```
?
ADD
DELETE
DISABLE
ENABLE
LIST
EXIT
```

Add C

Adds a community name to the list of SNMP communities, adds an address to a community, or assigns a portion of the MIB (subtree) to a view.

Syntax: add

```
address
community
sub_tree
```

address ***community-name ip-address ip-mask***

Note: SNMP requests may arrive for any of the routers' addresses.

You may specify one or more address for a community. You must enter the command each time you want to add another address.

If you specify no addresses for a community, requests are handled from any host. The addresses also specify those hosts that receive the traps. If no addresses are specified, no traps are generated.

Example:

```
add address
Community Name [trap]?
New Address [0.0.0.0]?
```

14.2 SNMP Configuration and Console Commands

community *community name*

Use the **add community** command to create a community with read_trap access, a view of all, allows all IP addresses access, and all traps disabled.

Note: The **add community** command no longer allows you to select access type or trap control. Use the **set community access** command to assign access types to existing SNMP communities.

Example:

```
add community
Community Name  []?
```

Community Name Specifies the name of community (32 characters maximum). Characters such as spaces, tabs, or <esc> key sequences are not accepted.

sub_tree *view-name MIB-OID*

Use the **add sub_tree** command to add a portion of the MIB to a view or to create a new view. The **add sub_tree** command is used to manage MIB views. More than one subtree can be added to a view defined by *view-name*. To create a new MIB view, issue the **add sub_tree** command with the new view name.

Note: You must assign a view to one or more communities using the **set community view** command to have it take effect.

Example:

```
add sub_tree
  View Name  [system-only]?
  MIB OID name [1.3.6.1.2.1.1]?
```

View Name Specifies the name of the view (32 visual characters maximum). Characters such as spaces, tabs, or <esc> key sequences are not accepted.

MIB OID Specifies the MIB Object ID for the sub_tree. This must be entered as a numeric value, not a symbolic value.

14.2 SNMP Configuration and Console Commands

Delete C

Deletes an address from a community, acommunity and all of its addresses, or a subtree from a view.

Syntax: delete
address
community
sub_tree

address ***community-name ip-address***

Removes an address from a community. You must supply the name.

Example:

```
delete address community-name ip-address
```

community ***community-name***

Removes a community and its IP addresses. You must supply the community name.

Example:

```
delete community community-name
```

sub_tree ***object-identifier***

Removes a MIB or a portion of the MIB from a view. You must supply the name of the subtree. If all subtrees are deleted, the MIB view is also deleted and all references to it from any associated SNMP communities are removed.

Example:

```
delete sub_tree object-identifier
```

Disable C

Disables the **snmp** protocol or specified traps on the router.

Syntax: disable
snmp
trap

snmp

Disables SNMP.

Example:

```
disable snmp
```

14.2 SNMP Configuration and Console Commands

trap ***trap-type community-name***

Disables specified traps or all traps. You must specify the trap type from the options shown in Table 14–3.

Example:

```
disable trap   trap-type  community-name
```

Table 14–3 Disable SNMP Trap types

Trap Type	Description
all	Disables all traps in a specified community.
cold_start	Disables cold start traps in a specified community. A cold start trap (0) means that the transmitting router is reinitializing and that the agent's configuration or the protocol entity implementation may be altered.
warm_start	Disables warm start traps in a specified community. A warm start trap (1) means that the transmitting router is reinitializing, but the configuration or protocol implementation remains the same.
link_down	Disables link_down traps in a specified community. A link_down trap (2) recognizes a failure in one of the communication links represented in the agent's configuration. The link_down trap-PDU contains the name and value of the ifIndex instance for the affected link as the first element of its variable-bindings.
link_up	Disables link_up traps in a specified community. A link_up trap recognizes that a previously inactive link in the network has come up. The link_up trap-PDU contains the name and value of the ifIndex instance for the affected link as the first element of its variable-bindings.
auth_fail	Disables authentication failure traps for a specified community. Authentication failure traps recognize that the sending entity is the addressee of a protocol message that is not properly authenticated.
egp	Disables egp neighbor loss traps in a specified community. EGP Neighbor Loss traps recognize that an EGP neighbor and peer are marked down and no longer a peer. The egpNeighborLoss trap-PDU contains the name and value of the egpNeighAddr instance for the affected neighbor as the first element of its variable-bindings.

Table 14–3 Disable SNMP Trap types (Continued)

Trap Type	Description
enterprise	Disables enterprise specific traps in a specified community. Enterprise specific traps recognize that some enterprise specific event has occurred. The specific-trap field identifies the particular trap that occurred.

Enable C

Enables the **snmp** protocol or specified traps on the router.

Syntax: enable

```
snmp
trap
```

snmp

Enables SNMP.

Example:

```
enable snmp
```

trap ***trap-type community-name***

Enables specified traps or all traps. You must specify the trap type from the options shown in Table 14–4.

Example:

```
enable trap   trap-type  community-name
```

Table 14–4 Enable SNMP Trap types

Trap Type	Description
all	Enables all traps in a specified community.
cold_start	Enables cold start traps in a specified community. A cold start trap (0) means that the transmitting router is reinitializing and that the agent's configuration or the protocol entity implementation may be altered.
warm_start	Enables warm start traps in a specified community. A warm start trap (1) means that the transmitting router is reinitializing, but the configuration or protocol implementation remains the same.

Table 14–4 Enable SNMP Trap types (Continued)

Trap Type	Description
link_down	Enables link_down traps in a specified community. A link_down trap (2) recognizes a failure in one of the communication links represented in the agent's configuration. The link_down trap-PDU contains the name and value of the ifIndex instance for the affected link as the first element of its variable-bindings.
link_up	Enables link_up traps in a specified community. A link_up trap recognizes that a previously inactive link in the network has come up. The link_up trap-PDU contains the name and value of the ifIndex instance for the affected link as the first element of its variable-bindings.
auth_fail	Enables authentication failure traps for a specified community. Authentication failure traps recognize that the sending entity is the addressee of a protocol message that is not properly authenticated.
egp	Enables egp neighbor loss traps in a specified community. EGP Neighbor Loss traps recognize that an EGP neighbor and peer are marked down and no longer a peer. The egpNeighborLoss trap-PDU contains the name and value of the egpNeighAddr instance for the affected neighbor as the first element of its variable-bindings.
enterprise	Enables enterprise specific traps in a specified community. Enterprise specific traps recognize that some enterprise specific event has occurred. The specific-trap field identifies the particular trap that occurred.

14.2 SNMP Configuration and Console Commands

List C M

Displays the current configuration of SNMP communities, access modes, traps, network addresses, and views.

Syntax: list

all
community
views

all

Displays the current configuration of SNMP communities for Access, Traps, Address, and View. See the description for the **list community** command on the next page for details on the options.

Example:

```
list all
      Community Name                       Access
--------------------------       --------------------
public                           Read, Trap
johnp                            Read, Write, Trap
trap                             Trap Only
snmp                             Read, Trap

      Community Name             IP Address       IP Mask
--------------------------       ------------     ---------
public                           All             N/A
jonhp                            16.24.10.98     255.255.255.0
trap                             16.24.10.98     255.255.255.0
snmp                             All             N/A

      Community Name                    Enabled Traps
--------------------------       ----------------------------
public                           None
johnp                            None
trap                             Cold Restart, Warm Restart,
                                 Link Down, Link Up,
                                 Authentication Failure,
                                 EGP Neighbor Loss, Enterprise Specific
snmp                             None

      Community Name                  View
--------------------------       ------------------
public                           All
johnp                            All
trap                             All
snmp                             mibii-snmp
```

14.2 SNMP Configuration and Console Commands

```
         View Name                      Sub-Tree
--------------------------       -----------------------
mibii-snmp                       1.3.6.1.2.1.11
```

community *option*

Displays the specified options for all communities. Options are access, traps, address, view.

Example:

```
list community option
```

Access Displays the access modes for all communities.

Address Displays the network address for all communities.

Traps Displays the types of traps generated for all communities.

View Displays the MIB view for all communities.

community access

Example:

```
list community access
      Community Name                      Access
--------------------------       --------------------
public                           Read, Trap
johnp                            Read, Write, Trap
trap                             Trap Only
snmp                             Read, Trap
```

community address

Example:

```
list community address
      Community Name             IP Address        IP Mask
--------------------------       ------------     ---------
public                           All             N/A
jonhp                            16.24.10.98     255.255.255.0
trap                             16.24.10.98     255.255.255.0
snmp                             All             N/A
```

14.2 SNMP Configuration and Console Commands

community traps

Example:

```
list community traps
      Community Name                          Enabled Traps
--------------------------       ----------------------------
public                           None
johnp                            None
trap                             Cold Restart, Warm Restart,
                                 Link Down, Link Up,
                                 Authentication Failure,
                                 EGP Neighbor Loss, Enterprise Specific
snmp                             None
```

community view

Example:

```
list community view
      Community Name                         View
--------------------------       ------------------
public                           All
johnp                            All
trap                             All
snmp                             mibii-snmp
```

views

Displays the current views for a specified SNMP community.

Example:

```
list views
         View Name                           Sub-Tree
--------------------------       -----------------------
mibii-snmp                       1.3.6.1.2.1.11
```

14.2 SNMP Configuration and Console Commands

Set C

Assigns a MIB view to a community or to set the SNMP UDP port numbers.

Syntax: set

community access . . .
community view . . .
trap_port . . .

community access *options community-name*

Use the **set community access** command to assign one of three access types to a community. You must supply the name of the community and the network address (in the standard *a.b.c.d.* notation).

Example:

```
set community access options  community_name
```

read_trap Sets read access and trap generation to the named community.

write_read_trap Sets write and read trap access to the community specified.

trap_only Sets the trap port to the named community.

community view *community_name options*

Use the **set community view** command to assign a MIB view to a community.

Example:

```
set community view community_name  options
```

all Assigns all supported MIB views to the named community. All is the default.

view Assigns a specified MIB view to the named community.

trap_port *udp-port#*

Use the **set trap_port** command to specify a UDP port number to send traps to the trap port. The default is the standard port.

Example:

```
set trap_port udp-port#
```

UDP Port Number Specifies a User Datagram Protocol port other than the standard UDP port (default # 162).

Statistics M

Displays the statistics about the number of defined variables and the size of the MIB. The statistics can change only when the load or hardware configuration changes.

Syntax: statistics

Example:

```
statistics
Number of defined variables = 231
Size of MIB = 14320 bytes
```

Exit C M

Returns to the previous level prompt.

Syntax: exit

Example:

```
exit
```

15

Configuring and Monitoring RMON

This chapter describes how to configure and monitor the Remote Monitoring feature.

15.1 About the RMON Feature

The RMON feature allows you to configure the router so that it independently monitors its own MIB variables and network traffic.   The RMON feature supports the Alarm and Event MIB groups and adheres to RMON MIB RFC-1757 for Ethernet objects.

15.1.1 RMON Alarm and Event Groups

The alarm group allows you to configure the router so that it monitors its own MIB variables. If the value of a monitored variable crosses its configured thresholds, the RMON feature generates an event. The event group associates an event with a set of actions. Two actions are defined: generate an SNMP trap message and add an entry to the event group log table.

You can configure alarms and events from a network management application, such as a MIB browser, that uses SNMP. You can also use the GWCON or Config processes to read and write MIB variables in the alarm and event groups.

You can separately configure the Event Logging System (ELS) to generate an ELS event when an RMON alarm occurs. Otherwise, the RMON event group and ELS are independent of each other.

The alarm and event are stored in NVRAM (nonvolatile RAM) memory and are preserved if you power cycle the router. You can delete individual table entries by using the `RMON Config>` **delete** command or you can use SNMP to set the table entry status to invalid. You can also use the `Config>` **clear rmon** command to delete all RMON table entries.

15.1 About the RMON Feature

If you use SNMP to create, delete, or modify alarm and event table rows, you must follow the conventions for EntryStatus as specified in the RMON MIB (rfc 1757). You are not required to follow the EntryStatus conventions when you configure alarm and event table rows from the RMON configuration process. It correctly transitions row status.

The number of alarm and event table entries is limited only by the amount of available memory in your router. Each entry requires approximately 500 bytes of heap memory.

You cannot write more than one alarm table or event table row at a time in a single SNMP set PDU. If you do not specify all the values for a row in a set PDU, the default values specified in Table 15–1 and Table 15–2 are used. The RMON configuration process uses these default values only the first time you enter an alarm or event from the command prompt. Then it uses the values you last entered as a default.

Table 15–1 RMON Alarm Table Default Values

Alarm Variable	Default Value
alarmIndex	1
alarmStatus	valid
alarmInterval	1
alarmVariable	0.0
alarmSampleType	deltaValue
alarmValue	0
alarmStartupAlarm	risingOrFallingAlarm
alarmRisingThreshold	0
alarmFallingThreshold	0
alarmRisingEventIndex	0
alarmFallingEventIndex	0
alarmOwner	Null string
alarmDescription	Null string

Table 15–2 RMON Event Table Default Value

Event Variable	Default Value
eventIndex	1
eventStatus	valid
eventDescription	Null string
eventType	log-and-trap
eventLastTimeSent	0:00:00.00
eventCommunity	Null string
eventOwner	Null string

15.2 RMON Basic Configuration Example

The following sections provide a configuration example. In this example, whenever the number of received SNMP packets increase, the router generates a *risingAlarm* trap to the community *rmon-trap t*hat has an IP address of *16.20.48.46*.

15.2.1 Configuring an SNMP Community

The following procedure describes how to configure SNMP with a new community named *rmon-trap* with an IP address of *16.20.48.46:*

1. At the `Config>` prompt, enter **protocol snmp** to display the `SNMP Config>` prompt. For example:

```
Config>protocol snmp
SNMP user configuration
SNMP Config>
```

2. Create an SNMP community and return to the `Config>` prompt. For example:

```
SNMP Config>add community rmon-trap
SNMP Config>
```

3. Configure the SNMP community. For example:

```
SNMP Config>add address rmon-trap
IP Address [0.0.0.0]? 16.20.48.46
IP Mask [255.255.255.255]?
SNMP Config>exit
Config>
```

The SNMP community *rmon-trap* is now configured.

15.2.2 Configuring an RMON Trap

The following sections describe how to configure RMON so that it generates a rising alarm trap if the MIB variable *snmpInPkts.0* (oid 1.3.6.1.2.1.11.1.0) increases by more than zero over a one second interval. This is accomplished by creating an event entry and an alarm entry.

15.2.2.1 Creating an Event Entry

To create an event entry with an event type *snmp-trap* and an event community *rmon-trap*, perform the following steps:

1. At the `Config>` prompt, enter **feature rmon** to display the `RMON Config>` prompt. For example:

```
Config>feature rmon
RMON user configuration
RMON Config>
```

2. Create an event entry. For example:

```
RMON Config>add event
Event Description []?
Event Type [log-and-trap]? snmp-trap
Event Community []? rmon-trap
Event Owner []?
Creating Event with index 4
RMON Config>
```

The RMON event entry is now configured, with an index value of 4.

15.2.2.2 Creating an Alarm Entry

To create an alarm entry for the above event entry, perform the following steps:

1. At the `RMON Config>` prompt, enter **add alarm** to create a new alarm entry. For example, this command adds a risingAlarm and associates it with the event defined previously (event index 4):

```
RMON Config>add alarm
Alarm Interval [1]?
Alarm Variable []? 1.3.6.1.2.1.11.1.0
Alarm Sample Type [deltaValue]?
Startup Alarm [risingOrFallingAlarm]? risingAlarm
Rising Alarm Threshold [0]? 1
Falling Alarm Threshold [0]?
Rising Event Index [0]? 4
Falling Event Index [0]?
Alarm Owner []?
Alarm Description []?
Creating Alarm with index 1
RMON Config>
```

The RMON alarm entry is now configured, with an index value of 1.

15.2.3 Summary

In the RMON example, whenever the router receives an SNMP request (snmpInPkts.0 increases by 1 or more), it sends an RMON risingAlarm trap.

15.3 Accessing the RMON Configuration Environment

You configure remote monitoring using the RMON feature.

To access the RMON feature configuration environment, use the **feature rmon** command at the `Config>` prompt.

Example:

```
Config>feature rmon
RMON user configuration
RMON Config>
```

Note: After you access the RMON feature configuration process, you may begin entering configuration commands. Whenever you make changes to a user-configurable feature parameter, the changes take effect *immediately*. You do not have to restart the router.

15.4 RMON Configuration and Console Commands

To access the RMON feature console prompt, enter **feature rmon** at the GWCON (+) prompt.

Example:

```
+ feature rmon
RMON User console
RMON Console>
```

Refer to Chapter 1 for information about accessing the configuration and console processes.

You can also configure RMON alarms and events from a network management application, such as a MIB browser, that uses SNMP.

15.4 RMON Configuration and Console Commands

This section summarizes and explains the RMON configuration and console commands.

The RMON configuration commands allow you to configure alarms and events. Enter the RMON configuration commands at the `RMON Config>` prompt.

The RMON console commands enable you to view the alarms, events and log table for the router. Enter the RMON console commands from the `RMON Console>` prompt.

Table 15–3 lists the RMON configuration and console commands.

Table 15–3 RMON Configuration and Console Commands Summary

Command	Tasks	Function
? (Help)	Configuring and Monitoring	Lists all the RMON configuration or console commands or lists the options associated with specific commands.
Add	Configuring	Adds an alarm or event.
Delete	Configuring and Monitoring	Removes an alarm or event or flushes the log table .
List	Configuring and Monitoring	Displays the current alarms and events and the maximum log table size.
Set	Configuring	Modifies an existing alarm or event, or defines the maximum size of the log table.
Exit	Configuring and Monitoring	Exits the RMON configuration or console process.

15.4 RMON Configuration and Console Commands

? (Help) C M

Lists the commands that are available from the current prompt level. You can also enter **?** after a specific command name to list its options.

Syntax: ?

Example:

```
?
ADD
SET
DELETE
LIST
EXIT
```

Add C

Adds an alarm definition or an event to monitor.

Syntax: add
alarm
event

alarm

RMON periodically takes statistical samples from variables in the probe and compares them to previously configured alarm thresholds. If the monitored variable crosses a threshold, an event is generated.

Example:

```
add alarm
Alarm Interval [1]?
Alarm Variable []? 1.3.6.1.2.1.11.1.0
Alarm Sample Type [deltaValue]?
Startup Alarm [risingOrFallingAlarm]? risingAlarm
Rising Alarm Threshold [0]? 1
Falling Alarm Threshold [0]?
Rising Event Index [0]? 4
Falling Event Index [0]?
Alarm Owner []?
Alarm Description []?
Creating Alarm with index 1
```

Alarm Interval

The interval in seconds over which the data is sampled and compared with the rising and falling thresholds.

Alarm Variable

The object identifier (OID) of the particular variable to be sampled. Only variables that resolve to an ASN.1 primitive type of INTEGER (INTEGER, Counter, Gauge, or TimeTicks) may be sampled.

If at any time the variable name of an established alarm entry is no longer available in the selected MIB view, the status of this alarm entry is set to **invalid**.

Alarm Sample Type

The method of sampling the selected variable and calculating the value to be compared against the thresholds. If the value of this object is **absoluteValue**, the value of the selected variable will be compared directly with the thresholds at the end of the sampling interval. If the value of this object is **deltaValue**, the value of the selected variable at the last sample will be subtracted from the current value, and the difference compared with the thresholds.

The default is **deltaValue**.

Startup Alarm

The alarm that may be sent when this entry is first set to valid. If the first sample after this entry becomes valid is greater than or equal to the *Rising Alarm Threshold* and *Startup Alarm* is **risingAlarm** or **risingOrFallingAlarm**, then a single rising alarm will be generated.

If the first sample after this entry becomes valid is less than or equal to the *Falling Alarm Threshold* and *Startup Alarm* is equal to **fallingAlarm** or **risingOrFallingAlarm**, then a single falling alarm will be generated.

Values are **risingAlarm**, **fallingAlarm** or **risingOrFallingAlarm** (default).

Rising Alarm Threshold

When the current sampled value is greater than or equal to this threshold, and the value at the last sampling interval was less than this threshold, a single event will be generated. A single event will also be generated if the first sample after this entry becomes valid is greater than or equal to this threshold and the associated *Startup Alarm* is either **risingAlarm** or **risingOrFallingAlarm**.

After a rising event is generated, another such event will not be generated until the sampled value falls below this threshold and reaches the *Falling Alarm Threshold*.

Falling Alarm Threshold

When the current sampled value is less than or equal to this threshold, and the value at the last sampling interval was greater than this threshold, a single event will be generated. A single event will also be generated if the first sample after this entry becomes valid is less than or equal to this threshold and the associated *Startup Alarm* is either **fallingAlarm** or **risingOrFallingAlarm**.

After a falling event is generated, another such event will not be generated until the sampled value rises above this threshold and reaches the *Rising Alarm Threshold*.

Rising Event Index

The index of the event entry that is used when a rising threshold is crossed.

The event entry identified by a particular value of this index is the same as identified by the same value of the event index object. If there is no corresponding entry event, then no association exists. In particular, if this value is zero, no associated event will be generated, as zero is not a valid event index.

Falling Event Index

The index of the event entry that is used when a falling threshold is crossed.

The event entry identified by a particular value of this index is the same as identified by the same value of the event index object. If there is no corresponding entry event, then no association exists. In particular, if this value is zero, no associated event will be generated, as zero is not a valid event index.

Alarm Owner

The entity that configured this entry and is therefore using the resources assigned to it.

Alarm Description A comment describing this alarm entry.

Creating Alarm with Index Index number assigned to this alarm entry.

event

Use the **add event** command to create an event to monitor. The configured events control the generation and notification of events from this device.

Example:

```
add event
Event Description []?
Event Type [log-and-trap]? snmp-trap
Event Community []? rmon-trap
Event Owner []?
Creating Event with index 4
```

Event Description A comment describing this event entry.

Event Type The type of notification that will be made about this event. Values are:

- **none** – No notification.
- **log**– Make an entry in the log table for each event.
- **snmp-trap** – Send an SNMP trap to one or more management stations
- **log-and-trap –** Make an entry in the log table and send an SNMP trap for each event. This is the default value.

Event Community The name of the SNMP community associated with this event.

Event Owner The entity that configured this entry and is therefore using the resources assigned to it.

Creating Event with index An index that uniquely identifies an entry in the event table. Each such entry defines one event that is to be generated when the appropriate conditions occur.

15.4 RMON Configuration and Console Commands

Delete C

Deletes a specified alarm or event, or flushes the log table.

Syntax: delete

alarm . . .
event . . .
log

alarm

Removes an alarm. You must supply the index number of the alarm to be deleted.

Example:

```
delete alarm
Alarm Index [0]?4
```

event

Removes an event. You must supply the index number of the event to be deleted.

Example:

```
delete event
Event Index [0]?1
```

log

Deletes all entries in the log table.

Example:

```
delete log
Do you want to flush the log table? (Yes or [No]):
```

Delete M

Deletes all entries in the log table.

Syntax: delete

log

Example:

```
delete log
Do you want to flush the log table? (Yes or [No]):
```

15.4 RMON Configuration and Console Commands

List C M

Displays the current configuration of RMON alarms, events, the maximum size of the log table and the entries in the log table.

Syntax: list

all
alarm
event
log

all

Displays the current configuration of RMON alarms, events, the maximum size of the log table and the entries in the log table.

Example:

```
list all
Alarm Index: 1
Alarm Status: valid
Alarm Interval: 1
Alarm Variable: 1.3.6.1.2.1.11.1.0
Alarm Sample Type: deltaValue
Alarm Value: 0
Startup Alarm: risingAlarm
Rising Alarm Threshold: 1
Falling Alarm Threshold:  0
Rising Event Index: 4
Falling Event Index: 0
Alarm Owner:
Alarm Description:
Rising Event Count: 0
Falling Event Count: 0

Event Index: 4
Event Status: valid
Event Description:
Event Type: snmp-trap
Event Last Time Sent: 0d  0:00:00.00
Event Community: rmon-trap
Event Owner:

Log Table Maximum Size: 30
Log Index: 1
Log Event Index: 1
Log Time: 0d  0:00:03.68
Log Description:
```

Alarm Index

Index number used to identify this alarm entry.

Alarm Status

Status of this alarm entry. Values are **valid**, **createRequest**, **underCreation** or **invalid**.

Alarm Interval

The interval in seconds over which the data is sampled and compared with the rising and falling thresholds.

Alarm Variable

The object identifier (OID) of the particular variable to be sampled.

Alarm Sample Type

The method of sampling the selected variable and calculating the value to be compared against the thresholds. The value is either **absoluteValue**, or **deltaValue**.

The default is **deltaValue**.

Alarm Value

The value of the statistic during the last sampling period. For example, if the sample type is **deltaValue**, this value will be the difference between the samples at the beginning and end of the period. If the sample type is **absoluteValue**, this value will be the sampled value at the end of the period.

This is the value that is compared with the *rising* and *falling* thresholds.

The value during the current sampling period is not made available until the period is completed and will remain available until the next period completes.

Startup Alarm

The alarm that may be sent when this entry is first set to valid. Values are **risingAlarm**, **fallingAlarm** or **risingOrFallingAlarm** (default).

Rising Alarm Threshold

The rising threshold. A single event will be generated the first time the sampled value is greater than or equal to this threshold if the associated *Startup Alarm* is either **risingAlarm** or **risingOrFallingAlarm**.

After a rising event is generated, another such event will not be generated until the sampled value falls below this threshold and reaches the *Falling Alarm Threshold*.

Falling Alarm Threshold The falling threshold. A single event will be generated the first time the sampled value is less than or equal to this threshold if the associated *Startup Alarm* is either **fallingAlarm** or **risingOrFallingAlarm**.

After a falling event is generated, another such event will not be generated until the sampled value rises above this threshold and reaches the *Rising Alarm Threshold*.

Rising Event Index The index of the event entry that is used when a rising threshold is crossed.

Falling Event Index The index of the event entry that is used when a falling threshold is crossed.

Alarm Owner The entity that configured this entry and is therefore using the resources assigned to it.

Alarm Description A comment describing this alarm entry.

Rising Event Count The number of rising events that have been reported.

Falling Event Count The number of falling events that have been reported.

Event Index Index number used to identify this event entry.

Event Status Status of this event entry. Values are **valid**, **createRequest**, **underCreation** or **invalid**.

Event Description A comment describing this event entry.

Event Type The type of notification that will be made about this event. Values are:

none -- No notification.

log -- Make an entry in the log table for each event.

snmp-trap -- Send an SNMP trap to one or more management stations

log-and-trap -- Make an entry in the log table and send an SNMP trap for each event. This is the default value.

Event Last Time Sent Time that this event was last reported.

Event Community The name of the SNMP community associated with this event.

Event Owner The entity that configured this entry and is therefore using the resources assigned to it.

Log Table Maximum Size The maximum number of reported events that can be held in the log table, if the *event type* is **log** or **log-and-trap**. When the table is full the oldest entry will be deleted to allow a new event to be reported.

Log Index Index number of the log table entry.

Log Event Index Index number of the event that was reported.

Log Time The time that the event was reported.

Log Description A comment describing this log entry, taken from the *Event Description* field of the reported event.

alarm *index-number*

Displays the specified options for all alarms, or for the alarm identified by the index number.

Example:

```
list alarm 1
Alarm Index: 1
Alarm Status: valid
Alarm Interval: 1
Alarm Variable: 1.3.6.1.2.1.11.1.0
Alarm Sample Type: deltaValue
Alarm Value: 0
Startup Alarm: risingAlarm
Rising Alarm Threshold: 1
Falling Alarm Threshold:  0
Rising Event Index: 4
Falling Event Index: 0
Alarm Owner:
Alarm Description:
Rising Event Count: 0
Falling Event Count: 0
```

15.4 RMON Configuration and Console Commands

event *index-number*

Displays the specified options for all events, or for the event identified by the index number.

Example:

```
list event 4
Event Index: 4
Event Status: valid
Event Description:
Event Type: snmp-trap
Event Last Time Sent: 0d  0:00:00.00
Event Community: rmon-trap
Event Owner:
```

log

Displays the maximum number of events that can be reported in the log table.

Example:

```
list log
Log Table Maximum Size: 30

Log Index: 1
Log Event Index: 1
Log Time: 0d  0:00:03.68
Log Description:

Log Index: 2
Log Event Index: 1
Log Time: 0d  0:01:10.68
Log Description:

Log Index: 3
Log Event Index: 1
Log Time: 0d  0:01:18.18
Log Description:
```

15.4 RMON Configuration and Console Commands

Set C

Modifies an alarm or event that was previously created using the **add** command or sets the maximum number of entries that can be held in the log table.

Syntax: set

alarm
event
log-table-max

alarm

Use the **set alarm** command to modify one or more options of an alarm added using the **add alarm** command. This command prompts for each option in turn, displaying current value as a default.

Example:

```
set alarm
Alarm Index [1]?
Alarm Status [valid]?
Alarm Interval [1]?
Alarm Variable [1.3.6.1.2.1.11.1.0]?
Alarm Sample Type [deltaValue]?
Startup Alarm [risingAlarm]?
Rising Alarm Threshold [1]?
Falling Alarm Threshold [0]?
Rising Event Index [4]?
Falling Event Index [0]?
Alarm Owner []?
Alarm Description []?
Modifying Alarm with index 1
```

event

Use the **set event** command to modify one or more options of an event added using the **add event** command. This command prompts for each option in turn, displaying current value as a default.

Example:

```
set event
Event Index [4]?
Event Status [valid]?
Event Description []?
Event Type [snmp-trap]?
Event Community [rmon-trap]?
Event Owner []?
Modifying Event with index 4
```

15.4 RMON Configuration and Console Commands

log-table-max

Use the **set log-table-max** command to specify the maximum number of event reports that can be retained in the log table. When the log table is full, the oldest entry will be deleted when a new event is reported.

The default maximum size for the log table is 30.

Example:

```
set log-table-max
Log Table Maximum Size: [30]?
```

Exit C M

Returns to the previous level prompt.

Syntax: exit

Example:

```
exit
```

16

Using the DIGITAL Trace Facility

This chapter gives an overview of the DIGITAL Trace Facility (DTF) and describes what information can be traced over the interfaces in the router that has tracepoints.

For more information on DTF, refer to the *DTF (DIGITAL Trace Facility) User's Guide*.

16.1 Overview of the DIGITAL Trace Facility Utility

The DIGITAL (or DECnis) Trace Facility is a utility which traces packets as they traverse through the protocol layers within a router. It is an enhancement to the Common Trace Facility (CTF) provided in DECnet/OSI, supporting multiple host platforms over both DECnet and TCP/IP networks. DTF can only access the DIGITAL RouteAbout router using TCP/IP.

The points within a router which can be traced are known as Tracepoints. Each protocol module within the router may have 0 or more tracepoints, for example the PPP data link module has a tracepoint for each interface.

Each tracepoint defines a number of Events. Each packet which is traced through the tracepoint is marked with one of these events by the router, for example Tx for a packet being transmitted or Rx for a packet being received.

When you run DTF on a host system it uses a transport protocol (in this case TCP/IP) to connect to the router you want to trace and sends the parameters to use for the trace session. Events at the activated tracepoints are transmitted back to the host for analysis (either live display, or recorded in a trace file for later analysis).

16.2 Accessing DTF

The DTF software is included with the clearVISN Router Configurator software. It is in the ***install-directory*\tools\supported\dtf\** subdirectory.

The latest versions of the DTF documentation and installation kits for each host platform are available over the Internet, and can be downloaded from the following World Wide Web locations:

- **http://www.networks.digital.com** (U.S.)
- **http://www.networks.europe.digital.com** (Europe)
- **http://www.networks.digital.com.au** (Asia Pacific)

Use the search feature to find the DTF Installation Kit.

16.3 Router Tracepoints

Table 16–1 summarizes the DTF tracepoints for the router. The sections that follow explain these tracepoints in more detail.

Table 16–1 Router Tracepoint Summary

Tracepoint	Displays
ppp interface *	All packets passing through the PPP datalink
osi interface *	All packets passing through OSI routing
decnet interface *	All packets passing through DECnet Phase IV routing. This tracepoint is only used when the router is configured for DECnet Phase IV Only.
ethernet interface *	Packets sent and received over the Ethernet interface. **Note:** The tracepoint may not see all packets on a busy Ethernet due to the high transfer rates.
ospf interface *	Packets received and transmitted by the OSPF protocol.
pim interface *	Packets received and transmitted by the PIM protocol
frbs interface *	Packets received and transmitted by a Frame Relay interface
lapb interface *	Packets transferred through the LAPB interface of the X.25 datalink
x25l3 dte *	Packets sent and received at level 3 of the X.25 datalink
igmp interface *	All packets received by the IGMP module, and IGMP packets transmitted by the IGMP module
els events *	ELS events which the router logs, providing a simple remote event logger facility.

16.3 Router Tracepoints

ppp interface *

Displays all packets passing through the PPP datalink. This includes all LCP and NCP startup packets.

To trace packets over a PPP-FR pseudo interface, you can use the PPP INTERFACE tracepoint specifying the PPP-FR pseudo interface number. DTF will trace the PPP packets. If you use the FRBS INTERFACE tracepoint and specify the frame relay interface that is being used, you will see the same data with the Frame Relay headers added, plus any normal frame relay traffic on that interface.

Events

Tx	Transmitted packets
Rx	Received packets

osi interface *

Displays all packets passing through OSI routing. Packets are traced just before they are passed to the datalink on transmit and just after they have been received.

Note: If the router is configured with both DECnet/OSI and DECnet Phase IV enabled, then all OSI and DECnet Phase IV traffic is traced through the OSI tracepoint.

Events

ISIS-Tx	Transmitted ISIS packets, for example ISIS Hellos, SNPs and LSPs
ISIS-Rx	Received ISIS packets, for example ISIS Hellos, SNPs and LSPs
ESIS-Tx	Transmitted ESIS packets such as ESIS Hellos
ESIS-Rx	Received ESIS packets such as ESIS Hellos
Tx	Transmitted forwarding data (includes Phase IV hellos and data)
Rx	Received forwarding data (includes Phase IV hellos and data)

16.3 Router Tracepoints

decnet interface *

This tracepoint is used only when the router is configured for DECnet Phase IV Only. If the router is configured to run DECnet Phase IV and DECnet/OSI, then all packets are traced through the OSI tracepoint.

Events

Tx	Transmitted packets
Rx	Received packets
ToOSItp	Indicates packets are being passed to the OSI tracepoint

ethernet interface *

This tracepoint traces packets as they are sent and received over the Ethernet interface. The tracepoint can generate large amounts of data when connected to a busy Ethernet, so there are a number of events to enable you to filter the Ethernet traffic that is traced.

Note: The tracepoint may not see all packets on a busy Ethernet due to the high transfer rates.

Events

Tx	Transmitted packets not included below
Rx	Received packets not included below
ipTx	Transmitted IP packets
ipRx	Received IP packets
arpTx	Transmitted ARP packets
arpRx	Received ARP packets
dnTx	Transmitted DECnet Phase IV packets
dnRx	Received DECnet Phase IV packets
xnsTx	Transmitted XNS packets
xnsRx	Received XNS packets

ipxTx	Transmitted IPX packets
ipxRx	Received IPX packets
aplTx	Transmitted AppleTalk packets
aplRx	Received AppleTalk packets
aarpTx	Transmitted AppleTalk ARP packets
arpRx	Received AppleTalk ARP packets
mopTx	Transmitted MOP packets
mopRx	Received MOP packets
osiTx	Transmitted OSI (DECnet Phase V) packets
osiRx	Received OSI (DECnet Phase V) packets

ospf interface *

Displays all the OSPF porting Control packets passing through the OSPF module.

Events

Tx	Transmitted packets
Rx	Received packets

pim interface *

Displays packets as they are received and transmitted by the PIM protocol. This includes both forwarded IP multicast packets and all PIM protocol packets, when PIM is used as the multicast routing protocol.

Events

Tx	Transmitted IP multicast packets
Rx	Received IP multicast packets
PimTx	Transmitted PIM protocol messages
PimRx	Received PIM protocol messages

16.3 Router Tracepoints

frbs interface *

The Frame Relay Bearer Services (frbs) tracepoint displays packets as received and as transmitted by a Frame Relay interface.

Events

Tx	Transmitted packets
Rx	Received packets

lapb interface *

Displays all packets passing through the frame layer an X.25 interface. This is equivalent to the OSI data link layer, supporting the LAP-B protocol.

Events

Tx	Transmitted frames
Rx	Received frames
HoldBack	Holdback timer expired
PollTime	Poll timer expired
RetryMax	Exceeded maximum retries
LineUp	Connection restored
LineDown	Connection lost

x25l3 dte *

Displays all packets passing through the packet layer of an X.25 interface. This is equivalent to the network layer of the OSI model.

Events

Tx	Transmitted packets
Rx	Received packets

igmp *

Displays all the Internet Group Membership Protocol (IGMP) packets arriving at the IGMP module, including PIM, DVMRP, and Mtrace protocol packets. Note that only IGMP Host Membership Queries and IGMP Host Membership Reports packets are traced on transmit.

Events

Tx	Transmitted IGMP host membership query and report packets
Rx	Received IGMP packets

els events ***‘subsystem;subsystem;subsystem’***

This tracepoint links into the Event Logging System (ELS) and allows ELS events to be displayed on the trace host or saved in a file. It provides a very simple remote event logger.

The ELS EVENTS tracepoint does not allow wildcards. You must specify the list of ELS subsystem names from which you wish to see events. This tracepoint can only be traced by one remote DTF user at a time. All other tracepoints can be traced by up to three remote DTF users at the same time.

16.3 Router Tracepoints

Example:

```
# dtf -l cvn1:”els events ppp;frl;isis;esis”
.
.
.
207809|Info    |Events  |  FRL.001: Frame received, PVC = 445
207849|Trace   |Events  | ESIS.008: sent hello 4900-01AA-0004-0095-0700
207849|Trace   |Events  | ESIS.008: sent hello 4900-01AA-0004-0095-0700
207849|Trace   |Events  | ISIS.032: Level 1 LAN hello sent on int 0
207856|Trace   |Events  |  PPP.017: LCP/Open rcv Echo Request, id 176
207856|Trace   |Events  |  PPP.016: LCP/Open snd Echo Reply, id 176 len
207856|Trace   |Events  |  PPP.017: LCP/Open rcv Echo Request, id 164
207856|Trace   |Events  |  PPP.016: LCP/Open snd Echo Reply, id 164 len
```

By default, DTF enables all events for the specified ELS subsystem, and displays them as they are received over the link from the router. You can filter the ELS events on the specified subsystems which are to be displayed by DTF by using these DTF event filters:

Events

Error	Only enables/displays ELS Error events
Info	Only enables/displays ELS Info events (noisy)
Trace	Only enables/displays ELS Trace events (noisy)
Always	Only enables/displays ELS Always events

Note: DTF enables only those events for display by the DTF user. ELS on the router’s console will display only those events that have been enabled using ELS commands on the console.

16.3 Router Tracepoints

16.3.1 Sample Output from DTF

The following output was obtained by running DTF on a DIGITAL UNIX host using the *ethernet interface ** tracepoint. The name of the router being traced is *cvn1*. The *login-name* and *password* fields are indicated in italics. The wildcard (*) indicates that all Ethernet interfaces on the router are to be traced. In this example, the output has been truncated.

Example:

```
# dtf -l cvn1/login-name/password:”ethernet interface *”
DTF for OSF Version V2.1-685, built on Feb  8 1996 13:02:55
Copyright Digital Equipment Corporation. 1995,1996, all rights reserved.

-DTF- tracing on node 16.36.16.122 using TCP
-DTF- tracepoint-0 is ”ethernet interface *”
-DTF- router type is Digital-RouteAbout (1)
-DTF- waiting for tracepoints to activate...
-DTF- 6 tracepoints being traced to stdout


     571035|0|osiRx   |Eth/0   |IEEE 802.3 01-80-C2-00-00-14
     571036|0|dnRx    |Eth/0   |EthernetII AB-00-00-03-00-00
     571036|0|Rx      |Eth/0   |IEEE 802.3 09-00-07-FF-FF-FF
     571037|0|ipRx    |Eth/0   |EthernetII 01-00-5E-00-00-02
     571040|0|osiRx   |Eth/0   |IEEE 802.3 01-80-C2-00-00-15
     571042|0|dnRx    |Eth/0   |EthernetII AB-00-00-03-00-00
```

DTF can also produce a verbose listing, which provides much more detail. The following extract from a verbose DTF listing is equivalent to the expansion of just one of the traced lines in the above example.

16.3 Router Tracepoints

Example:

```
     # dtf -lv cvn1/login-name/password:"ethernet interface *"
     .
     .
     .
-DTF- 0 ipTx       166 of 166  at    08:50:46L                      5
  Status: 00000000        Context: 00000000         Function: 00000000
  --------------------------------------------------------------------
                                       |
  NI length: EthernetII
  ---------------------
  Destination Address  FF-FF-FF-FF-FF-FF [Multicast to All stations]
  Source Address       AA-55-EE-01-37-06 | Protocol type      Internet Protocol
                                       |
  IP Packet: IP
  -------------
  Type of service      0x00              | Precedence          Routine
  Total length         152               | Packet identifier   0x4200
  Fragment offset      0x0000            | Time to live        60
  Verified checksum    0x1090            | Source Address      17.4.4.191
  Destination Address  17.4.4.255        | Protocol            UDP
                                       |


  IP protocol: UDP
  ----------------
  Verified UDP checksum 0x51BA           | UDP length           132
  Source port           router           | Destination port     router
                                       |
  UDP port: RIP
  -------------
  RIP version            1               | RIP Command           Response
                                       |
  RIP command: Response
  ---------------------
  RIP Networks...      6                 | network @ metric      17.5.5.0 @ 16
  network @ metric     17.0.1.0 @ 16     | network @ metric      17.9.9.0 @ 16
  network @ metric     17.3.3.0 @ 1      | network @ metric      17.2.2.0 @ 2
  network @ metric     17.1.1.0 @ 2      |
  .
  .
  .
```

If the router has been configured with remote console logins enabled, then DTF users must specify a valid login name and password before they are allowed to trace. If remote console login is disabled, then no login name and password are required.

Note: Passwords on the router are restricted to a maximum of 8 characters. DTF connections to the router using passwords which are greater than 8 characters long will fail.

16.3 Router Tracepoints

Refer to the *System Software Guide* for details about how to set up an administrative user login name and password for remote access to the router console.

Refer to the *DTF User Guide* for details of the command syntax on supported hosts and for explanations of the trace output.

A

SNMP Objects

This appendix summarizes the SNMP object supported by the router software. The base is MIB II (Management Information Base II) which is specified in RFC 1213. The groups from MIB II which have been implemented are the following:

- System Group
- Interfaces Group
  - Object ifInNUcastPkts is always 0, those packets are counted in object ifInUcastPkts
  - Object ifOutNUcastPkts is always 0, those packets are counted in object ifOutUcastPkts
- Address Translation Group - no objects are settable
- IP Group
  - Object ipDefaultTTL is settable
  - Routing table objects are not settable
- ICMP Group
- UDP Group
- EGP Group
- Transmission Group
- SNMP Group

The TCP Group B is not implemented. In addition, a proprietary MIB is implemented, with the following groups:

```
proteon               = { iso(1) org(3) dod(6) internet(1) private(4)
enterprise(1) 1 }

admin               = { proteon(1) 1 }
objid               = { admin(1) 1 }
status              = { admin(1) 2 }
els                 = { admin(1) 3 }

xface               = { proteon(1) 2 }
generic             = { xface(2) 1 }
tokenr              = { xface(2) 2 }
ieth                = { xface(2) 4 }
peth                = { xface(2) 5 }
comsl               = { xface(2) 6 }
gwsl                = { xface(2) 7 }
x25                 = { xface(2) 8 }
fddi                = { xface(2) 12}
ceth                = { xface(2) 13}
csl                 = { xface(2) 14}
seth                  = { xface(2) 15}
```

B

Packet Sizes

This appendix discusses the sizes of packets for the various networks and protocols that the p45xx supports. This appendix contains the following sections:

- General Issues
- Network-Specific Size Limits
- Protocol-Specific Size Limits
- Changing Maximum Packet Sizes

B.1 General Issues

For the purposes of this discussion, the packets that the routers handle consist of user data and header information.

The amount of user data within a packet is limited by the amount of header information on the packet. The amount of header information minimally depends on:

- The network-types over which the packet must travel
- The protocols in use on these networks

The following factors affect the size of the packet contents:

- Length of the Data-Link header information that the current network type and interface require the packet to have
- Length of the trailer information (if any) that the current network type and interface require the packet to have

On any given network, the sum of the maximum data size together with header and trailer sizes will equal the network's maximum packet size. When routing between networks of different maximum packet size, fragmentation of the packet may result.

B.2 Network-Specific Size Limits

Given the information in the previous section, the maximum amount of **Network** layer data supported by each **Data Link** layer (network interface) can be determined.

Table B–1 lists the packet sizes with defaults.

Table B–1 Network-Specific Packet Size Limits

Network Type (Data Link)	Network Layer max packet size (bytes)	Length of Network Header	Information Trailer
FDDI	4479	20	0
Ethernet	1500	18	4
Serial Line	2046*	2	0

* Default

The maximum packet size is the maximum amount of data the protocol forwarder can pass to the device.

Note: These numbers correspond to the MTUs in 4.2 BSD UNIX.

For an IP packet, this includes the IP header, the UDP or TCP header, and all data. For a DECnet packet, this includes the Routing header (long format data packet), the NSP header, and any data. For an XNS packet, this includes the IDP header, the SPP or PXP header, and any data.

The packet size in use is displayed when the router’s GWCON **memory** command is used. The “Pkt” size is the Network layer packet size. The Hdr (header) and Tlr (trailer) sizes depend on the networks and their network interfaces.

B.3 Protocol-Specific Size Limits

This section explains the protocol-specific size limits.

B.3.1 IP Packet Lengths

The IP protocol specifications do not require a host IP implementation to accept IP packets of more than 576 octets. Router IP implementations *must* accommodate IP packets of any length up to the limits imposed by the network-specific packets in use.

Furthermore, router IP performs transparent fragmentation and reassembly of packets that would otherwise exceed network-specific length restrictions, as required by the IP specification.

Packet size mismatches do not cause connectivity problems. However, fragment reassembly does pose a performance penalty, so fragmentation should be avoided whenever possible.

B.3.2 DECnet Packet Lengths

In the default configuration and according to the specifications, all DECnet routers must forward packets that are 576 bytes long. It is allowable to configure hosts to use larger packets, up to the limits imposed by Ethernet. Not all DIGITAL routers support packets larger than 576 bytes, especially those based on PDP-11s, such as the DECSA (DIGITAL Ethernet Router Server). The DECnet-VAX buffer sizes should not be increased from 576 bytes if there are any PDP-11 routers in the DECnet network.

The maximum possible packet size in DECnet is 1498 bytes because DECnet adds two bytes of length to the Ethernet Data Link. This provides information on the network length of all packets, even when they are shorter than the minimum Ethernet packet size of 60 bytes (excluding 4 byte CRC).

The DECnet forwarder keeps internal maximums on the packet sizes for each network (circuit) that are two bigger than the minimum of 1498 packet size for that network interface.

If a packet is too large to be forwarded in DECnet, it is dropped. An event 4.3 (oversized packet loss) is logged, and the executor Oversized Packet Loss counter is incremented.

B.4 Changing Maximum Packet Sizes

Normally, the router automatically sets the maximum Network layer packet size to the size of the largest possible packet on all the connected networks. It then adds any headers and trailers required by the networks to determine the internal buffer size, which is larger than the Network layer size.

Some networks allow you to configure maximum packet sizes. Configuring maximum packet sizes affects the size of buffers used on the router and this in turn affects the number of buffers available for a given memory size. Routers automatically determine what size buffer it is going to need. You can change the maximum Network layer packet size that the router handles by using the **set packet-size** command. Do not use this command unless specifically directed to by Customer Service.

C

Comparison of Protocols

This appendix compares some of the well-known protocols that your router supports. It is provided as a memory aid and is not meant as a reference.

C.1 Protocol Comparison Table

Table C–1 compares the protocols.

Table C–1 Comparison of Protocols

ISO OSI	TCP/IP	Xerox	DECnet	Other	IPX	OSI
7 Application	Application		CTERM			
6 Presentation	(Telnet, FTP		DAP			
5 Session	TFTP, SGMP)		Session			
4 Transport	Transport (TCP, UDP)	SPP PXP	NSP		PXP SPX	TP
3 Network	Internet (IP, RIP, EGP, ICMP)	XNS	Routing	ARP	IPX RIP SAP	CLNP ES-IS IS-IS
2 Data Link						
	Local Net	Ethernet	PPP			
1 Physical				HDLC		

C.2 Key to Protocols

Table C–2 is a key to the protocols.

Table C–2 Protocol Key

Protocol	Description
AP	Authentication Protocol. Used with the Simple Gateway Monitoring Protocol (SGMP) to validate requests for router statistics.
CLNP	Connection-Less Network Protocol.
EGP	External Gateway Protocol. An IP routing protocol.
ES-IS	End System-Immediate System protocol (ISO 9542). Used between an end system and an immediate system to provide configuration information and route redirection information.
FTP, TFTP	File Transfer Protocol; Trivial File Transfer Protocol.
ICMP	Internet Control Message Protocol. Used to send network level error and control messages between routers and hosts.
IP	Internet protocol. IP is a widely used standard transport protocol. IP is the routers' basic protocol. IP leaves some error checking to higher level (end-to-end) protocols.
IPX	Internet Exchange Packet Protocol.
IS–IS	Immediate System–Immediate System protocol (ISO 10589). Used by immediate systems to communicate with each other.
RIP	Routing Information Protocol (Routing protocols are used to determine network topology and data paths). RIP is the common IP routing protocol. XNS also has a routing protocol called RIP.
SGMP	Simple Gateway Monitoring Protocol. Used to obtain statistics in machine-readable form from the routers.
SNMP	Simple Network Monitoring Protocol. Used to obtain statistics in machine-readable form from the routers.
TCP	Transport Control Protocol. An end-to-end (host-to-host) protocol that is often used with IP. Useful for sending streams of data. Uses checksums, acknowledgements, and timeouts to ensure the correct delivery and sequence of data.

D

DIGITAL MIB Support

This appendix describes the MIBs or portions of MIBs contained in the DIGITAL-Router-SNMP-Agent.

D.1 Standard MIBs

DIGITAL supports the following standard MIBs shown in Table D–1. This table indicates whether this implementation has exceptions from the RFC for each MIB. The sections following this table expand on the exceptions for PPP since they are too complex to include here. There is also a discussion of the differences in the interface numbering schemes used by the Router software and SNMP.

Table D–1 Standard MIBs

MIB	Supported Components	RFC Number
AppleTalk	The following groups are supported: • ARP • Port • DDP • RTMP • ZIP • Echo	RFC 1243
Bridge	All except: • dot1dStaticTable • dot1dTpFdbTable	RFC 1286
Ethernet	All	RFC 1623
FDDI	All	RFC 1285

Table D–1 Standard MIBs (Continued)

MIB	Supported Components	RFC Number
Frame Relay	All	RFC 1315
IPX	All	Refer to Novell documentation
MIB-II	All except: • TCP group	RFC 1213
MIB-II Extensions	ifStack group	RFC 1573
OSPF	All	RFC 1253
PPP	See Section D.1.2	RFC 1471
RMON	The following groups are supported: • Event • Alarm	RFC 1757
RS-232 Serial Line	All	RFC 1317
Token Ring	All	RFC 1231

D.1.1 SNMP MIB-II Interface Numbering

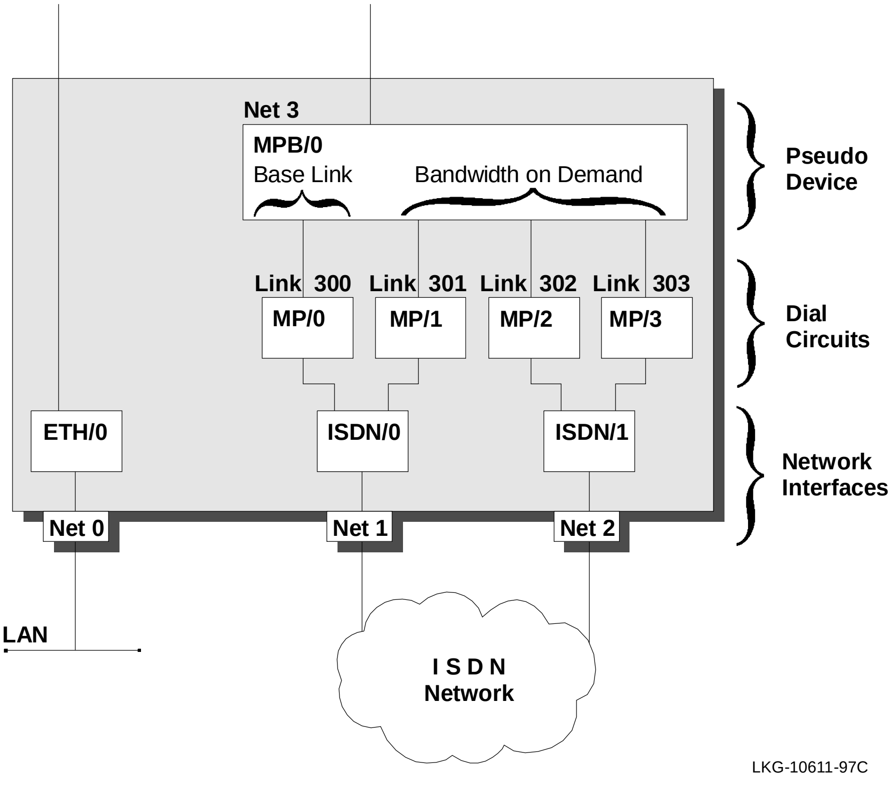
This implementation supports the **ifStack** group of RFC 1573 (Evolution of the Interfaces Group of MIB-II). Network managers should be aware that the interface numbers used by SNMP in MIB-II are allocated differently to the network interface numbers used throughout the router software.

Network interfaces on the router are referred to by the software with numbers starting from 0.

SNMP refers to interfaces (in the MIB-II interfaces table) with numbers starting from 1. For example, device Eth/0 on network interface 0, would be referred to in MIB-II as interface 1. It is usually sufficient to add 1 to the network interface number of a physical device on the router to get the SNMP interface number. It becomes more complicated when the router is configured with ISDN interfaces, pseudo devices, or dial circuits.

Consider the example illustrated in Figure D–1. The Net numbers are 0 for ETH/0, 1 for ISDN/0, and 2 for ISDN/1. When the Multilink bundle device MPB/0 is configured, it is allocated to Net 3. The Multilink link circuits do not have a Net number.

Figure D–1 Network Interface Numbers in Multilink PPP over ISDN



To determine the SNMP interface numbers allocated in this configuration, use SNMP to display ifDescr settings:

Example:

```
% walk ifDescr
ifDescr.1 = "eth0/net0 Ethernet/802.3"
ifDescr.2 = "isdn0/net1 ISDN Basic Rate"
ifDescr.3 = "isdn1/net2 ISDN Basic Rate"
ifDescr.4 = "isdn-d/isdn0 ISDN data channel"
ifDescr.5 = "isdn-b1/isdn0 ISDN bearer channel 1"
ifDescr.6 = "isdn-b2/isdn0 ISDN bearer channel 2"
ifDescr.7 = "isdn-d/isdn1 ISDN data channel"
ifDescr.8 = "isdn-b1/isdn1 ISDN bearer channel 1"
ifDescr.9 = "isdn-b2/isdn1 ISDN bearer channel 2"
ifDescr.10 = "ppp/isdn0 Point to Point Protocol"
ifDescr.11 = "ppp/isdn0 Point to Point Protocol"
ifDescr.12 = "ppp/isdn1 Point to Point Protocol"
ifDescr.13 = "ppp/isdn1 Point to Point Protocol"
ifDescr.14 = "ppp/MPB0 Point to Point Protocol"
```

You can see from the example listing that SNMP interface 1 relates to Network interface 0 (ETH/0). Table D–2 shows the relationship between the SNMP interface numbers and the Network Interface numbers from this listing.

Table D–2 SNMP Interfaces and Network Interface Numbers

SNMP Interface	Network Interface	Interface name
1	0	Eth/0
2	1	ISDN/0
3	2	ISDN/1
14	3	MPB/0

Notice that SNMP has allocated many other interface numbers. For each ISDN device, SNMP allocates an interface to the device, plus one for the D channel and one for each B channel. In this example, SNMP interface 4 is the D channel for ISDN/0 and interfaces 5 and 6 are the two B channels for ISDN/0. Similarly, interfaces 7, 8, and 9 are the D and B channels for ISDN/1. Interfaces 10, 11, 12, and 13 are the PPP dial circuits over the ISDN interfaces in the Multilink bundle. None of these components have an equivalent Network interface number on the router.

These assignments can also be confirmed using ifType:

Example:

```
% walk ifType
ifType.1 = "ethernet-csmacd"
ifType.2 = basicISDN
ifType.3 = basicISDN
ifType.4 = lapd
ifType.5 = ds0
ifType.6 = ds0
ifType.7 = lapd
ifType.8 = ds0
ifType.9 = ds0
ifType.10 = ppp
ifType.11 = ppp
ifType.12 = ppp
ifType.13 = ppp
ifType.14 = mppp-bundle
```

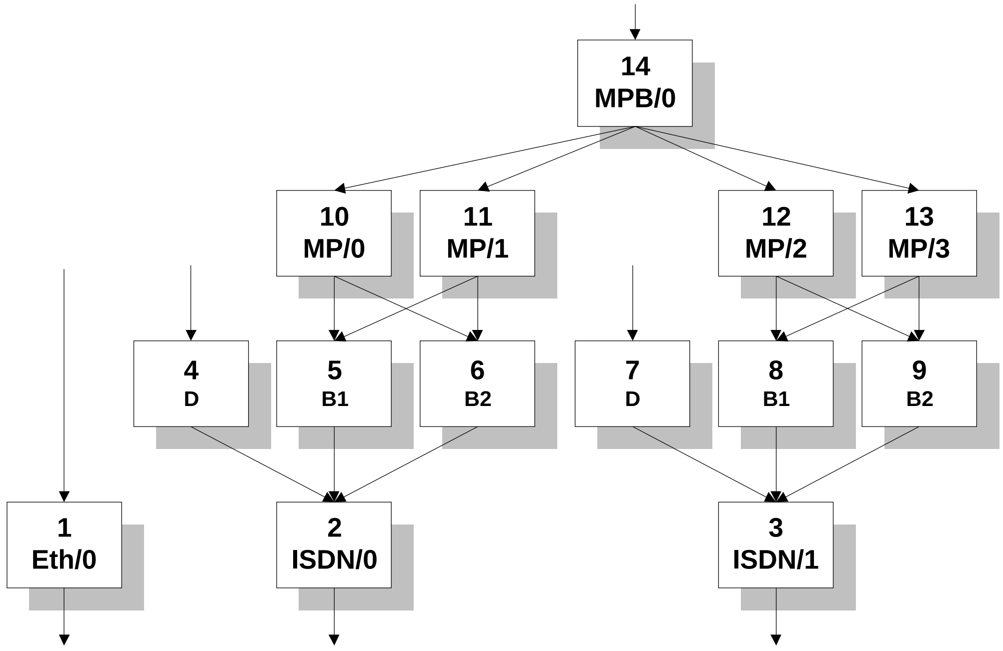
You can determine the hierarchy of the SNMP interface numbers using ifStackStatus:

Example:

```
% walk ifStackStatus
ifStackStatus.0.1 = "active"
ifStackStatus.0.4 = "active"
ifStackStatus.0.7 = "active"
ifStackStatus.0.14 = "active"
ifStackStatus.1.0 = "active"
ifStackStatus.2.0 = "active"
ifStackStatus.3.0 = "active"
ifStackStatus.4.2 = "active"
ifStackStatus.5.2 = "active"
ifStackStatus.6.2 = "active"
ifStackStatus.7.3 = "active"
ifStackStatus.8.3 = "active"
ifStackStatus.9.3 = "active"
ifStackStatus.10.5 = "active"
ifStackStatus.10.6 = "active"
ifStackStatus.11.5 = "active"
ifStackStatus.11.6 = "active"
ifStackStatus.12.8 = "active"
ifStackStatus.12.9 = "active"
ifStackStatus.13.8 = "active"
ifStackStatus.13.9 = "active"
ifStackStatus.14.10 = "active"
ifStackStatus.14.11 = "active"
ifStackStatus.14.12 = "active"
ifStackStatus.14.13 = "active"
```

The two numbers in each line indicate the higher and lower interface number in the hierarchy. Therefore, 0.1 indicates that there is no interface “above” interface 1. Similarly, 1.0 indicates that there is no interface “below” interface 1, while 8.3 indicates that interface 8 is mapped to interface 3. Figure D–2 illustrates the hierarchy described in this listing.

Figure D–2 SNMP Interface Number Hierarchy



LKG-10685-97V

We can see that the interfaces 1, 2, and 3 are the physical interfaces (1 for ETH/0, 2 for ISDN/0 and 3 for ISDN/1). The values 4.2, 5.2, and 6.2 indicate that SNMP interfaces 4, 5, and 6 are mapped over interface 2 (ISDN/0). These are the ISDN D-channel and 2 B-channels. Similarly, interfaces 7, 8, and 9 are mapped over interface 3 (ISDN/1). There are four dial circuits (10, 11, 12, and 13). Dial circuits 10 and 11 are both mapped over both of the B-channels of ISDN/0 (interfaces 5 and 6). Similarly, 12 and 13 are both mapped over interfaces 8 and 9. Interface 14 is mapped over the four dial circuits, so this is the Multilink PPP Bundle device MPB/0.

For more information about the way SNMP allocates interface numbers, refer to RFC 1573.

D.1.2 PPP MIB

Table D–3 lists the group objects of the PPP MIB supported by DIGITAL. This section describes the level of support provided for extensions of the Link Control Protocol of the Point-to-Point Protocol. These extensions are defined by the Internet standard RFC 1471.

Table D–3 PPP MIB Groups Supported

PPP Group Objects	Supported	Not Supported
PPP Link Group (Link Status Group only)	x	
PPP Link Quality Reporting Group		x
PPP Link Quality Reporting Extensions Group		x
PPP IP Group		x
PPP Bridge Group		x
PPP Security Group		x

Table D–4 lists the objects supported within the PPP Link Group specified in RFC 1471.

Table D–4 PPP Link Group Objects Supported

PPP Link Group Objects	Supported	Not Supported
PPP Link Status Physical Index[1]	x	
PPP Link Status Bad Addresses	x	
PPP Link Status Bad Controls	x	
PPP Link Status Packet Too Longs	x	
PPP Link Status BadFCSs	x	
PPP Link Status Local MRU	x	
PPP Link Status Remote MRU	x	
PPP Link Status Local To Peer ACC Map	x	
PPP Link Status Peer To Local ACC Map	x	
PPP Link Status Local To Remote Protocol Compression	x	
PPP Link Status Remote to Local Protocol Compression	x	
PPP Link Status Local To Remote ACC Compression	x	
PPP Link Status Remote To Local ACC Compression	x	
PPP Link Status Transmit Fcs Size	x	

Table D–4 PPP Link Group Objects Supported (Continued)

PPP Link Group Objects	Supported	Not Supported
PPP Link Status Receive Fcs Size	x	
PPP Link Config Table (Run-time SNMP Objects)		x

[1] Each Serial Line that supports PPP has two entries in the Interface table. One entry is associated with the physical hardware link over which PPP is running. The other entry represents the PPP layer running over that link.
For example, if the router contains four interfaces, two of which run PPP over RS-232 ports, the number of interfaces reported in the Interface Table is six. Six interface numbers are defined, four for the network interfaces and an additional two designating the PPP-layer links.

D.2 DIGITAL Proprietary MIBs

The full text of the proprietary MIBs used by DIGITAL are provided in files on the Distributed Routing Software V3.0 CDROM supplied with your router. The filenames of these proprietary MIBs are listed in Table D–5.

Table D–5 DIGITAL Proprietary MIBs Supported

File Name	MIB Description
dec-comet-mib-v1-2.txt	Distributed Routing Software MIB. The main group of objects in this MIB is **ELS**.
dec-prot-brs-mib-v1-0.txt	Bandwidth Reservation System MIB.
telesaving-mib-v1-0.txt	Telesaving MIB.

They are also available from the DIGITAL Network Products home page on the World Wide Web at the following addresses:

North America:	**http://www.networks.digital.com**
Europe:	**http://www.networks.europe.digital.com**
Asia Pacific:	**http://www.networks.digital.com.au**

D.2.1 Telesaving MIB

There are two groups of objects in the Telesaving MIB: **System** and **Peer**. The *System* group is used for system-wide information, and the *Peer* group is used for circuit level information.

D.2.1.1 Telesaving Monitoring Tool

There is a web-based Telesaving Monitoring tool provided with the Router Configurator software. This monitoring tool displays a graphic image of the budget usage at system and circuit level on the router, and can display a snapshot of the Telesaving MIB System group contents for the router or the Peer Group contents for each circuit. The display arranges the objects into sets of objects which relate to the same function.

D.2.1.2 Telesaving MIB System Groups

Table D–6 lists the System group objects of the Telesaving MIB in the functional sets that are displayed by the monitoring tool.

Table D–6 Telesaving MIB System Group

Functional Set	Object
Budget Details	Budget Status Budget Period Budget Period Used Budget Period Used Percent Budget Charge Units Budget Charge Units Used Budget Charge Units Percent Budget Time Budget Time Used Budget Time Used Percent Threshold Percentage Threshold Alarm Action Budget Overdraft Units Budget Overdraft Units Used Budget Overdraft Time Budget Overdraft Time Used Budget Alarm Status Budget Profile Status Call Policy Action

Table D–6 Telesaving MIB System Group (Continued)

Functional Set	Object
Calls Blocked Information	Out Calls blocked by budget In Calls blocked by budget Out Calls blocked by Blocking In Calls blocked by Blocking No Peer Rejects
Calls Connected Information	Total Short Calls Telesaving Time Total Out Calls Total Connect Time Out Average Out Duration Total In Calls Total Connect Time In Average In Duration
Call Initiation Information	Ip Call Initiations Ipx Call Initiations Osi Call Initiations Iv Call Initiations Apx Call Initiations Call Back Response Call Initiations Trigger Rip Initiations
Call Utilisation Information	Overall Compression Ratio Overall Decompression Ratio Tx Utilisation Percent Rx Utilisation Percent

Table D–7 lists the Peer group objects of the Telesaving MIB in the functional sets that are displayed by the monitoring tool when you access a circuit.

Table D–7 Telesaving MIB Peer Group

Functional Set	Object
Budget Details	Budget Status Budget Charge Units Budget Charge Units Used Budget Charge Units Used Percent Budget Time Budget Time Used Budget Time Used Percent Threshold Percentage Threshold Alarm Action Budget Overdraft Units Budget Overdraft Units Used Budget Overdraft Time Budget Overdraft Time Used Budget Alarm Status Budget Profile Status Call Policy Action
Telesaving Feature Setting	IMCT Idle Blocking Action Call Back Action Compression Options Minimum Channels Maximum Channels
Calls Blocked/Cleared Information	Out Calls Blocked by Budget In Calls Blocked by Budget Out Calls Blocked by Blocking In Calls Blocked by Blocking Last Disconnect Reason Last Call Failure Reason Last Network Cause Code Last Network Diag Code Last Incoming Reject Reason

Table D–7 Telesaving MIB Peer Group (Continued)

Functional Set	Object
Calls Connected History Information	Total Short Calls Telesaving Time Total Out Calls Total Connect Time Out Average Out Duration Total In Calls Total Connect Time In Average In Duration
Call Initiation History Information	Time since last call Last Call Initiation Reason IP Call Initiations IPX Call Initiations DECnet Phase V Call Initiations DECnet Phase IV Call Initiations Appletalk Call Initiations Call Back Response Call Initiations Trigger Rip Initiations
Call Utilisation History Infomration	Overall Compression Ratio Overall Decompression Ratio Recent Compression Ratio Recent Decompression Ratio Tx Utilisation Percent Rx Utilisation Percent
Current Call Information	Current Call Direction Current Call Duration Current Channels Current Call Raw Tx Bytes Current Call Data Tx Bytes Current Call Raw Rx Bytes Current Call Data Rx Bytes Current Call Compression Ratio Current Call Decompression Ratio
Configuration Information	Called Name Called Address Called SubAddress Calling Name Calling Address Calling SubAddress

Index

Page reference numbers in bold type indicate a reference to a command description.