Router Products Command Reference

The products and specifications, configurations, and other technical information regarding the products contained in this manual are subject to change without notice. All statements, technical information, and recommendations contained in this manual are believed to be accurate and reliable but are presented without warranty of any kind, express or implied, and users must take full responsibility for their application of any products specified in this manual.

This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual for this device, may cause interference to radio communications. This equipment has been tested and found to comply with the limits for a Class A computing device pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference when operated in a commercial environment. Operation of this equipment in a residential area is likely to cause interference, in which case users at their own expense will be required to take whatever measures may be required to correct the interference.

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.

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.

The following are trademarks of Digital Equipment Corporation: DDCMP, DEC, DECnet, DECNIS, DECserver, DECsystem, DECwindows, Digital, DNA, OpenVMS, ULTRIX, VAX, VAXstation, VMS, VMScluster, and the DIGITAL logo.

Portions of this document is used with permission of Cisco Systems, Incorporated. Copyright © 1990 - 1995, Cisco Systems, Inc.

The following third-party software may be included with your product and will be subject to the software license agreement:

CiscoWorks software and documentation are based in part on HP OpenView under license from the Hewlett-Packard Company. HP OpenView is a trademark of the Hewlett-Packard Company. Copyright © 1992, 1993 Hewlett-Packard Company.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

Network Time Protocol (NTP). Copyright © 1992, David L. Mills. The University of Delaware makes no representations about the suitability of this software for any purpose.

Point-to-Point Protocol. Copyright © 1989, Carnegie-Mellon University. All rights reserved. The name of the University may not be used to endorse or promote products derived from this software without specific prior written permission.

The Cisco implementation of TN3270 is an adaptation of the tn3270, curses, and termcap programs developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981-1988, Regents of the University of California.

Cisco incorporates Fastmac software in some Token Ring products. Fastmac software is licensed to Cisco by Madge Networks Limited.

XRemote is a trademark of Network Computing Devices, Inc. Copyright © 1989, Network Computing Devices, Inc., Mountain View, California. NCD makes no representations about the suitability of this software for any purpose.

TABLE OF CONTENTS

PART 2
System and Interface Configuration and Management

Chapter 3

PART 5
Bridging

Chapter 22

LIST OF TABLES

CHAPTER 10

ISDN Commands

This chapter describes the commands available to configure your router for Integrated Services Digital Network (ISDN) operations.

For ISDN configuration information and examples, refer to the chapter entitled "Configuring ISDN" in the *Router Products Configuration Guide*.

For information about the Channel Interface Processor (CIP), see the chapter entitled "IBM Channel Attach Commands" in this manual. The CIP is described in a separate chapter because of the interrelation of host system configuration values and router configuration values.

For hardware technical descriptions, and for information about installing the router interfaces, refer to the hardware installation and maintenance publication for your particular product.

interface bri

To configure a Basic Rate Interface (BRI) interface and enter interface configuration mode, use the **interface bri** global configuration command.

interface bri *number*

To configure a BRI subinterface, use the following form of the **interface bri** global configuration command.

interface bri *number.subinterface-number* [**multipoint** | **point-to-point**]

Syntax Description

number	Port, connector, or interface card number. The numbers are assigned at the factory at the time of installation or when added to a system, and can be displayed with the **show interfaces** command.
.*subinterface-number*	Subinterface number in the range 1 to 4294967293. The *number* that precedes the period (.) must match the *number* this subinterface belongs to.
multipoint | **point-to-point**	(Optional) Specifies a multipoint or point-to-point subinterface. The default is **multipoint**.

Default

The default mode for subinterfaces is multipoint.

Command Mode

Global configuration

Usage Guidelines

Subinterfaces can be configured to support partially meshed Frame Relay networks (refer to the "Configuring Frame Relay" chapter in the *Router Products Configuration Guide*).

Example

The following example configures BRI 0 to call and receive calls from two sites, use PPP encapsulation on outgoing calls, and use CHAP authentication on incoming calls.

```
interface bri 0
encapsulation ppp
no keepalive
dialer map ip 131.108.36.10 name EB1 234
dialer map ip 131.108 36.9 name EB2 456
dialer-group 1
isdn spid1 0146334600
isdn spid2 0146334610
isdn T200 1000
ppp authentication chap
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

dialer map†
dialer-group†
encapsulation ppp†
isdn spid1
isdn spid2
ppp authentication chap†
ppp authentication pap†
show interfaces bri

isdn answer1, isdn answer2

To have the router verify a called-party number or subaddress number in the incoming setup message for ISDN BRI calls, if the number is delivered by the switch, use the **isdn answer1** interface configuration command. To remove the verification request, use the **no** form of this command.

isdn answer1 [*called-party-number*][**:***subaddress*]
no isdn answer1 [*called-party-number*][**:***subaddress*]

To have the router verify an additional called-party number or subaddress number in the incoming setup message for ISDN BRI calls, if the number is delivered by the switch, use the **isdn answer2** interface configuration command. To remove this second verification request, use the **no** form of this command.

isdn answer2 [*called-party-number*][**:***subaddress*]
no isdn answer2 [*called-party-number*][**:***subaddress*]

Syntax Description

called-party-number	(Optional) Telephone number of the called party. At least one of the *called-party-number* or *subaddress* must be specified.
:	Identifies the number that follows as a subaddress. Use the colon (:) when you configure both the called party number and the subaddress or when you configure only the subaddress.
subaddress	(Optional) Subaddress number, 20 or fewer characters long, used for ISDN multipoint connections. At least one of the *called-party-number* or *subaddress* must be specified.

Default

The router does not verify the called-party or subaddress number.

Command Mode

Interface configuration

Usage Guidelines

If you do not specify the **isdn answer1** or **isdn answer2** command, all calls are processed/accepted. If you specify the **isdn answer1** or **isdn answer2** command, the router must verify the incoming called-party number and the subaddress before processing/accepting the call. The verification proceeds from right to left for the called-party number; it also proceeds from right to left for the subaddress number.

It is possible to configure just the called-party number or just the subaddress. In such a case, only that part is verified. To configure a subaddress only, include the colon (:) before the subaddress number.

You can declare a digit a “don’t care” digit by configuring it as an “x” or “X”. In such a case, any incoming digit is allowed.

Examples

In the following example, 5552222 is the called-party number and 1234 is the subaddress:

```
interface bri 0
isdn answer1 5552222:1234
```

In the following example, only the subaddress is configured:

```
interface bri 0
isdn answer1 :1234
```

isdn caller

To configure ISDN caller ID screening, use the **isdn caller** interface configuration command. To disable this feature, use the **no** form of this command.

isdn caller *number*
no isdn caller *number*

Syntax Description

number	Telephone number for which to screen. Specify an "x" to represent a single "don't-care" character. The maximum length of each number is 25 characters.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

This command configures the router to accept calls from the specified number.

Caller ID screening is available on Cisco 7000 series, Cisco 4000 series, Cisco 3000 series, and Cisco 2500 series routers that have one or more BRIs.

The maximum length of each number is 25 characters. You can specify up to 64 numbers per interface.

Note Caller ID screening requires a local switch that is capable of delivering the caller ID to the router. If you enable caller ID screening but do not have such a switch, no calls will be allowed in.

Examples

The following example configures the router to accept a call with a delivered caller ID equal to 4155551234:

```
isdn caller 4155551234
```

The following example configures the router to accept a call with a delivered caller ID having 41555512 and any numbers in the last two positions:

```
isdn caller 41555512xx
```

Related Command

A dagger (†) indicates that the command is documented in another chapter.

show dialer †

isdn calling-number

To configure an Australian basic-ts013 ISDN BRI interface to present a billing number of the device making the outgoing call, use the **isdn calling-number** interface configuration command. To remove a previously configured calling number, use the **no** form of this command.

isdn calling-number *calling-number*
no isdn calling number

Syntax Description

calling-number	Number of the device making the outgoing call; only one entry is allowed and it is limited to 16 digits.

Default

No calling number is presented.

Command Mode

Interface configuration

Usage Guidelines

An interface can have only one ISDN calling-number entry.

This command is intended for use only in Australia because the Australian network offers better pricing on calls in which devices present the calling number (that is, the billing number).

This command can be used only with Australian basic-ts013 switch types.

Example

In the following example, the ISDN BRI interface is configured to present the number 5551212 when it makes outgoing calls:

```
interface bri 0
isdn calling-number 5551212
```

Related Command

interface bri

isdn not-end-to-end

For incoming calls, to override the speed that the network reports it will use to deliver the call data, use the **isdn not-end-to-end** interface configuration command.

isdn not-end-to-end {**56** | **64**}

Command Syntax

56 | **64** Line speed used for incoming calls that are not ISDN from end to end.

Default

The default line speed is 64 kbps.

Command Mode

Interface configuration

Usage Guidelines

This command is useful when calls originate at 56 kbps, but the network delivers the calls as 64 kbps calls. If calls originate at one speed and are delivered at another, a speed mismatch occurs and no data can be transferred.

Example

In the following example, the line speed for incoming calls is set to 56 kbps:

```
isdn not-end-to-end 56
```

isdn spid1

Use the **isdn spid1** interface configuration command to define at the router the service profile identifier (SPID) number that has been assigned by the ISDN service provider for the B1 channel. Use the **no isdn spid1** command to disable the specified SPID, thereby preventing access to the switch. If you include the LDN in the **no** form of this command, the access to the switch is permitted, but the other B-channel may not be able to receive incoming calls.

isdn spid1 *spid-number* [*ldn*]
no isdn spid1 *spid-number* [*ldn*]

Syntax Description

spid-number	Number identifying the service to which you have subscribed. This value is assigned by the ISDN service provider and is usually a ten-digit telephone number with some extra digits.
ldn	(Optional) Local directory number, as delivered by the service provider in the incoming setup message. This is a seven-digit number assigned by the service provider.

Default

No SPID number is defined.

Command Mode

Interface configuration

Usage Guidelines

This command is required for DMS-100 and National ISDN-1 (NI-1) switches only.

You must define the LDN if you want to receive any incoming calls on the B2-channel. The ISDN switch checks for the LDN to determine whether both channels can be used to transmit and receive data. If the LDN is not present, then only the B1-channel can be used for full-duplex communication. However, the other channel can still be used for making outgoing calls.

Example

The following example defines, on the router, a SPID and LDN for the B1 channel:

```
isdn spid1 415555121301 5551215
```

isdn spid2

Use the **isdn spid2** interface configuration command to define at the router the SPID number that has been assigned by the ISDN service provider for the B2 channel. Use the **no isdn spid2** command to disable the specified SPID, thereby preventing access to the switch. If you include the LDN in the **no** form of this command, the access to the switch is permitted, but the other B-channel might not be able to receive incoming calls.

isdn spid2 *spid-number* [*ldn*]
no isdn spid2 *spid-number* [*ldn*]

Syntax Description

spid-number	Number identifying the service to which you have subscribed. This value is assigned by the ISDN service provider and is usually a ten-digit telephone number with some extra digits.
ldn	(Optional) Local directory number, as delivered by the service provider in the incoming setup message. This is a seven-digit number also assigned by the service provider.

Default

No SPID number is defined.

Command Mode

Interface configuration

Usage Guidelines

This command is required for DMS-100 and National ISDN-1 (NI-1) switches only.

You must define the LDN if you want to receive any incoming calls on the B1-channel. The ISDN switch checks for the LDN to determine whether both channels can be used to transmit and receive data. If the LDN is not present, then only the B2-channel can be used for full-duplex communication. However, the other channel can still be used for making outgoing calls.

Example

The following example defines, on the router, a SPID and LDN for the B2 channel:

```
isdn spid2 415555121202 5551214
```

isdn switch-type

To configure a central office switch on the ISDN interface, use the **isdn switch-type** global configuration command.

isdn switch-type *switch-type*

Syntax Description

switch-type	Service provider switch type; see the "ISDN Service Provider Switch Types" table for a list of supported switches.

Default

The switch type defaults to **none**, which disables the switch on the ISDN interface.

Command Mode

Global configuration

Usage Guidelines

To disable the switch on the ISDN interface, specify **isdn switch-type none**.

Table 10-1 lists supported switch types by geographic area.

Table 10-1 ISDN Service Provider Switch Types

Keywords by Area	Switch Type
none	No switch defined
Australia	
basic-ts013	Australian TS013 switches
Europe	
basic-1tr6	German 1TR6 ISDN switches
basic-nwnet3	Norway NET3 switches (phase 1)
basic-net3	NET3 ISDN switches (UK and others)
primary-net5	NET5 switches (UK and Europe)
vn2	French VN2 ISDN switches
vn3	French VN3 ISDN switches
Japan	
ntt	Japanese NTT ISDN switches
primary-ntt	Japanese ISDN PRI switches
North America	
basic-5ess	AT&T basic rate switches
basic-dms100	NT DMS-100 basic rate switches
basic-ni1	National ISDN-1 switches

Keywords by Area	Switch Type
primary-4ess	AT&T 4ESS switch type for the U.S. (ISDN PRI only)
primary-5ess	AT&T 5ESS switch type for the U.S. (ISDN PRI only)
primary-dms100	NT DMS-100 switch type for the U.S. (ISDN PRI only)
New Zealand	
basic-nznet3	New Zealand Net3 switches

Example

The following example configures the French VN3 ISDN switch type:

```
isdn switch-type vn3
```

isdn tei

To configure when ISDN Layer 2 terminal endpoint identifier (TEI) negotiation should occur, use the **isdn tei** global configuration command. Use the **no** form of this command to restore the default.

isdn tei [**first-call** | **powerup**]
no isdn tei

Syntax Description

first-call	(Optional) ISDN TEI negotiation should occur when the first ISDN call is placed or received.
powerup	(Optional) ISDN TEI negotiation should occur when the router is powered on.

Default

powerup

Command Mode

Global configuration

Usage Guidelines

Use this command with care. This command is only used for BRI configuration.

Example

The following example configures the router to negotiate TEI when the first ISDN call is placed or received:

```
isdn tei first-call
```

linecode b8zs

Use the **linecode b8zs** controller configuration command to select the B8ZS line-code type for the T1 line attached to an ISDN PRI.

linecode b8zs

Syntax Description

This command has no arguments or keywords.

Command Mode

Controller configuration

Usage Guidelines

This command is used in configurations where the router is intended to communicate with a T1 fractional data line.

Example

The following example specifies B8ZS as the line-code type:

```
linecode b8zs
```

pri-group

To specify ISDN Primary Rate Interface (PRI) on a channelized T1 card on the Cisco 7000 series, use the **pri-group** controller configuration command. Use the **no** form of this command to remove the ISDN PRI.

pri-group [**timeslots** *range*]
no pri-group

Syntax Description

timeslots *range*	(Optional) Specifies a single range of values from 1 to 23.

Default

Disabled

Command Mode

Controller configuration

Usage Guidelines

When you configure ISDN PRI, you must first specify an ISDN switch type for PRI and a T1 controller.

Example

The following example specifies ISDN PRI on T1 slot 1, port 0:

```
isdn switch-type primary-4ess
controllers t1 1/0
framing esf
linecode b8zs
pri-group timeslots 2-6
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

controllers t1†
framing†
isdn switch-type
linecode

show controllers bri

To display information about the ISDN Basic Rate Interface (BRI), use the **show controllers bri** privileged EXEC command.

show controllers bri *number*

Syntax Description

number	Interface number. The value is 0 through 7 if the router has one BRI NIM or 0 through 15 if the router has two BRI NIMs.

Command Mode

Privileged EXEC

Sample Display

The following is sample output from the **show controllers bri** command:

```
Router# show controllers bri 0

BRI unit 0
D Chan Info:
Layer 1 is ACTIVATED
idb 0x32089C, ds 0x3267D8, reset_mask 0x2
buffer size 1524
RX ring with 2 entries at 0x2101600 : Rxhead 0
00 pak=0x4122E8 ds=0x412444 status=D000 pak_size=0
01 pak=0x410C20 ds=0x410D7C status=F000 pak_size=0
TX ring with 1 entries at 0x2101640: tx_count = 0, tx_head = 0, tx_tail = 0
00 pak=0x000000 ds=0x000000 status=7C00 pak_size=0
0 missed datagrams, 0 overruns, 0 bad frame addresses
0 bad datagram encapsulations, 0 memory errors
0 transmitter underruns
B1 Chan Info:
Layer 1 is ACTIVATED
idb 0x3224E8, ds 0x3268C8, reset_mask 0x0
buffer size 1524
RX ring with 8 entries at 0x2101400 : Rxhead 0
00 pak=0x421FC0 ds=0x42211C status=D000 pak_size=0
01 pak=0x4085E8 ds=0x408744 status=D000 pak_size=0
02 pak=0x422EF0 ds=0x42304C status=D000 pak_size=0
03 pak=0x4148E0 ds=0x414A3C status=D000 pak_size=0
04 pak=0x424D50 ds=0x424EAC status=D000 pak_size=0
05 pak=0x423688 ds=0x4237E4 status=D000 pak_size=0
06 pak=0x41AB98 ds=0x41ACF4 status=D000 pak_size=0
07 pak=0x41A400 ds=0x41A55C status=F000 pak_size=0
TX ring with 4 entries at 0x2101440: tx_count = 0, tx_head = 0, tx_tail = 0
00 pak=0x000000 ds=0x000000 status=5C00 pak_size=0
01 pak=0x000000 ds=0x000000 status=5C00 pak_size=0
02 pak=0x000000 ds=0x000000 status=5C00 pak_size=0
03 pak=0x000000 ds=0x000000 status=7C00 pak_size=0
0 missed datagrams, 0 overruns, 0 bad frame addresses
0 bad datagram encapsulations, 0 memory errors
0 transmitter underruns
B2 Chan Info:
Layer 1 is ACTIVATED
idb 0x324520, ds 0x3269B8, reset_mask 0x2
buffer size 1524
```

```
RX ring with 8 entries at 0x2101500 : Rxhead 0
00 pak=0x40FCF0 ds=0x40FE4C status=D000 pak_size=0
01 pak=0x40E628 ds=0x40E784 status=D000 pak_size=0
02 pak=0x40F558 ds=0x40F6B4 status=D000 pak_size=0
03 pak=0x413218 ds=0x413374 status=D000 pak_size=0
04 pak=0x40EDC0 ds=0x40EF1C status=D000 pak_size=0
05 pak=0x4113B8 ds=0x411514 status=D000 pak_size=0
06 pak=0x416ED8 ds=0x417034 status=D000 pak_size=0
07 pak=0x416740 ds=0x41689C status=F000 pak_size=0
TX ring with 4 entries at 0x2101540: tx_count = 0, tx_head = 0, tx_tail = 0
00 pak=0x000000 ds=0x000000 status=5C00 pak_size=0
01 pak=0x000000 ds=0x000000 status=5C00 pak_size=0
02 pak=0x000000 ds=0x000000 status=5C00 pak_size=0
03 pak=0x000000 ds=0x000000 status=7C00 pak_size=0
0 missed datagrams, 0 overruns, 0 bad frame addresses
0 bad datagram encapsulations, 0 memory errors
0 transmitter underruns
```

Table 10-2 describes the significant fields in the display.

Table 10-2 Show Controllers BRI Field Descriptions

Field	Description
BRI unit 0	Interface type and unit number.
Chan Info	D- and B-channel numbers.
Layer 1 is ACTIVATED	Status can be DEACTIVATED, PENDING ACTIVATION, or ACTIVATED.
idb ds reset_mask	Information about internal data structures and parameters.
buffer size	Number of bytes allocated for buffers.
RX ring with - entries at -	Information about the Receiver Queue.
Rxhead	Start of the Receiver Queue.
pak ds status pak_size	Information about internal data structures and parameters.
TX ring with - entries at -	Information about the Transmitter Queue.
tx_count	Number of packets to transmit.
tx_head	Start of the transmit list.
tx_tail	End of the transmit list.
missed datagrams	Incoming packets missed due to internal errors.
overruns	Number of times the receiver hardware was unable to hand received data to a hardware buffer because the input rate exceeded the receiver's ability to handle the data.
bad frame addresses	Frames received with a CRC error and noninteger number of octets.
bad datagram encapsulations	Packets received with bad encapsulation.
memory errors	Internal DMA memory errors.
transmitter underruns	Number of times that the transmitter has been running faster than the router can handle.

show interfaces bri

Use the **show interfaces bri** privileged EXEC command to display information about the BRI D- and B-channels.

show interfaces bri *number* [*first*] [*last*] [**accounting**]

Syntax Description

number	Interface number. The value is 0 through 7 if the router has one BRI NIM or 0 through 15 if the router has two BRI NIMs. Specifying just the number will display the D-channel for that BRI interface.
first	(Optional) Specifies the first of the B-channels; the value can be either 1 or 2.
last	(Optional) Specifies the last of the B-channels; the value can only be 2, indicating B-channels 1 and 2.
accounting	(Optional) Displays the number of packets of each protocol type that have been sent through the interface.

Command Mode

Privileged EXEC

Usage Guidelines

To obtain D-channel information, use the command without the optional *first* and *last* arguments.

Use the command syntax sample combinations in Table 10-3 to display the associated output.

Table 10-3 Sample Show Interfaces BRI Combinations

Command Syntax	Displays
show interfaces	All interfaces in the router
show interfaces bri 2	Channel D for BRI interface 2
show interfaces bri 4 1	Channel B1 on BRI interface 4
show interfaces bri 4 2	Channel B2 on BRI interface 4
show interfaces bri 4 1 2	Channels B1 and B2 on BRI interface 4
show interfaces bri	Error message: "% Incomplete command."

Sample Display

The following is sample output from the **show interfaces** command for BRI:

```
Router# show interfaces bri 0

BRI0 is up, line protocol is up (spoofing)
   Hardware is BRI
   Internet address is 150.136.190.203, subnet mask is 255.255.255.0
   MTU 1500 bytes, BW 64 Kbit, DLY 20000 usec, rely 255/255, load 1/255
   Encapsulation HDLC, loopback not set, keepalive set (10 sec)
   Last input 0:00:07, output 0:00:00, output hang never
```

```
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
Five minute input rate 0 bits/sec, 0 packets/sec
Five minute output rate 0 bits/sec, 0 packets/sec
    16263 packets input, 1347238 bytes, 0 no buffer
    Received 13983 broadcasts, 0 runts, 0 giants
    2 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 2 abort
    22146 packets output, 2383680 bytes, 0 underruns
    0 output errors, 0 collisions, 2 interface resets, 0 restarts
    1 carrier transitions
```

Table 10-4 describes the fields shown in the display.

Table 10-4 Show Interfaces BRI Field Descriptions

Field	Description
BRI ... is {up | down | administratively down	Indicates whether the interface hardware is currently active (whether line signal is present) and if it has been taken down by an administrator.
line protocol is {up | down | administratively down}	Indicates whether the software processes that handle the line protocol consider the line usable (that is, whether keepalives are successful).
Hardware is	Hardware type.
Internet address is	IP address and subnet mask, followed by packet size.
MTU	Maximum Transmission Unit of the interface.
BW	Bandwidth of the interface in kilobits per second.
DLY	Delay of the interface in microseconds.
rely	Reliability of the interface as a fraction of 255 (255/255 is 100% reliability), calculated as an exponential average over 5 minutes.
load	Load on the interface as a fraction of 255 (255/255 is completely saturated), calculated as an exponential average over 5 minutes.
Encapsulation	Encapsulation method assigned to interface.
loopback	Indicates whether loopback is set or not.
keepalive	Indicates whether keepalives are set or not.
Last input	Number of hours, minutes, and seconds since the last packet was successfully received by an interface. Useful for knowing when a dead interface failed.
output	Number of hours, minutes, and seconds since the last packet was successfully transmitted by an interface.
output hang	Number of hours, minutes, and seconds (or never) since the interface was last reset because of a transmission that took too long. When the number of hours in any of the “last” fields exceeds 24 hours, the number of days and hours is printed. If that field overflows, asterisks are printed.
Output queue, drops Input queue, drops	Number of packets in output and input queues. Each number is followed by a slash, the maximum size of the queue, and the number of packets dropped due to a full queue.
Five minute input rate Five minute output rate	Average number of bits and packets transmitted per second in the last 5 minutes.
packets input	Total number of error-free packets received by the system.

Field	Description
bytes	Total number of bytes, including data and MAC encapsulation, in the error free packets received by the system.
no buffer	Number of received packets discarded because there was no buffer space in the main system. Compare with ignored count. Broadcast storms on Ethernets and bursts of noise on serial lines are often responsible for no input buffer events.
broadcasts	Total number of broadcast or multicast packets received by the interface.
runts	Number of packets that are discarded because they are smaller than the medium's minimum packet size.
giants	Number of packets that are discarded because they exceed the medium's maximum packet size.
input errors	Total number of no buffer, runts, giants, CRCs, frame, overrun, ignored, and abort counts. Other input-related errors can also increment the count, so this sum may not balance with the other counts.
CRC	Cyclic redundancy checksum generated by the originating station or far-end device does not match the checksum calculated from the data received. On a serial link, CRCs usually indicate noise, gain hits, or other transmission problems on the data link.
frame	Number of packets received incorrectly having a CRC error and a noninteger number of octets. On a serial line, this is usually the result of noise or other transmission problems.
overrun	Number of times the serial receiver hardware was unable to hand received data to a hardware buffer because the input rate exceeded the receiver's ability to handle the data.
ignored	Number of received packets ignored by the interface because the interface hardware ran low on internal buffers. Broadcast storms and bursts of noise can cause the ignored count to be increased.
abort	Illegal sequence of one bits on a serial interface. This usually indicates a clocking problem between the serial interface and the data link equipment.
packets output	Total number of messages transmitted by the system.
bytes	Total number of bytes, including data and MAC encapsulation, transmitted by the system.
underruns	Number of times that the transmitter has been running faster than the router can handle. This may never be reported on some interfaces.
output errors	Sum of all errors that prevented the final transmission of datagrams out of the interface being examined. Note that this may not balance with the sum of the enumerated output errors, as some datagrams may have more than one error, and others may have errors that do not fall into any of the specifically tabulated categories.
collisions	Number of collisions. This could happen when you have several devices connected on a multiport line.

Field	Description
interface resets	Number of times an interface has been completely reset. This can happen if packets queued for transmission were not sent within several seconds. On a serial line, this can be caused by a malfunctioning modem that is not supplying the transmit clock signal or by a cable problem. If the system notices that the carrier detect line of a serial interface is up, but the line protocol is down, it periodically resets the interface in an effort to restart it. Interface resets can also occur when an interface is looped back or shut down.
restarts	Number of times the controller was restarted because of errors.
carrier transitions	Number of times the carrier detect signal of a serial interface has changed state. Indicates modem or line problems if the carrier detect line is changing state often.

show isdn

To display the information about memory, Layer 2 and Layer 3 timers and, on the Cisco 7000 series only, to display information about the status of PRI channels, use the **show isdn** global configuration command.

show isdn {**memory** | **timers** | **services**}

Syntax Description

memory	Displays memory pool statistics.This is only used by technical development staff.
timers	Displays the values of Layer 2 and Layer 3 timers.
services	Displays the status of PRI channels. (Cisco 7000 series only).

Command Mode

Global configuration

Sample Display

The following is sample output from the **show isdn timers** command.

```
Router# show isdn timers

ISDN Layer 2 values:
         K    = 0   outstanding I-frames
         N200 = 0   max number of retransmits
         T200 = 0   seconds
         T202 = 2   seconds
         T203 = 0   seconds
ISDN Layer 3 values:
         T303 = 0   seconds
         T305 = 0   seconds
         T308 = 0   seconds
         T310 = 0   seconds
         T313 = 0   seconds
         T316 = 0   seconds
         T318 = 0   seconds
         T319 = 0   seconds
```

The following is sample output from the **show isdn services** command for an ISDN T1 PRI. The channels are displayed in numeric order with channel 1 in the left-most position. Because a T1 has 23 channels only, channels 24 through 31 are shown as unavailable.

```
Router# show isdn services

PRI Channel Statistics:
Dsl 3, Channel (1-31)
  State (0=Idle 1=Propose 2=Busy 3=Reserved 4=Restart 5=Maint)
  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 3 3 3 3 3 3 3
  Channel (1-31) Service (0=Inservice 1=Maint 2=Outofservice)
  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

The following is sample output from the **show isdn services** command for an ISDN E1 PRI. Because channel 16 is a D-channel, it is shown as unavailable for placing calls.

```
Router# show isdn services

PRI Channel Statistics:
Dsl 3, Channel (1-31)
  State (0=Idle 1=Propose 2=Busy 3=Reserved 4=Restart 5=Maint)
  0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
  Channel (1-31) Service (0=Inservice 1=Maint 2=Outofservice)
  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Table 10-5 displays some typical values of the timers shown in the **show isdn timers** display. The values of the timers depend on the switch type and typically are used only for homologation purposes. See the Q.921 specifications for detailed technical definitions of the Layer 2 timers; see the Q.931 specifications for detailed technical definitions of the Layer 3 timers.

Table 10-5 Show ISDN Timers Command Output

Field	Typical Value
ISDN Layer 2 values:	
K = 0 outstanding I-frames	1
N200 = 0 max number of retransmits	3
T200 = 0 seconds	1
T202 = 2 seconds	2
T203 = 0 seconds	10
ISDN Layer 3 values:	
T303 = 0 seconds	4
T305 = 0 seconds	30
T308 = 0 seconds	4
T310 = 0 seconds	40
T313 = 0 seconds	0
T316 = 0 seconds	4
T318 = 0 seconds	4
T319 = 0 seconds	4

Table 10-6 describes the fields shown in the **show isdn services** display.

Table 10-6 Show ISDN Services Command Output

Field	Description
Dsl 3	Digital Services Loop, an interface on Cisco 7000 series routers.
State	
Idle	Channel is available for use.
Propose	Attempting to place or receive a call on this channel.
Busy	Channel is currently in use.

Field	Description
Reserved	Channel is not available for calls to be placed. D-channels are reserved; channels 24 through 31 are unavailable on a T1 PRI.
Restart	Restart message was sent on the channel.
Maint	Channel is in maintenance mode.
Channel Service (1-31)	
Inservice	Channel is available.
Maint	Channel is unavailable.
Outofservice	Network made this channel unavailable.

CHAPTER 11

SMDS Commands

Use the commands in this chapter to configure the Switched Multimegabit Data Service (SMDS), a wide-area networking service offered by some Regional Bell Operating Companies (RBOCs) and MCI.

For SMDS configuration information and examples, refer to the "Configuring SMDS" chapter in the *Router Products Configuration Guide*.

arp

Use the following variation of the **arp** global configuration command to enable ARP entries for static routing over the SMDS network. Use the **no** form of this command to disable this capability.

arp *ip-address smds-address* **smds**
no arp *ip-address smds-address* **smds**

Syntax Description

ip-address	IP address of the remote router.
smds-address	12-digit SMDS address in the dotted notation *nnnn.nnnn.nnnn* (48 bits long).
smds	Enables ARP for SMDS.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

This command requires a 12-digit (48-bit) dotted-format SMDS address. It does not support 15-digit SMDS addresses.

Example

The following example creates a static ARP entry that maps the IP address 131.108.173.28 to the SMDS address C141.5797.1313 on interface serial 0:

```
interface serial 0
arp 131.108.173.28 C141.5797.1313 smds
```

Related Command

smds enable-arp
smds static-map

encapsulation smds

Use the **encapsulation smds** interface configuration command to enable SMDS service on the desired interface.

encapsulation smds

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

The interface to which this command applies must be a serial interface. All subsequent SMDS configuration commands only apply to an interface with encapsulation SMDS.

Note The maximum packet size allowed in the SMDS specifications (TA-772) is 9188. This is larger than the packet size used by servers with most media. The Cisco default MTU size is 1500 bytes to be consistent with Ethernet. However, on HSSI interfaces, the default MTU size is 4470 bytes. If a larger MTU is used, the **mtu** command must be entered before the **encapsulation smds** command.

Keep in mind, however, that the Cisco MCI card has buffer limitations that prevent setting the MTU size higher than 2048, and the HSSI card has buffer limitations that prevent setting the MTU size higher than 4500. Configuring higher settings has caused router inconsistencies and performance problems.

Example

The following example shows how to configure the SMDS service on serial interface 0:

```
interface serial 0
encapsulation smds
```

Related Command

A dagger (†) indicates that the command is documented in another chapter.

mtu†

show arp

Use the **show arp** privileged EXEC command to display the entries in the ARP table for the router.

show arp

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Sample Display

The following is sample output from the **show arp** command:

```
Router# show arp

Protocol   Address          Age (min)   Hardware Addr         Type    Interface

Internet   131.108.42.112   120         0000.a710.4baf        ARPA    Ethernet3
AppleTalk  4028.5           29          0000.0c01.0e56        SNAP    Ethernet2
Internet   131.108.42.114   105         0000.a710.859b        ARPA    Ethernet3
AppleTalk  4028.9           -           0000.0c02.a03c        SNAP    Ethernet2
Internet   131.108.42.121   42          0000.a710.68cd        ARPA    Ethernet3
Internet   131.108.36.9     -           0000.3080.6fd4        SNAP    TokenRing0
AppleTalk  4036.9           -           0000.3080.6fd4        SNAP    TokenRing0
Internet   131.108.33.9     -           c222.2222.2222        SMDS    Serial0
```

Table 11-1 describes significant fields shown in the first line of output in the display.

Table 11-1 Show ARP Field Descriptions

Field	Description
Protocol	Type of network address this entry includes.
Address	Network address that is mapped to the MAC address in this entry.
Age (min)	Interval (in minutes) since this entry was entered in the table, rather than the interval since the entry was last used. (The timeout value is 4 hours.)
Hardware Addr	MAC address mapped to the network address in this entry.
Type	Encapsulation type the router is using for the network address in this entry. Possible values include: • ARPA • SNAP • ETLK (EtherTalk) • SMDS
Interface	Interface associated with this network address.

show smds addresses

Use the **show smds addresses** privileged EXEC command to display the individual addresses and the interface they are associated with.

show smds addresses

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Sample Display

The following is sample output from the **show smds addresses** command:

```
Router# show smds addresses

SMDS address - Serial0   c141.5555.1212.FFFF
```

Table 11-2 describes the fields shown in the display.

Table 11-2 Show SMDS Addresses Field Descriptions

Field	Description
Serial0	Interface to which this SMDS address has been assigned.
c141.5555.1212	SMDS address that has been assigned to the interface.

show smds map

To display all SMDS addresses that are mapped to higher-level protocol addresses, use the **show smds map** privileged EXEC command.

show smds map

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Sample Display

The following is sample output from the **show smds map** command:

```
Router# show smds map

Serial0:  ARP maps to e180.0999.9999.FFFF multicast
Serial0:  IP maps to e180.0999.9999.FFFF 150.108.42.112 255.255.255.0 multicast
Serial0:  XNS 1006.AA00.0400.0C55 maps to c141.5688.1212.FFFF static [broadcast]
```

Table 11-3 describes the fields shown in the output.

Table 11-3 Show SMDS Map Field Descriptions

Field	Description
Serial0	Name of interface on which SMDS has been enabled.
ARP maps to	Higher-level protocol address that maps to this particular SMDS address.
e180.0999.9999.FFFF	SMDS address. Includes all SMDS addresses entered with either the **smds static-map** command (static) and **smds multicast** command (multicast).
150.108.21.112	IP address.
255.255.255.0	Subnet mask for the IP address.

show smds traffic

To display statistics about bad SMDS packets the router has received, use the **show smds traffic** privileged EXEC command.

show smds traffic

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Sample Display

The following is sample output from the **show smds traffic** command:

```
Router# show smds traffic
624363 Input packets
759695 Output packets
2 DXI heartbeat sent
0 DXI heartbeat received
0 DXI DSU polls received
0 DXI DSU polls sent
0 DXI invalid test frames
0 Bad BA size errors
0 Bad Header extension errors
65 Invalid address errors
1 Bad tag errors
```

Table 11-4 describes the fields shown in the output.

Table 11-4 Show SMDS Traffic Field Descriptions

Field	Description
0 Input packets	Number of input packets.
0 Output packets	Number of output packets.
0 DXI heartbeat sent	Number of DXI heartbeat polls transmitted.
0 DXI heartbeat received	Number of DXI heartbeat polls received.
0 DXI DSU polls sent	Number of DXI DSU polls sent.
0 DXI DSU polls received	Number of DXI DSU polls received.
0 DXI invalid test frames	Number of invalid test frames seen.
0 Bad BA size errors	Number of packets that have a size less than 32 bytes or greater than 9188 bytes.
0 DXI Header extension errors	Number of extended SIP L3 header errors.
0 DXI Invalid address errors	Number of address errors

Field	Description
0 Bad tag errors	Status indicating the number of errors that occur when there is a mismatch between the Tag value in the header and the BeTag value in the trailer of an SMDS frame. This usually indicates that there is a misconfiguration (that is, a DXI is connected to a non-DXI) or that the SDSU is scrambling the L2 PDUs.

smds address

To specify the SMDS individual address for a particular interface, use the **smds address** interface configuration command. To remove the address from the configuration file, use the **no** form of this command.

smds address *smds-address*
no smds address *smds-address*

Syntax Description

smds-address	Individual address provided by the SMDS service provider. This address is protocol independent. For more information, see the "Usage Guidelines" section.

Default

No address is specified.

Command Mode

Interface configuration

Usage Guidelines

All addresses for SMDS service are assigned by the service provider, and can be assigned to individuals and groups.

Addresses are entered in the Cisco SMDS configuration software using an E prefix for Multicast addresses and a C prefix for Unicast addresses. Our software expects the addresses to be entered in E.164 format, which is 64 bits. The first 4 bits are the address type and the remaining 60 bits are the address. If the first 4 bits are 1100 (0xC), the address is a unicast SMDS address, which is the address of an individual SMDS host. If the first 4 bits are 1110 (0xE), the address is a multicast SMDS address, which is used when broadcasting a packet to multiple end points. The 60 bits of the address are in binary-coded decimal (BCD) format. Each 4 bits of the address field presents a single telephone number digit, allowing for up to 15 digits. At a minimum, you must specify at least 11 digits (44 bits). Unused bits at the end of this field are filled with ones.

Note If bridging is enabled on any interface, the SMDS address is erased and must be reentered.

Example

The following example specifies an individual address in Ethernet-style notation:

```
interface serial 0
smds address c141.5797.1313.FFFF
```

smds dxi

To enable the DXI 3.2 support, use the **smds dxi** interface configuration command. To disable the DXI 3.2 support, use the **no** form of this command.

smds dxi
no smds dxi

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

Usage Guidelines

Adding this command to the configuration enables the Data Exchange Interface (DXI) version 3.2 mechanism and encapsulates SMDS packets in a DXI frame before they are transmitted. DXI 3.2 adds an additional four bytes to the SMDS packet header to communicate with the SDSU. These bytes specify the frame type. The interface will expect all packets to arrive with DXI encapsulation.

The DXI 3.2 support also includes the heartbeat process as specified in the SIG-TS-001/1991 standard, revision 3.2. The heartbeat (active process) is enabled when both DXI and keepalives are enabled on the interface. The echo (passive process) is enabled when DXI is enabled on the interface. The heartbeat mechanism automatically generates a heartbeat poll frame every 10 seconds. This default value can be changed with the **keepalive** command. The Interim Local Management Interface (ILMI) is not supported.

Note If you are running serial lines back to back, disable keepalive on SMDS interfaces. Otherwise, DXI will declare the link down.

Note Switching in or out of DXI mode causes the IP cache to be cleared. This is necessary to remove all cached IP entries for the serial line being used. Stale entries must be removed to allow the new MAC header with or without DXI framing to be installed in the cache. This is not frequently done and is not considered to be a major performance penalty.

Fast switching of DXI frames is also supported as of this software release.

Example

The following example enables DXI 3.2 on interface HSSI 0:

```
interface hssi 0
encapsulation smds
```

```
smds dxi-mode
smds address C120.1111.2222.FFFF
ip address 131.108.1.30 255.255.255.0
smds multicast ip E180.0999.9999
smds enable-arp
```

Related Command

A dagger (†) indicates that the command is documented in another chapter.

keepalive †

smds enable-arp

To enable dynamic Address Resolution Protocol (ARP), use the **smds enable-arp** interface configuration command. The multicast address for ARP must be set before this command is issued. Once ARP has been enabled, use the **no** form of this command to disable the interface.

smds enable-arp
no smds enable-arp

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Example

The following example enables the dynamic ARP routing table:

```
interface serial 0
ip address 131.108.1.30 255.255.255.0
smds multicast IP E180.0999.9999.2222
smds enable-arp
```

Related Command

arp

smds multicast

To assign a multicast SMDS E.164 address to a higher-level protocol, use the **smds multicast** interface configuration command. To remove an assigned multicast address, use the **no** form of this command with the appropriate address.

smds multicast *protocol smds-address*
no smds multicast *protocol smds-address*

Syntax Description

protocol	Protocol type. See the "SMDS Multicast Supported Protocols" table in the "Usage Guidelines" section for a list of supported protocols and their keywords.
smds-address	SMDS address. Because SMDS does not incorporate broadcast addressing, a group address for a particular protocol must be defined to serve the broadcast function.

Default

No mapping is defined.

Command Mode

Interface configuration

Usage Guidelines

When configuring DECnet, all four DEC keywords (**decnet**, **decnet_router-L1**, **decnet_router-L2**, and **decnet_node**) must be entered in the configuration.

Table 11-5 lists the high-level protocols supported by the **smds multicast** command.

Table 11-5 SMDS Multicast Supported Protocols

Keyword	Protocol
aarp	AppleTalk ARP address
appletalk	AppleTalk
arp	ARP
bridge	Transparent bridging
clns	ISO CLNS
clns_es	Multicast address for all CLNS End Systems
clns_is	Multicast address for all CLNS Intermediate Systems
decnet	DECnet
decnet_node	DECnet multicast address for all end systems
decnet_router-L1	DECnet multicast address for all Level 1 (intra-area) routers
decnet_router-L2	DECnet multicast address for all Level 2 (interarea) routers
ip	IP

Keyword	Protocol
ipx	Novell IPX
vines	Banyan VINES
xns	XNS

For IP, the IP NETwork and MASK fields are no longer required. The router will accepts these arguments, but will ignore the values. These were required commands for the previous Multi-LIS configuration. The router continues to accept the arguments to allow for backward compatibility, but ignores the contents.

Example

The following example maps the IP broadcast address to the SMDS group address E180.0999.9999:

```
interface serial 0
smds multicast IP E180.0999.9999.FFFF
```

smds multicast arp

To map the SMDS address to a multicast address, use the **smds multicast arp** interface configuration command. Use the **no** form of this command to disable this feature.

smds multicast arp *smds-address* [*ip-address mask*]
no smds multicast arp *smds-address* [*ip-address mask*]

Syntax Description

smds-address	SMDS address in E.164 format.
ip-address	(Optional) IP address.
mask	(Optional) Subnet mask for the IP address.

Default

No mapping is defined.

Command Mode

Interface configuration

Usage Guidelines

This command is only used when an ARP server is present on a network. When broadcast ARPs are sent, SMDS first attempts to send the packet to all multicast ARP SMDS addresses. If none exist in the configuration, they are sent to all multicast IP SMDS multicast addresses. If the optional ARP multicast address is missing, each entered IP multicast command will be used for broadcasting.

Example

The following example configures broadcast ARP messages:

```
interface serial 0
smds multicast arp E180.0999.9999.2222
```

Related Command

smds multicast ip

smds multicast bridge

To enable spanning tree updates, use the **smds multicast bridge** interface configuration command. Use the **no** form of this command to disable this function.

smds multicast bridge *smds-address*
no smds multicast bridge *smds-address*

Syntax Description

smds-address	SMDS multicast address in E.164 format.

Default

No multicast SMDS address is defined. Spanning tree updates are disabled for transparent bridging across SMDS networks.

Command Mode

Interface configuration

Usage Guidelines

Transparent bridging of packets across an SMDS network must already be enabled to allow this update function. Enable transparent bridging across an SMDS network by adding an SMDS interface to an active bridge group.

When the **smds multicast bridge** command is added to the configuration, broadcast packets will be encapsulated using the specified SMDS multicast address configured for bridging. All bridge packets are first encapsulated in an 802.3 MAC header before encapsulating in an SMDS L3 header with LLC/SNAP. The EtherType field of the 802.3 header will specify the particular packet enclosed in the bridge datagram.

Broadcast ARP packets are treated differently. Two packets are sent to the multicast address. One is sent using a standard (SMDS) ARP encapsulation, the other is sent with the ARP packet encapsulated in an 802.3 MAC header. The native ARP is sent as a regular ARP broadcast. Standard bridging commands are necessary to enable bridging on an SMDS interface.

Bridging over multiple logical IP subnets (multiLIS) is not supported in IOS Release 10.2. Bridging of IP packets in a multiLIS environment is unpredictable.

This implementation of 802.6 bridging supports the transmission and reception of only 802.3 encapsulated bridge packets. Other encapsulations will be supported in a future release.

Example

In the following example, all broadcast bridge packets will be sent to the configured SMDS multicast address:

```
interface hssi 0
encapsulation smds
smds address C120.1111.2222.FFFF
ip address 131.108.1.30 255.255.255.0
smds multi bridge E180.0999.9999.FFFF
```

smds multicast ip

To map an SMDS group address to a secondary IP address, use the **smds multicast ip** interface configuration command. Use the **no** form of this command to remove the address map.

smds multicast ip *smds-address* [*ip-address mask*]
no smds multicast ip *smds-address* [*ip-address mask*]

Syntax Description

smds-address	SMDS address in E.164 format.
ip-address	(Optional) IP address.
mask	(Optional) Subnet mask for the IP address.

Default

The IP address and mask will default to the primary address of the interface if they are left out of the configuration.

Command Mode

Interface configuration

Usage Guidelines

This command allows a single SMDS interface to be treated as multiple logical IP subnets (MultiLIS). If taking advantage of the MultiLIS support in SMDS, you can use more than one multicast address on the SMDS interface, that is, multiple commands can be entered. However, each **smds multicast ip** command entry must be associated with a different IP address on the SMDS interface.

Broadcasts can be sent on the SMDS interface using the multicast address. By sending broadcasts in this manner, the router is not required to replicate broadcasts messages to every remote host.

In addition, the higher-level protocols such as OSPF and IS-IS can use the multicast capability by sending one update packet or routing packet to the multicast address.

If the optional IP address and mask arguments are not present, the SMDS address and multicast address are associated with the primary IP address of the interface. This allows the command to be backward compatible with earlier versions of the software.

If an ARP multicast address is missing, each entered IP multicast command will be used for broadcasting. The ARP multicast command has the same format as the IP multicast command and is typically used only when an ARP server is present in the network.

Note All routers at the other end of the SMDS cloud must have the MultiLIS capability enabled. A receiving router must have the primary IP network address of the transmitter configured as a secondary IP network. This is required in order for replies to return. IP discards all packets with a destination address not equal to the primary network address on the SMDS interface.

Example

The following example configures an interface that supports two different subnets with different multicast addresses to each network. The first multicast configuration command associates the multicast address with the primary IP address and mask of the interface.

```
interface hssi 0
encapsulation smds
smds address C120.1111.2222.FFFF
ip address 131.108.1.30 255.255.255.0
ip address 131.108.5.30 255.255.255.0 secondary
smds multicast ip E180.0999.9999.FFFF
smds multicast ip E180.0333.3333.FFFF 131.108.5.0 255.255.255.0
smds enable-arp
```

Related Command

smds multicast arp

smds static-map

To configure a static map between an individual SMDS address and a higher-level protocol address, use the **smds static-map** interface configuration command. Use the **no** form of this command with the appropriate arguments to remove the map.

smds static-map *protocol protocol-address smds-address* [**broadcast**]
no smds static-map *protocol protocol-address smds-address* [**broadcast**]

Syntax Description

protocol	Higher-level protocol. It can be one of the following values: **appletalk**, **clns**, **decnet**, **ip**, **ipx**, **vines,** or **xns**.
protocol-address	Address of the higher-level protocol.
smds-address	SMDS address, to complete the mapping.
broadcast	(Optional) Marks the specified protocol address as a candidate for broadcast packets. All broadcast requests will be sent to the unicast SMDS address.

Default

No mapping is defined.

Command Mode

Interface configuration

Usage Guidelines

The **smds static-map** command provides pseudo-broadcasting by allowing the use of broadcasts on those hosts that cannot support SMDS multicast addresses.

Examples

The following example illustrates how to enable pseudo-broadcasting. In addition to broadcasting IP and ARP requests to E180.0999.9999, the device at address C120.4444.9999 will also receive a copy of the broadcast request. The host at address 131.108.1.15 is incapable of receiving multicast packets. The multicasting is simulated with this feature.

```
interface hssi 0
encapsulation smds
smds address C120.1111.2222.FFFF
ip address 131.108.1.30 255.255.255.0
smds static-map ip 131.108.1.15 C120.4444.9999.FFFF broadcast
smds enable-arp
```

The following example illustrates how to enable multicasting. In addition to IP and ARP requests to E180.0999.9999, the device at address C120.4444.9999 will also receive a copy of the multicast request. The host at address 131.108.1.15 is incapable of receiving broadcast packets.

```
interface hssi 0
encapsulation smds
smds address C120.1111.2222.FFFF
```

```
ip address 131.108.1.30 255.255.255.0
smds multicast ip E100.0999.999.FFFF
smds static-map ip 131.108.1.15 C120.4444.9999.FFFF
smds enable-arp
```

CHAPTER 12

X.25 and LAPB Commands

Use the commands in this chapter to configure Link Access Procedure Balanced (LAPB), X.25, DDN X.25, and Blacker Front End (BFE). X.25 provides remote terminal access; encapsulation for the IP, DECnet, XNS, ISO CLNS, AppleTalk, Novell IPX, Banyan VINES, and Apollo Domain protocols; and bridging. X.25 virtual circuits may also be switched between interfaces (local routing), between two router (remote routing or tunneling), and over non-serial media (CMNS).

To translate between X.25 and another protocol, refer to the *Protocol Translator Configuration Guide and Command Reference* publication.

For X.25 and LAPB configuration information and examples, refer to the "Configuring X.25 and LAPB" chapter in the *Router Products Configuration Guide.*

bfe

To allow the router to participate in emergency mode or to end participation in emergency mode when the interface is configured for **x25 bfe-emergency decision** and **x25 bfe-decision ask**, use the **bfe** EXEC command.

bfe {**enter** | **leave**} *type number*

Syntax Description

enter	Causes the router to send a special address translation packet that includes an **enter emergency mode** command to the BFE if the emergency mode window is open. If the BFE is already in emergency mode, this command enables the sending of address translation information.
leave	Disables the sending of address translation information from the router to the BFE when the BFE is in emergency mode.
type	Interface type.
number	Interface number.

Command Mode

EXEC

Example

The following example enables an interface to participate in BFE emergency mode:

```
bfe enter serial 0
```

Related Commands

encapsulation bfex25
x25 bfe-decision
x25 bfe-emergency

clear x25-vc

To clear switched virtual circuits (SVCs) and to reset permanent virtual circuits (PVCs), use the **clear x25-vc** privileged EXEC command. To clear *all* X.25 virtual circuits at once by restarting the packet layer service, use this command without an *lcn* argument.

clear x25-vc *type number* [*lcn*]

Syntax Description

type	Interface type.
number	Interface number.
lcn	(Optional) Virtual circuit.

Command Mode

Privileged EXEC

Example

The following example illustrates how to clear all virtual circuits on an interface:

```
clear x25-vc serial 1
```

Related Command

x25 idle

cmns enable

To enable Connection-Mode Network Service (CMNS) on a nonserial interface, use the **cmns enable** interface configuration command. To disable this capability, use the **no** form of this command.

cmns enable
no cmns enable

Syntax Description

This command has no arguments or keywords.

Default

The CMNS protocol is implicitly enabled whenever an X.25 encapsulation command is included with a serial interface configuration. A particular nonserial interface, however, must be explicitly configured to use CMNS.

Command Mode

Interface configuration

Usage Guidelines

After processing this command on the LAN interfaces (Ethernet, FDDI, and Token Ring), all the X.25-related interface configuration commands are made available.

Example

The following example enables CMNS on interface Ethernet 0:

```
interface ethernet 0
cmns enable
```

Related Command

x25 map cmns

encapsulation lapb

To exchange datagrams over a serial interface using LAPB encapsulation, use the **encapsulation lapb** interface configuration command.

encapsulation lapb [**dte** | **dce**] [**multi** | *protocol*]

Syntax Description

dte	(Optional) Specifies operation as a DTE. This is the default LAPB mode.
dce	(Optional) Specifies operation as a DCE.
multi	(Optional) Specifies use of multiple local-area network (LAN) protocols to be carried on the LAPB line.
protocol	(Optional) A single protocol to be carried on the LAPB line. A single protocol can be one of the following: **apollo**, **appletalk**, **clns** (ISO CLNS), **decnet**, **ip**, **ipx** (Novell IPX), **vines**, and **xns**. IP is the default protocol.

Default

The default serial encapsulation is HDLC. You must explicitly configure a LAPB encapsulation method.

DTE operation is the default LAPB mode. IP is the default protocol.

Command Mode

Interface configuration

Usage Guidelines

LAPB encapsulations are appropriate only for private connections, where you have complete control over both ends of the link. Connections to X.25 networks should use an **x25 encapsulation** configuration, which operates the X.25 Layer 3 protocol above a LAPB Layer 2.

One end of the link must be a logical DCE and the other end a logical DTE. (This assignment is independent of the interface's hardware DTE/DCE identity.)

Both ends of the LAPB link must specify the same protocol encapsulation.

LAPB encapsulation is supported on serial lines configured for dial-on-demand (DDR) routing. It can be configured on DDR synchronous serial and ISDN interfaces and on DDR dialer rotary groups. It is not supported on asynchronous dialer interfaces.

A single-protocol LAPB encapsulation exchanges datagrams of the given protocol, each in a separate LAPB information frame. You should configure the interface with the protocol-specific parameters needed (for example, a link that carries IP traffic will have an IP address defined for the interface).

A multiprotocol LAPB encapsulation can exchange any or all of the protocols allowed for a single-protocol interface. It also exchanges datagrams, each in a separate LAPB information frame, although two bytes of protocol identification data precede the protocol data. You should configure the interface with all of the protocol-specific parameters needed for each protocol carried.

Beginning with Cisco IOS Release 10.3, LAPB encapsulation supports the priority and custom queueing features.

Example

The following example sets the operating mode as DTE and specifies that AppleTalk protocol traffic will be carried on the LAPB line:

```
interface serial 1
encapsulation lapb dte appletalk
```

encapsulation x25

To specify an interface's operation as an X.25 device, use the **encapsulation x25** interface configuration command.

encapsulation x25 [**dte** | **dce**] [**ddn** | **bfe**] | [**ietf**]

Syntax Description

dte	(Optional) Specifies operation as a DTE. This is the default X.25 mode.
dce	(Optional) Specifies operation as a DCE.
ddn	(Optional) Specifies DDN encapsulation on an interface using DDN X.25 standard service
bfe	(Optional) Specifies BFE encapsulation on an interface attached to a Blacker Front End device. Available for BFE operation only.
ietf	(Optional) Specifies that the interface's datagram encapsulation should default to use of the IETF standard method, as defined by RFC 1356.

Default

The default serial encapsulation is HDLC. You must explicitly configure an X.25 encapsulation method.

DTE operation is the default X.25 mode. Cisco's traditional X.25 encapsulation method is the default.

Command Mode

Interface configuration

Usage Guidelines

One end of an X.25 link must be a logical DCE and the other end a logical DTE. (This assignment is independent of the interface's hardware DTE/DCE identity.) Typically, when connecting to a public data network (PDN), the customer equipment acts as the DTE and the PDN attachment acts as the DCE.

Cisco has supported the encapsulation of a number of datagram protocols for quite some time, using a standard means when available and proprietary means when necessary. More recently the IETF adopted a standard, RFC 1356, for encapsulating most types of datagram traffic over X.25. X.25 interfaces use Cisco's traditional method unless explicitly configured for IETF operation; if the **ietf** keyword is specified, that standard will be used unless Cisco's traditional method is explicitly configured. For details see the **x25 map** command.

When an X.25 interface is reconfigured, all of the interface's X.25 parameters are initialized except the **x25 map** commands. The **x25 map** statements that are configured for an interface are not deleted when the encapsulation is changed, so they will be retained if the interface is later reconfigured for X.25 operation.

A router attaching to the Defense Data Network (DDN) or to a Blacker Front End (BFE) device can be configured to use their respective algorithms to convert between IP and X.121 addresses by using the **ddn** or **bfe** options, respectively. An IP address should be assigned to the interface, from which the algorithm will generate the interface's X.121 address; for proper operation, this X.121 address should not be modified.

A router DDN attachment can operate as either a DTE or a DCE device. A BFE attachment can operate only as a DTE device. The **ietf** option is not available if either the **ddn** or **bfe** option is selected.

Example

The following example configures the interface for connection to a Blacker Front End device:

```
interface serial 0
encapsulation x25 bfe
```

lapb interface-outage

To specify a period during which a link will remain connected, even if a brief hardware outage occurs, use the **lapb interface-outage** interface configuration command.

lapb interface-outage *milliseconds*

Syntax Description

milliseconds	Number of milliseconds a hardware outage can last without having the protocol disconnect the service. The default is 0 milliseconds, which disables this feature.

Default

0 milliseconds, which disables this feature.

Command Mode

Interface configuration

Usage Guidelines

If a hardware outage lasts longer than the LAPB hardware outage period you select, normal protocol operations will occur. The link will be declared to be down and, when it is restored, a link set up will be initiated.

Example

The following example sets the interface outage period to 100 milliseconds. The link will remain connected for outages equal to or shorter than that period.

```
encapsulation lapb dte ip
lapb interface-outage 100
```

lapb k

To specify the maximum permissible number of outstanding frames, called the window size, use the **lapb k** interface configuration command.

lapb k *window-size*

Syntax Description

window-size	Frame count. It can be a value from 1 to the modulo size minus 1 (the maximum is 7 if the modulo size is 8; it is 127 if the modulo size is 128). The default is 7 frames.

Default

7 frames

Command Mode

Interface configuration

Usage Guidelines

If the window size is changed while the protocol is up, the new value will take effect only when the protocol is reset. You will be informed that the new value will not take effect immediately.

When using the LAPB modulo 128 mode (extended mode), the window parameter k should be increased to make use of the ability to send a larger number of frames before acknowledgment is required. This is the basis for its ability to achieve greater throughput on high speed links that have a low error rate.

This configured value should match the value configured in the peer X.25 switch. Nonmatching values will cause repeated LAPB REJ frames.

Example

The following example sets the LAPB window size (the k parameter) to ten frames:

```
interface serial 0
lapb modulo
lapb k 10
```

lapb modulo

To specify the LAPB basic (modulo 8) or extended (modulo 128) protocol mode, use the **lapb modulo** interface configuration command.

lapb modulo *modulus*

Syntax Description

modulus	Either 8 or 128. The value 8 specifies LAPB's basic mode; the value 128 specifies LAPB's extended mode. The default is 8.

Default

Modulo 8

Command Mode

Interface configuration

Usage Guidelines

The modulo parameter determines which of LAPB's two modes is to be used. The modulo values derive from the fact that basic mode numbers information frames between 0 and 7, whereas extended mode numbers them between 0 and 127. Basic mode is widely available and is sufficient for most links. Extended mode is an optional LAPB feature that may achieve greater throughput on high-speed links that have a low error rate.

The LAPB operating mode may be set on X.25 links as well as LAPB Links. The X.25 modulo is independent of the LAPB layer modulo. Both ends of a link must use the same LAPB mode.

When using modulo 128 mode, the window parameter k should be increased to make use of the ability to send a larger number of frames before acknowledgment is required. This is the basis for its ability to achieve greater throughput on high-speed links that have a low error rate.

If the modulo value is changed while the protocol is up, the new value will take effect only when the protocol is reset. The operator will be informed that the new value will not take effect immediately.

Example

The following example configures a high-speed X.25 link to use LAPB's extended mode:

```
interface serial 1
encapsulation x25
lapb modulo 128
lapb k 40
clock rate 2000000
```

Related Command

lapb k

lapb n1

To specify the maximum number of bits a frame can hold (the LAPB N1 parameter), use the **lapb n1** interface configuration command.

lapb n1 *bits*

Syntax Description

bits	Number of bits from 1088 through 32840; it must be a multiple of eight.

Default

N1 defaults to the largest value available for the interface, which is determined from the interface MTU (typically 1500 bytes), plus the required overhead (for example, 7 bytes total for standard modulo 8 X.25). An X.25 encapsulation commonly has a default of 12056 bits (1507 bytes or 1503 bytes for an X.25 packet, or 1500 bytes of user data).

Command Mode

Interface configuration

Usage Guidelines

It is not necessary to set N1 to an exact value to support a particular X.25 data packet size, although both ends of a connection should have the same N1 value. The N1 parameter serves to avoid processing of any huge frames that result from a "jabbering" interface, an unlikely event.

The N1 default value corresponds to the hardware interface buffer size. Any changes to this value must allow for an X.25 data packet and LAPB frame overhead.The software supports an X.25 data packet with a maximum packet size plus 3 or 4 bytes of overhead for modulo 8 or 128 operation, respectively, and LAPB frame overhead of 2 bytes of header for modulo 8 operation plus 2 bytes of CRC.

In addition, the various standards bodies specify that N1 be given in bits rather than bytes. While some equipment can be configured using bytes or will automatically adjust for some of the overhead information present, our devices are configured using the true value of N1.

Table 12-1 specifies the *minimum* N1 values needed to support a given X.25 data packet. Note that N1 cannot be set to a value less than what is required to support an X.25 data packet size of 128 bytes under modulo 128 operation. This is because all X.25 implementations must be able to support 128-byte data packets.

Table 12-1 Minimum LAPB N1 Values

Maximum data in X.25 packet	Minimum N1 value for X.25 and LAPB modulo 8	Minimum N1 value for X.25 or LAPB modulo 128	Minimum N1 value for X.25 and LAPB modulo 128
128	1088	1088	1096
256	2104	2112	2120
512	4152	4160	4168

Maximum data in X.25 packet	Minimum N1 value for X.25 and LAPB modulo 8	Minimum N1 value for X.25 or LAPB modulo 128	Minimum N1 value for X.25 and LAPB modulo 128
1024	8248	8256	8264
2048	16440	16448	16456
4096	32824	32832	32840

Configuring N1 to be less than 2104 will generate a warning message that X.25 may have problems because some nondata packets can use up to 259 bytes.

The N1 parameter cannot be set to a value larger than the default without first increasing the hardware maximum transmission unit (MTU) size.

The X.25 software will accept default packet sizes and CALLs that specify maximum packet sizes greater than what the LAPB layer will support, but will negotiate the CALLs placed on the interface to the largest value that can be supported. For switched CALLs, the packet size negotiation takes place end-to-end through the Cisco router so the CALL will not have a maximum packet size that exceeds the capability of either of the two interfaces involved.

Example

The following example sets the N1 bits to 16440:

```
interface serial 0
lapb n1 16440
mtu 2048
```

Related Command

mtu†

lapb n2

To specify the maximum number of times a data frame can be transmitted (the LAPB N2 parameter), use the **lapb n2** interface configuration command.

lapb n2 *tries*

Syntax Description

tries	Transmission count. It can be a value from 1 through 255. The default is 20 transmissions**.**

Default

20 transmissions

Command Mode

Interface configuration

Example

The following example sets the N2 tries to 50:

```
interface serial 0
lapb n2 50
```

lapb protocol

Use the **lapb protocol** interface configuration command to configure the protocol carried on the LAPB line.

lapb protocol *protocol*

Syntax Description

protocol	Protocol, entered by keyword. It can be one of the following: **appletalk**, **apollo**, **clns** (ISO CLNS), **decnet**, **ip**, **ipx** (Novell IPX), **vines**, and **xns**.

Default

IP

Command Mode

Interface configuration

Usage Guidelines

This command is not available when using a multiprotocol LAPB encapsulation.

Example

The following example sets AppleTalk as the only protocol on the LAPB line:

```
interface serial 1
encapsulation lapb
lapb protocol appletalk
```

Related Commands

encapsulation lapb
encapsulation lapb-dce
encapsulation multi-lapb

lapb t1

To set the retransmission timer period (the LAPB T1 parameter), use the **lapb t1** interface configuration command.

lapb t1 *milliseconds*

Syntax Description

milliseconds	Time in milliseconds. It can be a value from 1 through 64000. The default is 3000 milliseconds.

Default

3000 milliseconds

Command Mode

Interface configuration

Usage Guidelines

The retransmission timer determines how long a transmitted frame can remain unacknowledged before the LAPB software polls for an acknowledgment. The design of the LAPB protocol specifies that a frame is presumed to be lost if it is not acknowledged within T1; a T1 value that is too small may result in duplicated control information, which can severely disrupt service.

To determine an optimal value for the retransmission timer, use the privileged EXEC command **ping** to measure the round-trip time of a maximum-sized frame on the link. Multiply this time by a safety factor that takes into account the speed of the link, the link quality, and the distance. A typical safety factor is 1.5. Choosing a larger safety factor can result in slower data transfer if the line is noisy. However, this disadvantage is minor compared to the excessive retransmissions and effective bandwidth reduction caused by a timer setting that is too small.

Example

The following example sets the T1 retransmission timer to 2,000 milliseconds:

```
interface serial 0
lapb t1 2000
```

lapb t4

To set the T4 idle timer, after which the router sends out a Poll packet to determine whether the link has suffered an unsignaled failure, use the **lapb t4** interface configuration command.

lapb t4 *seconds*

Syntax Description

seconds	Number of seconds between reception of the last frame and the transmission of the outgoing Poll. The default value is 0 seconds, which disables the T4 timer feature.

Defaults

0 seconds, which disables the T4 timer feature.

Command Mode

Interface configuration

Usage Guidelines

Any nonzero T4 duration must be greater than T1, the LAPB retransmission timer period.

Example

The following example will poll the other end of an active link if it has been 10 seconds since the last frame was received; if the far host has failed, the service will be declared down after N2 tries are timed out.

```
interface serial0
encapsulation x25
lapb t4 10
```

Related Commands

lapb n2
lapb t1

show cmns

To display X.25 Level 3 parameters for LAN interfaces (such as Ethernet or Token Ring) and other information pertaining to CMNS traffic activity, use the **show cmns** EXEC command.

show cmns [*type number*]

Syntax Description

type	(Optional) Interface type.
number	(Optional) Interface number.

Command Mode

EXEC

Sample Display

The following is sample output from the **show cmns** command for an Ethernet interface:

```
Router# show cmns
Ethernet1 is administratively down, line protocol is down
  Hardware address is 0000.0c02.5f4c, (bia 0000.0c2.5f4c), state R1
    Modulo 8, idle 0, timer 0, nvc 1
    Window size: input 2, output 2, Packet size: input 128, output 128
    Timer: TH 0
    Channels: Incoming-only none, Two-way 1-4095, Outgoing-only none
    RESTARTs 0/0 CALLs 0+0/0+0/0+0 DIAGs 0/0
```

Table 12-2 describes significant fields shown in the display.

Table 12-2 Show CMNS Field Descriptions

Field	Description
Ethernet1 is down	Interface is currently active and inserted into network (up) or inactive and not inserted (down), or disabled (administratively down).
line protocol is {up | down}	Indicates whether the software processes that handle the line protocol believes the interface is usable.
Hardware address	MAC address for this interface.
bia	Burned-in address.
state R1	State of the interface. R1 is normal ready state (this should always be R1).
modulo 8	Modulo value; determines the packet sequence numbering scheme used.
idle 0	Number of minutes the router waits before closing idle virtual circuits.
timer 0	Value of the interface time; should always be zero.
nvc 1	Maximum number of simultaneous virtual circuits permitted to and from a single host for a particular protocol.
Window size:	Default window sizes (in packets) for the interface. (CMNS cannot originate or terminate calls.)

Field	Description
input 2	Default input window size is two packets.
output 2	Default output window size is two packets.
Packet size:	Default packet sizes for the interface. (CMNS cannot originate or terminate calls).
input 128	Default input maximum packet size is 128 bytes.
output 128	Default output maximum packet size is 128 bytes.
TH 0	X.25 delayed acknowledgment threshold. Should always be zero.
Channels: Incoming-only: none, Two-way: 1-4095, Outgoing-only: none	Virtual circuit ranges for this interface per LLC2 connection.
RESTARTs 0/0	Restarts sent/received.
CALLs 0+0/0+0/0+0	Successful calls + failed calls/calls sent + calls failed/calls received + calls failed.
DIAGs 0/0	Diagnostic messages sent+received.

Related Command

A dagger (†) indicates that the command is documented in another chapter.

show interfaces serial†

show interfaces serial

To display information about a serial interface, use the **show interfaces serial** EXEC command.

show interfaces serial *number*

Syntax Description

number	Interface port number.

Command Mode

EXEC

Sample Displays

The following is a partial sample output from the **show interfaces serial** command for a serial interface using LAPB encapsulation:

```
Router# show interfaces serial 1

LAPB state is SABMSENT, T1 3000, N1 12056, N2 20, k7,Protocol ip
VS 0, VR 0, RCNT 0, Remote VR 0, Retransmissions 2
IFRAMEs 0/0 RNRs 0/0 REJs 0/0 SABMs 3/0 FRMRs 0/0 DISCs 0/0
```

Table 12-3 shows the fields relevant to all LAPB connections.

Table 12-3 Show Interfaces Serial Fields and Descriptions when LAPB is Enabled

Parameter	Description
LAPB state is	State of the LAPB protocol.
T1 3000, N1 12056, ...	Current parameter settings.
Protocol	Protocol encapsulated on a LAPB link; this field is not present on interfaces configured for multiprotocol LAPB or X.25 encapsulations.
VS	Modulo 8 frame number of the next outgoing I-frame.
VR	Modulo 8 frame number of the next I-frame expected to be received.
RCNT	Number of received I-frames that have not yet been acknowledged.
Remote VR	Number of the next I-frame the remote expects to receive.
Retransmissions	Count of current retransmissions due to expiration of T1.
Window is closed	No more frames can be transmitted until some outstanding frames have been acknowledged. This message should be displayed only temporarily.
IFRAMEs	Count of Information frames in the form of sent/received.
RNRs	Count of Receiver Not Ready frames in the form of sent/received.
REJs	Count of Reject frames in the form of sent/received.

Parameter	Description
SABMs	Count of Set Asynchronous Balanced Mode commands in the form of sent/received.
FRMRs	Count of Frame Reject frames in the form of sent/received.
DISCs	Count of Disconnect commands in the form of sent/received.

The following is a partial sample output from the **show interfaces** command for a serial X.25 interface:

```
Router# show interfaces serial 1

X25 address 000000010100, state R1, modulo 8, idle 0, timer 0, nvc 1
  Window size: input 2, output 2, Packet size: input 128, output 128
  Timers: T20 180, T21 200, T22 180, T23 180, TH 0
  Channels: Incoming-only none, Two-way 1-1024, Outgoing-only none
(configuration on RESTART: modulo 8,
  Window size: input 2 output 2, Packet size: input 128, output 128
  Channels: Incoming-only none, Two-way 5-1024, Outgoing-only none)
  RESTARTs 3/2 CALLs 1000+2/1294+190/0+0/ DIAGs 0/0
```

The stability of the X.25 protocol requires that some parameters not be changed without a RESTART of the protocol. Any change to these parameters will be held until a RESTART is sent or received. If any of these parameters will change, the configuration on RESTART information will be output as well as the values that are currently in effect.

Table 12-4 describes significant fields shown in the display.

Table 12-4 Show Interfaces X25 Field Descriptions

Field	Description
X25 address 000000010100	Address used to originate and accept calls.
state R1	State of the interface. Possible values are: • R1 is the normal ready state • R2 is the DTE RESTARTing state • R3 is the DCE RESTARTing state If the state is R2 or R3, the interface is awaiting acknowledgment of a Restart packet.
modulo 8	Modulo value; determines the packet sequence numbering scheme used.
idle 0	Number of minutes the router waits before closing idle virtual circuits that it originated or accepted.
timer 0	Value of the interface timer, which is zero unless the interface state is R2 or R3.
nvc 1	Default maximum number of simultaneous virtual circuits permitted to and from a single host for a particular protocol.
Window size: input 2, output 2	Default window sizes (in packets) for the interface. The **x25 facility** interface configuration command can be used to override these default values for the switched virtual circuits originated by the router.

Field	Description
Packet size: input 128, output 128	Default maximum packet sizes (in bytes) for the interface. The **x25 facility** interface configuration command can be used to override these default values for the switched virtual circuits originated by the router.
Timers: T20 180, T21 200, T22 180, T23 180	Values of the X.25 timers: • T10 through T13 for a DCE device • T20 through T23 for a DTE device
TH0	Packet acknowledgment threshold (in packets). This value determines how many packets are received before sending an explicit acknowledgment; the default value (0) sends an explicit acknowledgment only when the incoming window is full.
Channels: Incoming-only none Two-way 5-1024 Outgoing-only none	Displays the virtual circuit ranges for this interface.
RESTARTs 3/2	Shows RESTART packet statistics for the interface using the format Sent/Received.
CALLs 1000+2/1294+190/0+0	Successful calls + failed calls/calls sent + calls failed/calls received + calls failed.
DIAGs 0/0	Diagnostic messages sent+received.

Related Command

show cmns

show llc2

To display active LLC2 connections, use the **show llc2** EXEC command.

show llc2c

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show llc2** command:

```
Router# show llc2

TokenRing0 DTE=1000.5A59.04F9,400022224444 SAP=04/04, State=NORMAL
V(S)=5, V(R)=5, Last N(R)=5, Local Window=7, Remote Window=127
ack-max=3, n2=8, Next timer in 7768
xid-retry timer 0/60000 ack timer 0/1000
p timer 0/1000 idle timer 7768/10000
rej timer 0/3200 busy timer 0/9600
ack-delay timer 0/3200
CMNS Connections to:
 Address 1000.5A59.04F9 via Ethernet2
 Protocol is up
 Interface type X25-DCE RESTARTS 0/1
 Timers: T10 1 T11 1 T12 1 T13 1
```

The display includes a CMNS addendum, indicating that LLC2 is running with CMNS. When LLC2 is not running with CMNS, the **show llc2** command does not display a CMNS addendum.

Table 12-5 describes significant fields shown in the display.

Table 12-5 Show LLC2 Field Descriptions

Field	Description
TokenRing0	Name of interface on which the session is established.
DTE=1000.5A59.04F9, 400022224444	Address of the station to which the router is talking on this session. (The router's address is the MAC address of the interface on which the connection is established, except when Local Acknowledgment or SDLLC is used, in which case the address used by the router is shown as in this example, following the DTE address and separated by a comma.)
SAP=04/04	Other station's and router's (remote/local) Service Access Point for this connection. The SAP is analogous to a "port number" on the router and allows for multiple sessions between the same two stations.

Field	Description
State=	Current state of the LLC2 session which are any of the following:
ADM	Asynchronous Disconnect Mode—A connection is not established, and either end can begin one.
SETUP	Request to begin a connection has been sent to the remote station, and this station is waiting for a response to that request.
RESET	A previously open connection has been reset because of some error by this station, and this station is waiting for a response to that reset command.
D_CONN	This station has requested a normal, expected, end of communications with the remote, and is waiting for a response to that disconnect request.
ERROR	This station has detected an error in communications and has told the other station about it. This station is waiting for a reply to its posting of this error.
NORMAL	Connection between the two sides is fully established, and normal communication is occurring.
BUSY	Normal communication state exists, except busy conditions on this station make it such that this station cannot receive information frames from the other station at this time.
REJECT	Out-of-sequence frame has been detected on this station, and this station has requested that the other resend this information
AWAIT	Normal communication exists, but this station has had a timer expire, and is trying to recover from it (usually by resending the frame that started the timer).
AWAIT_BUSY	A combination of the AWAIT and BUSY states.
AWAIT_REJ	A combination of the AWAIT and REJECT states.
V(S)=5	Sequence number of the next information frame this station will send.
V(R)=5	Sequence number of the next information frame this station expects to receive from the other station.
Last N (R)=5	Last sequence number of this station's transmitted frames acknowledged by the remote station.
Local Window=7	Number of frames this station may send before requiring an acknowledgment from the remote station.
Remote Window=127	Number of frames this station can accept from the remote.
ack-max=3, n2=8	Value of these parameters, as given in the previous configuration section.

Field	Description
Next timer in 7768	Number of milliseconds before the next timer, for any reason, goes off.
xid-retry timer 0/60000	A series of timer values in the form of next-time/time-between, where "next-time" is the next time, in milliseconds, that the timer will wake, and "time-between" is the time, in milliseconds, between each timer wakeup. A "next-time" of zero indicates that the given timer is not enabled, and will never wake.
CMNS Connections to:	CMNS addendum when LLC2 is running with the CMNS protocol contains the following:
Address 1000.5A59.04F9 via Ethernet2	MAC address of remote station.
Protocol is up	Up indicates the LLC2 and X.25 protocols are in a state where incoming and outgoing Call Requests can be made on this LLC2 connection.
Interface type X25-DCE	One of the following: X25-DCE, X25-DTE, or X25-DXE (either DTE or DCE).
RESTARTS 0/1	Restarts sent/received on this LLC2 connection.
Timers:	T10, T11, T12, T13 (or T20, T21, T22, T23 for DTE); these are Request packet timers. These are similar in function to X.25 parameters of the same name.

show x25 map

To display information about configured address maps, use the **show x25 map** EXEC command.

show x25 map

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

The **show x25 map** command shows information about the following:

- Configured maps (defined by the **x25 map** command)
- Maps implicitly defined by encapsulation PVCs (defined by the **x25 pvc** command)
- Dynamic maps (from the X.25 DDN or BFE operations)
- Temporary maps (from unconfigured CMNS endpoints)

Sample Display

The following is sample output from the **show x25 map** command:

```
Router# show x25 map

Serial0: X.121 1311001 <--> ip 131.108.170.1
  PERMANENT, BROADCAST, 2 VCS: 3 4*
Serial0: X.121 1311005 <--> appletalk 128.1
  PERMANENT
Serial1: X.121 1311005 <--> bridge
  PERMANENT, BROADCAST
Serial2: X.121 001003 <--> apollo 1.3,
         appletalk 1.3,
         ip 131.108.1.3,
         decnet 1.3,
         novell 1.0000.0c04.35df,
         vines 00000001:0003,
         xns 1.0000.0c04.35df,
         clns
  PERMANENT, NVC 8, 1 VC: 1024
```

The display shows that four maps have been configured for the router, two for serial interface 0, one for serial interface 1, and one for the serial interface 2 (which maps eight protocols to the host).

Table 12-6 describes fields shown in the display.

Table 12-6 Show X25 Map Field Description

Field	Description
Serial0	Interface on which this map is configured.
X.121 1311001	X.121 address of the mapped encapsulation host.
ip 131.108.170.1	Type and address of the higher-level protocol(s) mapped to the remote host. Bridge maps do not have a higher-level address; all bridge datagrams are sent to the mapped X.121 address. CLNS maps refer to a configured neighbor as identified by the X.121 address.
PERMANENT	Address-mapping type that has been configured for the interface in this entry. Possible values include the following: • CONSTRUCTED—Derived using the DDN or BFE address conversion scheme. • PERMANENT—Map was entered using the **x25 map** interface configuration command. • PVC—Map was configured using the **x25 pvc** interface command. • TEMPORARY—A temporary map was created for an incoming unconfigured CMNS connection.
BROADCAST	If any options are configured for an address mapping, they will be listed; the example shows a maps that is configured to forward datagram broadcasts to the mapped host.
2 VCs:	If the map has any active virtual circuits, they are identified.
3 4*	Identifies the circuit number of the active virtual circuits. The asterisk (*) marks the virtual circuit last used to send data. Note that a single protocol virtual circuit can be associated with a multiprotocol map.

show x25 remote-red

To display the one-to-one mapping of the host IP addresses and the remote BFE device's IP addresses, use the **show x25 remote-red** EXEC command.

show x25 remote-red

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show x25 remote-red** command:

```
Router# show x25 remote-red
Entry       REMOTE-RED     REMOTE-BLACK   INTERFACE
1           21.0.0.3       21.0.0.7       serial3
2           21.0.0.10      21.0.0.6       serial1
3           21.0.0.24      21.0.0.8       serial3
```

Table 12-7 describes significant fields shown in the display.

Table 12-7 Show X25 Remote-Red Display Field Description

Field	Description
Entry	Address mapping entry.
REMOTE-RED	Host IP address.
REMOTE-BLACK	IP address of the remote BFE device.
INTERFACE	Name of interface through which communication with the remote BFE device will take place.

show x25 route

To display the X.25 routing table, use the **show x25 route** EXEC command.

show x25 route

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show x25 route** command:

```
Router# show x25 route

Number         X.121        CUD        Forward To
1              1311001                 Serial0, 0 uses
2              1311002                 131.108.170.10, 0 uses
3              1311003      00         alias Serial0, 2 uses
```

Table 12-8 describes significant fields shown in the display.

Table 12-8 Show X25 Route Display Field Description

Field	Description
Number	Number identifying the entry in the X.25 routing table.
X.121 address	X.121 address pattern associated with this entry.
CUD	Call User Data, if any, that has been configured for this route.
Forward To	Router interface or IP address to which the router will forward a CALL destined for the X.121 address pattern in this entry. This field also includes the number of uses of this route.

Related Command

x25 route

show x25 vc

To display information about active switched virtual circuits (SVCs) and permanent virtual circuits (PVCs), use the **show x25 vc** EXEC command.

show x25 vc [*lcn*]

Syntax Description

lcn	(Optional) Logical channel number (LCN).

Command Mode

EXEC

Usage Guidelines

To examine a particular virtual circuit, add an LCN argument to the **show x25 vc** command.

This command displays information about virtual circuits that are used for any of the following:

- Encapsulation traffic
- Locally switched traffic
- Remotely switched traffic
- CMNS switched traffic

The connectivity information displayed will vary according to the traffic carried by the virtual circuit. For multiprotocol circuits, the output varies depending on the number and identity of the protocols mapped to the X.121 address and the encapsulation method selected for the circuit.

Sample Displays

This section provides three sample displays and tables that describe the fields in each display.

The following sample display shows a virtual circuit that is being used to encapsulate traffic between the router and a remote host:

```
Router# show x25 vc 1024

SVC 1024,  State: D1,  Interface: Serial0
 Started 0:00:31, last input 0:00:31, output 0:00:31
 Connects 170090 <-->
    compressedtcp 131.108.170.90
    ip 131.108.170.90
 multiprotocol CUD PID, standard Tx data PID, Reverse charged
 Window size input: 2, output: 2
 Packet size input: 128, output: 128
 PS: 5  PR: 5  ACK: 4  Remote PR: 5  RCNT: 1  RNR: FALSE
 Window is closed
 Retransmits: 0  Timer (secs): 0  Reassembly (bytes): 0
 Held Fragments/Packets: 0/0
 Bytes 505/505 Packets 5/5 Resets 0/0 RNRs 0/0 REJs 0/0 INTs 0/0
```

Table 12-9 describes the general fields shown in the output; Table 12-10 describes the fields specific to encapsulation virtual circuits shown in the output.

Table 12-9 Show X25 VC Field Descriptions

Field	Description
SVC 1024	Identifies the type (switched or permanent) and the number of the virtual circuit.
State	State of the virtual circuit (which is independent of the states of other virtual circuits); D1 is the normal ready state. (See the International Telecommunication Union Telecommunication Standardization Sector (ITU-T)[1] X.25 Recommendation for a description of virtual circuit states.)
Interface	Interface or subinterface on which the virtual circuit is established.
Started	Time elapsed since the virtual circuit was created.
last input	Time of last input.
output	Shows time of last output.
Connects...<- ->...	Describes the traffic-specific connection information. See Table 12-10, Table 12-11, and Table 12-12 for more information.
Window size	Window sizes for the virtual circuit.
Packet size	Maximum packet sizes for the virtual circuit.
PS	Current send sequence number.
PR	Current receive sequence number.
ACK	Last acknowledged incoming packet.
Remote PR	Last PR number received from the other end of the circuit.
RCNT	Count of unacknowledged input packets.
RNR	State of the Receiver Not Ready flag; this field is true if the network sends a receiver-not-ready packet.
Window is closed	This line appears if the router cannot transmit any more packets until the X.25 layer 3 peer has acknowledged some outstanding packets.
Retransmits	Number of times a supervisory packet (RESET or CLEAR) has been retransmitted.
Timer	A nonzero time value indicates that a control packet has not been acknowledged yet or that the virtual circuit is being timed for inactivity.
Reassembly	Number of bytes received and held for reassembly (packets with the More bit set are reassembled into datagrams for encapsulation virtual circuits; switched X.25 traffic is not reassembled).
Held Fragments/Packets	Number of X.25 data fragments to transmit to complete an outgoing datagram, and the number of datagram packets waiting for transmission.
Bytes	Total number of bytes sent and received. The Packets, Resets, RNRs, REJs, and INTs fields show the total sent and received packet counts of the indicated types. (RNR is Receiver Not Ready, REJ is Reject, and INT is Interrupt).

1. The ITU-T carries out the functions of the former Consultative Committee for International Telegraph and Telephone (CCITT).

Table 12-10 describes the connection description fields for virtual circuits carrying encapsulation traffic.

Table 12-10 Show X25 VC Encapsulation Traffic Field Descriptions

Field	Description
170090	The X.121 address of the remote host.
ip 131.108.170.90	The higher-level protocol and address values that are mapped to the virtual circuit.
multiprotocol CUD PID	Identifies the method used for the protocol identification (PID) in the Call User Data (CUD) field. Since PVCs are not set up using a Call packet, this field is not displayed for encapsulation PVCs. The available methods are as follows: • cisco—Cisco's traditional method was used to set up a single protocol virtual circuit. • ietf—The IETF's standard RFC 1356 method was used to set up a single protocol virtual circuit. • snap—The IETF's SNAP method for IP encapsulation was used. • multiprotocol—the IETF's multiprotocol encapsulation method was used.
standard Tx data PID	Identifies the method used for protocol identification (PID) when sending datagrams. The available methods are as follows: • no—The virtual circuit is a single-protocol virtual circuit; no PID is used. • standard—The IETF's standard RFC 1356 method for identifying the protocol is used. • snap—The IETF's SNAP method for identifying IP datagrams is used.
Reverse charged	Some important virtual circuit information might be displayed as needed; a DDN IP precedence encapsulation value, reverse charged virtual circuits, and virtual circuits that allow the D-bit procedure.

The following sample display shows virtual circuits carrying locally switched X.25 traffic:

```
Router# show x25 vc
PVC 1,  State: D1,  Interface: Serial2
 Started 0:01:26, last input never, output never
 PVC <--> Serial1 PVC 1 connected
 Window size input: 2, output: 2
 Packet size input: 128, output: 128
 PS: 0  PR: 0  ACK: 0  Remote PR: 0  RCNT: 0  RNR: FALSE
 Retransmits: 0  Timer (secs): 0  Reassembly (bytes): 0
 Held Fragments/Packets: 0/0
 Bytes 0/0 Packets 0/0 Resets 0/0 RNRs 0/0 REJs 0/0 INTs 0/0

 SVC 5,  State: D1,  Interface: Serial2
  Started 0:00:16, last input 0:00:15, output 0:00:15
  Connects 170093 <--> 170090 from Serial1 VC 5
  Window size input: 2, output: 2
  Packet size input: 128, output: 128
  PS: 5  PR: 5  ACK: 4  Remote PR: 5  RCNT: 1  RNR: FALSE
  Retransmits: 0  Timer (secs): 0  Reassembly (bytes): 0
  Held Fragments/Packets: 0/0
  Bytes 505/505 Packets 5/5 Resets 0/0 RNRs 0/0 REJs 0/0 INTs 0/0
```

Table 12-11 lists the connection description fields for virtual circuits carrying locally switched X.25 traffic.

Table 12-11 Show X25 VC Local Traffic Field Descriptions

Field	Description
PVC	Flags PVC information.
Serial1 PVC 1	Identifies the other half of a local PVC connection.
connected	Identifies the state of the PVC. If the PVC is not connected, the status of the PVC will also be displayed. See Table 12-13 for PVC status messages.
170093	Identifies the Calling (source) Address of the connection. If a Calling Address Extension was encoded in the call facilities, it will also be displayed. If the source host is a CMNS host, its MAC address will also be displayed.
170090	Identifies the Called (destination) Address of the connection. If a Called Address Extension was encoded in the call facilities, it will also be displayed. If the destination host is a CMNS host, its MAC address will also be displayed.
from Serial1 VC 5	Indicates the direction of the call (“from” or “to”) and the connecting interface and virtual circuit number.

The following sample display shows virtual circuits carrying remotely switched X.25 traffic.

```
Router# show x25 vc
PVC 2,  State: D1,  Interface: Serial2
 Started 0:01:25, last input never, output never
 PVC <--> [131.108.165.92] Serial2/0 PVC 1 connected
 XOT between 131.108.165.91, 1998 and 131.108.165.92, 27801
 Window size input: 2, output: 2
 Packet size input: 128, output: 128
 PS: 0  PR: 0  ACK: 0  Remote PR: 0  RCNT: 0  RNR: FALSE
 Retransmits: 0  Timer (secs): 0  Reassembly (bytes): 0
 Held Fragments/Packets: 0/0
 Bytes 0/0 Packets 0/0 Resets 0/0 RNRs 0/0 REJs 0/0 INTs 0/0

SVC 6,  State: D1,  Interface: Serial2
 Started 0:00:04, last input 0:00:04, output 0:00:04
 Connects 170093 <--> 170090 from
 XOT between 131.108.165.91, 1998 and 131.108.165.92, 27896
 Window size input: 2, output: 2
 Packet size input: 128, output: 128
 PS: 5  PR: 5  ACK: 4  Remote PR: 5  RCNT: 1  RNR: FALSE
 Retransmits: 0  Timer (secs): 0  Reassembly (bytes): 0
 Held Fragments/Packets: 0/0
 Bytes 505/505 Packets 5/5 Resets 0/0 RNRs 0/0 REJs 0/0 INTs 0/0
```

Table 12-12 lists the connection description fields for virtual circuits carrying remotely switched X.25 traffic.

Table 12-12 Show X25 VC Remote X.25 Traffic Field Descriptions

Field	Description
PVC	Flags PVC information.
[131.108.165.92]	Indicates the IP address of the router remotely connecting the PVC.
Serial 2/0 PVC 1	Identifies the remote interface and PVC number.

Field	Description
connected	Identifies the state of the PVC. If the PVC is not connected, the status of the PVC will also be displayed. See Table 12-13 for the PVC status messages.
170093	Identifies the Calling (source) Address of the connection. If a Calling Address Extension was encoded in the call facilities, it will also be displayed.
170090	Identifies the Called (destination) Address of the connection. If a Called Address Extension was encoded in the call facilities, it will also be displayed.
from	Indicates the direction of the call (“from” or “to”).
XOT between...	Identifies the IP addresses and port numbers of the XOT connection.

Table 12-13 lists the PVC states that can be reported. These states are also reported by the **debug x25** command in PVC-SETUP packets (for remote PVCs only) as well as in the PVCBAD system error message. Some states apply only to remotely switched PVCs.

Table 12-13 X.25 PVC States

Field	Description
waiting to connect	The PVC is waiting to be processed for connecting.
dest. disconnected	The other end disconnected the PVC.
PVC/TCP connection refused	A remote PVC XOT TCP connection was tried and refused.
PVC/TCP routing error	A remote PVC XOT TCP connection routing error was reported.
PVC/TCP connect timed out	A remote PVC SOT TCP connection attempt timed out.
trying to connect via TCP	A remote PVC XOT TCP connection is established and is in the process of connecting.
awaiting PVC-SETUP reply	A remote PVC has initiated an XOT TCP connection and is waiting fro a reply to the setup message.
connected	The PVC is up.
no such dest. interface	The remote destination interface was reported to be in error by the remote router.
dest interface is not up	The target interface’s X.25 service is down.
non-X.25 dest. interface	The target interface isn’t configured for X.25.
no such dest. PVC	The targeted PVC does not exist.
dest PVC config mismatch	The targeted PVC is already connected.
mismatched flow control values	The configured flow control values do not match.
can’t support flow control values	The window sizes or packet sizes of the PVC cannot be supported by one of its two interfaces.

x25 accept-reverse

To configure the router to accept all reverse charge calls, use the **x25 accept-reverse** interface configuration command. To disable this facility, use the **no** form of this command.

x25 accept-reverse
no x25 accept-reverse

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

This command causes the interface to accept reverse charge calls by default. This behavior also can be configured on a per-peer basis using the **x25 map** interface configuration command.

Example

The following example sets acceptance of reverse charge calls:

```
interface serial 0
x25 accept-reverse
```

Related Command

x25 map

x25 address

To set the X.121 address of a particular network interface, use the **x25 address** interface configuration command.

x25 address *x.121-address*

Syntax Description

x.121-address	Variable-length X.121 address. The address is assigned by the X.25 network service provider.

Default

DDN and BFE encapsulations have a default interface address generated from the interface IP address; for proper DDN or BFE operation, this generated X.121 address should not be changed. Standard X.25 encapsulations do not have a default.

Command Mode

Interface configuration

Usage Guidelines

When connecting to a PDN, the PDN administration will assign the X.121 address that should be used. Other applications (for example, a private X.25 service), may assign arbitrary X.121 addresses as required by the network and service design. X.25 interfaces that only engage in X.25 switching do not need to assign an X.121 address.

Example

The following example sets the X.121 address for the interface:

```
interface serial 0
encapsulation x25
x25 address 00000123005
```

The address must match that assigned by the X.25 network service provider.

x25 bfe-decision

To specify how a router configured for **x25 bfe-emergency decision** will participate in emergency mode, use the **x25 bfe-decision** interface configuration command.

x25 bfe-decision {**no** | **yes** | **ask**}

Syntax Description

no Prevents the router from participating in emergency mode and from sending address translation information to the BFE device.

yes Allows the router to participate in emergency mode and to send address translation information to the BFE when the BFE enters emergency mode. The router obtains this information from the table created by the **x25 remote-red** command.

ask Configures the router to prompt the console operator to enter the **bfe** EXEC command.

Default

The router does not participate in emergency mode.

Command Mode

Interface configuration

Example

The following example configures interface Serial 0 to require an EXEC command from the administrator before it participates in emergency mode. The host IP address is 21.0.0.12, and the address of the remote BFE unit is 21.0.0.1. When the BFE enters emergency mode, the router will prompt the administrator for EXEC command **bfe enter** to direct the router to participate in emergency mode.

```
interface serial 0
x25 bfe-emergency decision
x25 remote-red 21.0.0.12 remote-black 21.0.0.1
x25 bfe-decision ask
```

Related Commands

bfe
x25 bfe-emergency
x25 remote-red

x25 bfe-emergency

To configure the circumstances under which the router participates in emergency mode, use the **x25 bfe-emergency** interface configuration command.

x25 bfe-emergency {**never** | **always** | **decision**}

Syntax Description

never Prevents the router from sending address translation information to the BFE. If it does not receive address translation information, the BFE cannot open a new connection for which it does not know the address.

always Allows the router to pass address translations to the BFE when it enters emergency mode and an address translation table has been created.

decision Directs the router to wait until it receives a diagnostic packet from the BFE device indicating that the emergency mode window is open. The window is only open when a condition exists that allows the BFE is to enter emergency mode. When the diagnostic packet is received, the router's participation in emergency mode depends on how it is configured using the **x25 bfe-decision** command.

Default

The router does not send address translation information to the BFE.

Command Mode

Interface configuration

Example

The following example configures interface Serial 0 to require an EXEC command from the administrator before it participates in emergency mode. The host IP address is 21.0.0.12, and the address of the remote BFE unit is 21.0.0.1. When the BFE enters emergency mode, the router will prompt the administrator for EXEC command **bfe enter** to direct the router to participate in emergency mode.

```
interface serial 0
x25 bfe-emergency decision
x25 remote-red 21.0.0.12 remote-black 21.0.0.1
x25 bfe-decision ask
```

Related Commands

bfe
x25 bfe-decision

x25 default

To set a default protocol, use the **x25 default** interface configuration command. To remove the default protocol specified, use the **no** form of this command.

x25 default *protocol*
no x25 default *protocol*

Syntax Description

protocol Specifies the protocol to assume; may be **ip** or **pad**.

Default

No default protocol is set.

Command Mode

Interface configuration

Usage Guidelines

This command specifies the protocol assumed by the router for incoming calls with unknown or missing Call User Data. If you do not use the **x25 default** interface configuration command, the router clears any incoming calls with unrecognized Call User Data.

Example

The following example establishes IP as the default protocol for X.25 calls:

```
interface serial 0
x25 default ip
```

Related Command

x25 map

x25 facility

To force facilities on a per-call basis for calls originated by the router (switched calls are not affected), use the **x25 facility** interface configuration command. To disable a facility, use the **no** form of this command.

x25 facility *facility-keyword value*
no x25 facility *facility-keyword value*

Syntax Description

facility-keyword User facility.

value Facility value; see Table 12-14 for a list of supported facilities and their values.

Default

No facility is sent.

Command Mode

Interface configuration

Usage Guidelines

Table 12-14 lists X.25 user facilities.

Table 12-14 X.25 User Facilities

User Facility	Description
cug *number*	Specifies a closed user group (CUG) number; CUGs 1 to 99 are allowed. CUGs can be used by a public data network to create a virtual private network within the larger network and to restrict access.
packetsize *in-size out-size*	Proposes input maximum packet size (*in-size*) and output maximum packet size (*out-size*) for flow control parameter negotiation. Both values must be one of the following values: 16, 32, 64, 128, 256, 512, 1024, 2048, or 4096.
windowsize *in-size out-size*	Proposes the packet count for input windows (*in-size*) and output windows (*out-size*) for flow control parameter negotiation. Both values must be in the range 1 to 127 and must not be greater than or equal to the value set for the **x25 modulo** command.
reverse	Specifies reverses charging on all calls originated by the interface.
throughput *in out*	Sets the requested throughput class negotiation values for input (*in*) and output (*out*) throughput across the network. Values for *in* and *out* are in bits per second (bps) and range from 75 to 64000 bps.
t**ransit-delay** *value*	Specifies a network transit delay for the duration of outgoing calls for networks that support transit delay. The transit delay value can be between 0 and 65534 milliseconds.
rpoa *name*	Specifies the name defined by the **x25 rpoa** command for a list of transit Recognized Private Operation Agencies (RPOAs) to use in outgoing Call Request packets.

Examples

The following example specifies a transit delay value in an X.25 configuration:

```
interface serial 0
x25 facility transit-delay 24000
```

The following example sets an RPOA name and then send the list via the X.25 user facilities:

```
x25 rpoa green_list 23 35 36
interface serial 0
x25 facility rpoa green_list
```

Related Command

x25 rpoa

x25 hic

To set the highest incoming-only virtual circuit number, use the **x25 hic** interface configuration command.

x25 hic *circuit-number*

Syntax Description

circuit-number	Virtual circuit number from 1 through 4095, or 0 if there is no incoming-only virtual circuit range. The default is 0.

Default

0

Command Mode

Interface configuration

Usage Guidelines

This command is applicable only if you have the X.25 switch configured for an incoming only virtual circuit range. Incoming is from the perspective of the X.25 DTE. If you do not want any outgoing calls from your DTE, configure both ends to disable the two-way range (set ltc and htc to 0) and configure an incoming-only range. Any incoming-only range must come before (that is, must be numerically less than) any two-way range. Any two-way range must come before any outgoing-only range.

Example

The following example sets a valid incoming-only virtual circuit range of 1 to 5:

```
interface serial 0
x25 lic 1
x25 hic 5
x25 ltc 6
```

Related Command

x25 lic

x25 hoc

To set the highest outgoing-only virtual circuit number, use the **x25 hoc** interface configuration command.

x25 hoc *circuit-number*

Syntax Description

circuit-number	Virtual circuit number from 1 through 4095, or 0 if there is no outgoing-only virtual circuit range. The default is 0.

Default

0

Command Mode

Interface configuration

Usage Guidelines

This command is applicable only if you have the X.25 switch configured for an outgoing only virtual circuit range. Outgoing is from the perspective of the X.25 DTE. If you do not want any incoming calls on your DTE, disable the two-way range (set ltc and htc to 0) and configure an outgoing-only range. Any outgoing-only range must come after (that is, be numerically greater than) any other range.

Example

The following example sets a valid outgoing-only virtual circuit range of 2000 to 2005:

```
interface serial 0
x25 loc 2000
x25 hoc 2005
```

Related Command

x25 loc

x25 hold-queue

To set the maximum number of packets to hold until a virtual circuit is able to transmit, use the **x25 hold-queue** interface configuration command. To remove this command from the configuration file and restore the default value, use the **no** form of this command without an argument.

x25 hold-queue *packets*
no x25 hold-queue [*packets*]

Syntax Description

packets	Number of packets. A hold queue value of 0 allows an unlimited number of packets in the hold queue. This argument is optional for the **no** form of this command. The default is 10 packets.

Default

10 packets

Command Mode

Interface configuration

Usage Guidelines

If you set the *queue-size* to 0 when using the **no x25 hold-queue** command, there will be no hold queue limit. While this will prevent drops until the router runs out of memory, it is only rarely appropriate. A virtual circuit hold queue value is determined when it is created; changing this parameter will not affect the hold queue limits of the existing virtual circuits.

Example

The following example sets the X.25 hold queue to hold 25 packets:

```
interface serial 0
x25 hold-queue 25
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

ip mtu †
x25 ips
x25 ops

x25 hold-vc-timer

To start the hold-vc-timer to prevent additional calls to a destination for a given period of time (thus preventing overruns on some X.25 switches caused by Call Request packets), use the **x25 hold-vc-timer** interface configuration command. To restore the default value for the timer, use the **no** form of this command

x25 hold-vc-timer *minutes*
no x25 hold-vc-timer

Syntax Description

minutes	Number of minutes to prevent calls from going to a previously failed destination. Incoming calls will still be accepted. The default is 0 minutes.

Default

0 minutes

Command Mode

Interface configuration

Usage Guidelines

Only Call Requests that the router originates will be held down; routed X.25 Call Requests are not affected by this parameter.

Upon receiving a Clear Request for an outstanding Call Request, the X.25 support code immediately tries another Call Request if it has more traffic to send, and this action might cause overrun problems.

The failed VC(s) may be observed with the **show x25 vc** command; they are renumbered to the illegal value 4096 and have the nonstandard state X1.

Example

The following example sets the hold-vc-timer to 3 minutes:

```
interface serial 0
x25 hold-vc-timer 3
```

x25 htc

To set the highest two-way virtual circuit number, use the **x25 htc** interface configuration command.

x25 htc *circuit-number*

Syntax Description

circuit-number	Virtual circuit number from 1 through 4095, or 0 if there is no two-way virtual circuit range. The default is 1024 for X.25 network service interfaces; 4095 for CMNS network service interfaces.

Default

1024 for X.25 network service interfaces; 4095 for CMNS network service interfaces.

Command Mode

Interface configuration

Usage Guidelines

This command is applicable if the X.25 switch is configured for a two-way virtual circuit range. Any two-way virtual circuit range must come after (that is, be numerically larger than) any incoming-only range, and must come before any outgoing-only range.

Example

The following example sets a valid two-way virtual circuit range of 5 to 25:

```
interface serial 0
x25 ltc 5
x25 htc 25
```

Related Commands

cmns enable
x25 ltc

x25 idle

To define the period of inactivity after which the router can clear a switched virtual circuit (SVC), use the **x25 idle** interface configuration command.

x25 idle *minutes*

Syntax Description

minutes	Idle period in minutes. The default is 0, which causes the router to keep the SVC open indefinitely.

Default

0 (causes the router to keep the SVC open indefinitely)

Command Mode

Interface configuration

Usage Guidelines

Both calls originated and terminated by the router are cleared; switched virtual circuits are not cleared. To clear one or all virtual circuits at once, use the privileged EXEC command **clear x25-vc**.

Example

The following example sets a 5-minute wait period before an idle circuit is cleared:

```
interface serial 2
x25 idle 5
```

Related Command

clear x25-vc

x25 ip-precedence

To enable the router to use the IP precedence value when it opens a new virtual circuit, use the **x25 ip-precedence** interface configuration command. To cause the precedence value to be ignored when opening virtual circuits, use the **no** form of this command.

x25 ip-precedence
no x25 ip-precedence

Syntax Description

This command has no arguments or keywords.

Default

The routers open one virtual circuit for all types of service.

Command Mode

Interface configuration

Usage Guidelines

This feature is only useful for DDN or BFE encapsulations, because only these methods have an IP precedence facility defined to allow the source and destination devices to both use the VC for traffic of the given IP priority.

There is a problem associated with this feature in that some hosts send nonstandard data in the IP TOS field, thus causing multiple wasteful virtual circuits to be created.

Four virtual circuits may be opened based on IP precedence to encapsulate routine, priority, immediate, and all higher precedences.

The nvc limit specified for the map or the interface default nvc limit still applies.

Example

The following example allows new IP encapsulation virtual circuits based on the IP precedence:

```
interface serial 3
x25 ip-precedence
```

x25 ips

To set the interface default maximum input packet size to match that of the network, use the **x25 ips** interface configuration command.

x25 ips *bytes*

Syntax Description

bytes	Byte count. It can be one of the following values: 16, 32, 64, 128, 256, 512, 1024, 2048, or 4096. The default is 128 bytes.

Default

128 bytes

Command Mode

Interface configuration

Usage Guidelines

X.25 network connections have a default maximum input packet size set by the network administrator. Larger packet sizes require less overhead processing. To send a packet larger than the X.25 packet size over an X.25 virtual circuit, a router must break the packet into two or more X.25 packets with the M-bit (“more data” bit) set. The receiving device collects all packets with the M-bit set and reassembles the original packet.

Note Set the **x25 ips** and **x25 ops** commands to the same value unless your network supports asymmetric input and output packet sizes.

Example

The following example sets the default maximum packet sizes to 512:

```
interface serial 1
x25 ips 512
x25 ops 512
```

Related Commands

x25 facility
x25 ops

x25 lic

To set the lowest incoming-only virtual circuit number, use the **x25 lic** interface configuration command.

x25 lic *circuit-number*

Syntax Description

circuit-number	Virtual circuit number from 1 through 4095, or 0 if there is no incoming-only virtual circuit range. The default is 0.

Default

0

Command Mode

Interface configuration

Usage Guidelines

This command is applicable only if you have the X.25 switch configured for an incoming only virtual circuit range. Outgoing is from the perspective of the X.25 DTE. If you do not want any incoming calls on your DTE, disable the two-way range (set ltc and htc to 0) and configure an outgoing-only range. Any outgoing-only range must come after (that is, be numerically greater than) any other range.

Usage Guidelines

This command is applicable if you have the X.25 switch configured for two way virtual circuit range.

Example

The following example sets a valid incoming-only virtual circuit range of 1 to 5 and sets the lowest two-way virtual circuit number:

```
interface serial 0
x25 lic 1
x25 hic 5
x25 ltc 6
```

Related Command

x25 hic

x25 linkrestart

To force X.25 Level 3 (packet-level) to restart when Level 2 (LAPB, the link level) resets, use the **x25 linkrestart** interface configuration command. To disable this function, use the **no** form of this command.

x25 linkrestart
no x25 linkrestart

Syntax Description

This command has no arguments or keywords.

Default

Forcing packet-level restarts is the default and is necessary for networks that expect this behavior.

Command Mode

Interface configuration

Example

The following example disables the link level restart:

```
interface serial 3
no x25 linkrestart
```

x25 loc

To set the lowest outgoing-only virtual circuit number, use the **x25 loc** interface configuration command.

x25 loc *circuit-number*

Syntax Description

circuit-number	Virtual circuit number from 1 through 4095, or 0 if there is no outgoing-only virtual circuit range. The default is 0.

Default

0

Command Mode

Interface configuration

Usage Guidelines

This command is applicable only if you have the X.25 switch configured for an outgoing only virtual circuit range. Outgoing is from the perspective of the X.25 DTE. If you do not want any incoming calls from your DTE, configure the loc and hoc values and set the ltc and htc values to 0.

Example

The following example sets a valid outgoing-only virtual circuit range of 2000 to 2005:

```
interface serial 0
x25 loc 2000
x25 hoc 2005
```

Related Command

x25 hoc

x25 ltc

To set the lowest two-way virtual circuit number, use the **x25 ltc** interface configuration command.

x25 ltc *circuit-number*

Syntax Description

circuit-number	Virtual circuit number from 1 through 4095, or 0 if there is no two-way virtual circuit range. The default is 1.

Default

1

Command Mode

Interface configuration

Usage Guidelines

This command is applicable if you have the X.25 switch configured for a two-way virtual circuit range. Any two-way virtual circuit range must come after (that is, be numerically larger than) any incoming-only range, and must come before any outgoing-only range.

Example

The following example sets a valid two-way virtual circuit range of 5 to 25:

```
interface serial 0
x25 ltc 5
x25 htc 25
```

Related Command

x25 htc

x25 map

To set up the LAN protocols-to-remote host mapping, use the **x25 map** interface configuration command. To retract a prior mapping, use the **no** form of this command with the appropriate network protocol(s) and X.121 address argument.

x25 map *protocol address* [*protocol2 address2*[...[*protocol9 address9*]]] *x.121-address* [*option*]
no x25 map *protocol address x.121-address*

Syntax Description

protocol	Protocol type, entered by keyword. Supported protocols are entered by keyword, as listed in Table 12-15. As many as nine protocol and address pairs can be specified in one command line.
address	Protocol address.
x.121-address	X.121 address of the remote host.
option	(Optional) Provides additional functionality or allows X.25 facilities to be specified for originated calls. Can be any of the options listed in Table 12-16.

Default

No LAN protocol-to-remote host mapping is set up.

Command Mode

Interface configuration

Usage Guidelines

Because no defined protocol can dynamically determine LAN protocol-to-remote host mappings, you must enter all of the information for each host with which the router may exchange X.25 encapsulation traffic.

Two methods are available to encapsulate traffic, Cisco's long-available encapsulation method and the IETF's standard method (defined in RFC 1356); the latter allows hosts to exchange several protocols over a single virtual circuit. Cisco's encapsulation method is the default (for backward compatibility, unless the interface configuration command specifies **ietf**.

When you configure multiprotocol maps, you can specify a maximum of nine protocol and address pairs in an **x25 map** command. However, you can specify a protocol once only. For example, you can specify the IP protocol and an IP address, but you cannot specify another IP address. If **compressedtcp** and **ip** are both specified, the same IP address must be used.

Bridging is supported only using Cisco's traditional encapsulation method. For correct operation, bridging maps must specify the **broadcast** option.

Since most datagram routing protocols rely on broadcasts or multicasts to send routing information to their neighbors, the **broadcast** keyword is needed to run such routing protocols over X.25.

Encapsulation maps might also specify that traffic between the two hosts should be compressed, thus increasing the effective bandwidth between them at the expense of memory and computation time. Each compression virtual circuit requires memory and computation resources, so compression should be used with care and monitored to maintain acceptable resource usage and overall router performance.

OSPF treats a nonbroadcast, multiaccess network such as X.25 much the same way it treats a broadcast network in that it requires selection of a designated router. In previous releases, this required manual assignment in the OSPF configuration using the **neighbor interface** router configuration command. When the **x25 map** command is included in the configuration with the broadcast, and the **ip ospf network** command (with the **broadcast** keyword) is configured, there is no need to configure any neighbors manually. OSPF will now run over the X.25 network as a broadcast network. (Refer to the **ip ospf network** interface command for more detail.)

Note The OSPF broadcast mechanism assumes that IP class D addresses are never used for regular traffic over X.25.

You can modify the options of an **x25 map** command by restating the complete set of protocols and addresses specified for the map, followed by the desired options. To delete a map command, you must also specify the complete set of protocols and addresses; the options can be omitted when deleting a map.

Once defined, a map's protocols and addresses cannot be changed; this is because the router cannot determine whether you want to add to, delete from, or modify an existing map's protocol and address specification (or simply mistyped the command). To change a map's protocol and address specification, you must delete it and create a new map.

A given protocol/address pair cannot be used in more than one map on the same interface.

Table 12-15 lists the protocols supported by X.25.

Table 12-15 Protocols Supported by X.25

Keyword	Protocol
apollo	Apollo Domain
appletalk	AppleTalk
bridge	Bridging[1]
clns	ISO Connectionless Network Service
cmns	ISO Connection-Mode Network Service[2]
compressedtcp	TCP header compression
decnet	DECnet
ip	IP
ipx	Novell IPX
qllc	SNA encapsulation in X.25[3]
vines	Banyan VINES
xns	XNS

1. Bridging traffic is supported only for Cisco's traditional encapsulation method, so a bridge map cannot specify other protocols.

2. CMNS maps implicitly define routing information so that an incoming Call will be directed to the interface and host that best match the Call's destination NSAP; CMNS maps cannot specify other protocols or any map options. Refer to the **x25 map cmns** command for details.
3. QLLC is not available for multiprotocol encapsulation.

Table 12-16 lists the map options supported by X.25.

Table 12-16 X.25 Map Options

Option	Description
compress	Specifies X.25 payload compression should be used when mapping the traffic to this host. Each virtual circuit established for compressed traffic uses a significant amount of memory (for a table of learned data patterns) and for computation (for compression and decompression of all data). It is recommended that compression be used with careful consideration to its impact on overall router performance.
method {**cisco** | **ietf** | **snap** | **multi**}	Specifies the encapsulation method. The choices are as follows: • **cisco**—Cisco's proprietary encapsulation; not available if more than one protocol is to be carried. • **ietf**—Default RFC 1356 operation: protocol identification of single-protocol virtual circuits and protocol identification within multiprotocol virtual circuits uses the standard encoding, which is compatible with RFC 877. Multiprotocol virtual circuits are used only if needed. • **snap**—RFC 1356 operation where IP is identified using SNAP rather than the standard IETF method (the standard method is compatible with RFC 877). • **multi**—Forces a map that specifies a single protocol to set up a multiprotocol virtual circuit when a call is originated; also forces a single-protocol PVC to use multiprotocol data identification methods for all datagrams sent and received.
no-incoming	Use the map only to originate calls.
no-outgoing	Do not originate calls when using the map.
idle *minutes*	Specifies an idle timeout for calls other than the interface default; 0 minutes disables the idle timeout.
reverse	Specifies reverse charging for outgoing calls.
accept-reverse	Causes the router to accept incoming reverse-charged calls. If this option is not present, the router clears reverse charged calls unless the interface accepts all reverse charged calls.
broadcast	Causes the router to direct any broadcasts sent through this interface to the specified X.121 address. This option also simplifies the configuration of OSPF; see "Usage Guidelines" for more detail.
cug *group-number*	Specifies a closed user group number (from 1 to 99) for the mapping in an outgoing call.
nvc *count*	Sets the maximum number of virtual circuits for this map/host. The default *count* is the **x25 nvc** setting of the interface. A maximum number of eight virtual circuits can be configured for each map. Compressed TCP may only use 1 virtual circuit.

Option	Description
packetsize *in-size out-size*	Proposes maximum input packet size (*in-size*) and maximum output packet size (*out-size*) for an outgoing call. Both values typically are the same and must be one of the following values: 16, 32, 64, 128, 256, 512, 1024, 2048, or 4096.
windowsize *in-size out-size*	Proposes the packet count for input window (*in-size*) and output window (*out-size*) for an outgoing call. Both values typically are the same, must be in the range 1 to 127, and must be less than the value set by the **x25 modulo** command.
throughput *in out*	Sets the requested throughput class values for input (*in*) and output (*out*) throughput across the network for an outgoing call. Values for *in* and *out* are in bits per second (bps) and range from 75 to 48000 bps.
transit-delay *milliseconds*	Specifies the transit delay value in milliseconds (0 to 65534) for an outgoing call, for networks that support transit delay.
nuid *username password*	Specifies that a network ID facility be sent in the outgoing call with the specified TACACS username and password (in a format defined by Cisco). This option should be used only when connecting to another Cisco router. The combined length of the username and password should not exceed 127 characters.
nudata *string*	Specifies the network user identification in a format determined by the network administrator (as allowed by the standards). This option is provided for connecting to non-Cisco equipment that requires an NUID facility. The string should not exceed 130 characters and must be enclosed in quotation marks (“”) if there are any spaces present.
rpoa *name*	Specifies the name defined by the **x25 rpoa** command for a list of transit RPOAs to use in outgoing Call Request packets.
passive	Specifies that the X.25 interface should send compressed outgoing TCP datagrams only if they were already compressed when they were received. This option is available only for compressed TCP maps.

Examples

The following example maps IP address 131.08.2.5 to X.121 address 000000010300. The **broadcast** keyword directs any broadcasts sent through this interface to the specified X.121 address.

```
interface serial 0
x25 map ip 131.08.2.5 000000010300 broadcast
```

The following example specifies an RPOA name to be used when originating connections:

```
x25 rpoa green_list 23 35 36
interface serial 0
x25 map ip 131.108.170.26 10 rpoa green_list
```

The following example specifies a network user identifier (NUID) facility to send on calls originated for the address map:

```
interface serial 0
x25 map IP 131.108.174.32 2 nudata “Network User ID 35”
```

Strings can be quoted, but quotation marks are not required unless embedded blanks are present.

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

ip ospf network†
show x25 map
x25 facility
x25 map bridge
x25 map cmns
x25 map compressedtcp
x25 rpoa

x25 map bridge

To configure an Internet-to-X.121 address mapping for bridging over X.25, use the **x25 map bridge** interface configuration command.

x25 map bridge *x.121-address* **broadcast** [*option*]

Syntax Description

x.121-address	The X.121 address.
broadcast	Required keyword for bridging over X.25.
option	(Optional) Services that can be added to this map; the same options as the **x25 map** command; see Table 12-16.

Default

No bridging over X.25 is configured.

Command Mode

Interface configuration

Example

The following example configures bridging of X.25 frames using a maximum of six virtual circuits:

```
interface serial 1
x25 map bridge 000000010300 broadcast nvc 6
```

Related Command

x25 map

x25 map cmns

To map NSAP addresses to either MAC-layer addresses or X.121 addresses after enabling CMNS on a nonserial interface, use the **x25 map cmns** interface configuration command. To retract a mapping, use the **no** form of this command with the appropriate address arguments.

x25 map cmns *nsap mac-address*
no x25 map cmns *nsap mac-address*

x25 map cmns *nsap* [*x.121-address*]
no x25 map cmns *nsap*[*x.121-address*]

Syntax Description

nsap	NSAP address. The NSAP can be either the actual DTE NSAP address or the prefix of the NSAP address. The NSAP prefix is sufficient for a best match to route a call.
mac-address	MAC-level address.
x.121-address	(Optional) X.121 address.

Default

No mapping is configured.

Command Mode

Interface configuration

Usage Guidelines

The address arguments specify the NSAP address-to-MAC address or NSAP address-to-X.121 address mappings. A mapping to a MAC address is only valid on a nonserial interface. A mapping to an X.121 address is only valid on a serial interface.

If a received call has a destination NSAP, the list of CMNS hosts is consulted and, if an NSAP (or NSAP preamble) match is found, the call is routed according to the best fit (which, depending on the map configuration, may be out either a CMNS or an X.25 interface). If no NSAP match is found, the call is handled according to its X.121 address for routing or acceptance as an encapsulation call.

Example

The following example switches traffic intended for any NSAP address with prefix 38.8261.17 to MAC address 0000.0C02.5F56 over interface Ethernet 0:

```
interface ethernet 0
cmns enable
x25 map cmns 38.8261.17 0000.0C02.5F56
```

Related Commands

cmns enable
x25 map

x25 map compressedtcp

To map compressed TCP traffic to an X.121 address, use the **x25 map compressedtcp** interface configuration command. To delete a TCP header compression map for the link, use the **no** form of this command.

x25 map compressedtcp *address x.121-address* [*option*]
no x25 map compressedtcp *address x.121-address*

Syntax Description

address	IP address.
x.121-address	X.121 address.
option	(Optional) The same options as those for the **x25 map** command; see Table 12-16.

Default

No mapping is configured.

Command Mode

Interface configuration

Usage Guidelines

TCP header compression is supported over X.25 links. The implementation of compressed TCP over X.25 uses a virtual circuit (VC) to pass the compressed packets. IP traffic (including standard TCP) uses separate virtual circuits. The **nvc** map option cannot be used for TCP header compression, as only one VC can carry compressed TCP header traffic to a given host.

Example

The following example establishes a map for TCP header compression on interface serial 4:

```
interface serial 4
ip tcp header-compression
x25 map compressedtcp 131.108.2.5 000000010300
```

Related Command

x25 map

x25 modulo

To set the window modulus, use the **x25 modulo** interface configuration command.

x25 modulo *modulus*

Syntax Description

modulus	Either 8 or 128. The value of the modulo parameter must agree with that of the device on the other end of the X.25 link. The default is 8.

Default

8

Command Mode

Interface configuration

Usage Guidelines

X.25 supports flow control with a sliding window sequence count. The window counter restarts at zero upon reaching the upper limit, which is called the *window modulus*. Modulo 128 operation is also referred to as extended packet sequence numbering, which allows larger packet windows.

Example

The following example sets the window modulus to 128:

```
interface serial 0
x25 modulo 128
```

Related Commands

x25 win
x25 wout
x25 facility windowsize

x25 nvc

To specify the maximum number of switched virtual circuits (SVCs) that a protocol can have open simultaneously to one host, use the **x25 nvc** interface configuration command. To increase throughput across networks, you can establish up to eight switched virtual circuits to a host/protocol.

x25 nvc *count*

Syntax Description

count	Circuit count from 1 to 8. A maximum of eight virtual circuits can be configured for each protocol/host pair. Protocols that do not tolerate out-of-order delivery, such as encapsulated TCP header compression, will only use one virtual circuit despite this value. The default is 1.

Default

1

Command Mode

Interface configuration

Usage Guidelines

When the windows and output queues of all existing connections to a host are full, a new virtual circuit will be opened to the designated circuit count. If a new connection cannot be opened, the data is dropped.

Note The *count* value specified for **x25 nvc** affects the default value for the number of SVCs. It does not affect the **nvc** option for any **x25 map** commands that are configured.

Example

The following example sets the default maximum number of switched virtual circuits that each map can open simultaneously to 4:

```
interface serial 0
x25 nvc 4
```

x25 ops

To set the interface default maximum output packet size to match that of the network, use the **x25 ops** interface configuration command.

x25 ops *bytes*

Syntax Description

bytes	Byte count that is one of the following: 16, 32, 64, 128, 256, 512, 1024, 2048, or 4096. The default is 128 bytes.

Default

128 bytes

Command Mode

Interface configuration

Usage Guidelines

X.25 networks use maximum output packet sizes set by the network administration. Larger packet sizes are better because smaller packets require more overhead processing. To send a packet larger than the X.25 packet size over an X.25 virtual circuit, a router must break the packet into two or more X.25 packets with the M-bit ("more data" bit) set. The receiving device collects all packets with the M-bit set and reassembles the original packet.

Note Set the **x25 ips** and **x25 ops** commands to the same value unless your network supports asymmetry between input and output packets.

Example

The following example sets the default maximum packet sizes to 512:

```
interface serial 1
x25 ips 512
x25 ops 512
```

Related Command

x25 ips

x25 pvc (encapsulating)

To establish an encapsulation permanent virtual circuit (PVC), use the encapsulating version of the **x25 pvc** interface configuration command. To delete the PVC, use the **no** form of this command with the appropriate channel number.

x25 pvc *circuit protocol address* [*protocol2 address2*[...[*protocol9 address9*]]] *x.121-address* [*option*]
no x25 pvc *circuit*

Syntax Description

circuit	Virtual-circuit channel number, which must be less than the virtual circuits assigned to the switched virtual circuits (SVCs).
protocol	Protocol type, entered by keyword. Supported protocols are listed in Table 12-17. As many as nine protocol and address pairs can be specified in one command line.
address	Protocol address of the host at the other end of the PVC.
x.121-address	X.121 address.
option	(Optional) Provides additional functionality or allows X.25 parameters to be specified for the PVC. Can be any of the options listed in Table 12-18.

Default

No encapsulation PVC is established. The PVC window and maximum packet sizes default to the interface default values.

Command Mode

Interface configuration

Usage Guidelines

PVCs are not supported for ISO CMNS.

You no longer need to specify a datagram protocol/address mapping before you can set up a PVC; a map is implied from the PVC configuration. Configurations generated by the router will no longer specify a map for encapsulating PVCs.

An X.121 address must be specified for the PVC, much as is done for an **x25 map** command, although the address does not appear in the PVC data exchange. When configuring a PVC to carry CLNS traffic, the X.121 address is used as the SNPA to associate the PVC with a CLNS neighbor configuration.

Table 12-17 lists supported protocols.

Table 12-17 Protocols Supported by X.25 PVCs

Keyword	Protocol
apollo	Apollo Domain
appletalk	AppleTalk
bridge	Bridging [1]
clns	OSI Connectionless Network Service
compressedtcp	TCP header compression
decnet	DECnet
ip	IP
ipx	Novell IPX
qllc	SNA encapsulation in X.25[2]
vines	Banyan VINES
xns	XNS

1. Bridging traffic is supported only for Cisco's traditional encapsulation method, so a bridge PVC cannot specify other protocols.
2. QLLC is not available for multiprotocol encapsulation.

Table 12-18 lists supported X.25 PVC options.

Table 12-18 X.25 PVC Options

Option	Description
broadcast	Causes the router to direct any broadcasts sent through this interface to this PVC. This option also simplifies the configuration of OSPF; see the "Usage Guidelines" section for more information.
method {**cisco** | **ietf** | **snap** | **multi**}	Specifies the encapsulation method. The choices are as follows: • **cisco**—Single protocol encapsulation; not available if more than one protocol is carried. • **ietf**—Default RFC 1356 operation; single-protocol encapsulation unless more than one protocol is carried, and protocol identification (when carrying more than one protocol). • **snap**—RFC 1356 operation where IP is identified (when carrying more than one protocol) using the SNAP encoding. • **multi**—Multiprotocol encapsulation used on the PVC.
packetsize *in-size out-size*	Maximum input packet size (*in-size*) and output packet size (*out-size*) for the PVC. Both values are typically the same and must be one of the following values: 16, 32, 64, 128, 256, 512, 1024, 2048, or 4096.
passive	Specifies that transmitted TCP datagrams will be compressed only if they were received compressed.This option is available only for PVCs carrying compressed TCP header traffic.
windowsize *in-size out-size*	Packet count for input window (*in-size*) and output window (*out-size*) for the PVC. Both values are typically the same, must be in the range 1 to 127, and must less than the value set for the **x25 modulo** command.

Example

The following example establishes a PVC on channel 2 to encapsulate VINES and IP with the far host:

```
interface serial 0
x25 ltc 5
x25 pvc 2 vines 60002A2D:0001 ip 131.108.170.91 11110001
```

Related Command

x25 map

x25 pvc (switched)

To configure a switched permanent virtual circuit (PVC) for a given interface, use the switched version of the **x25 pvc** interface configuration command.

x25 pvc *number1* **interface** *type number* **pvc** *number2* [*option*]

Syntax Description

number1	PVC number that will be used on the local interface (as defined by the primary interface command).
interface	Required keyword to specify an interface.
type	Remote interface type.
number	Remote interface number.
pvc	Required keyword to specify a switched PVC.
number2	PVC number that will be used on the remote interface.
option	(Optional) Adds certain features to the mapping specified; can be either option listed in Table 12-19.

Default

No switched PVC is configured. The PVC window and maximum packet sizes default to the interface default values.

Command Mode

Interface configuration

Usage Guidelines

You can configure X.25 PVCs in the X.25 switching software. This means that DTEs that require permanent circuits can be connected to the router acting as an X.25 switch and have a properly functioning connection. X.25 RESETs will be sent to indicate when the circuit comes up or goes down.

PVC circuit numbers must come before (that is, be numerically smaller than the circuit numbers allocated to any SVC range).

Table 12-19 lists the switched PVC options supported by X.25.

Table 12-19 Switched PVC Options

Option	Description
packetsize *in-size out-size*	Maximum input packet size (*in-size*) and output packet size (*out-size*) for the PVC. Both values must be one of the following values: 16, 32, 64, 128, 256, 512, 1024, 2048, or 4096.
windowsize *in-size out-size*	Packet count for input window (*in-size*) and output window (*out-size*) for the PVC. Both values should be the same, must be in the range 1 to 127, and must not be greater than the value set for the **x25 modulo** command.

Example

The following example configures a PVC connected between two serial interfaces on the same router. In this type of interconnection configuration, the alternate interface must be specified along with the PVC number on that interface. To make a working PVC connection, two commands must be specified, each pointing to the other as this example illustrates.

```
interface serial 0
encapsulation x25
x25 ltc 5
x25 pvc 1 interface serial 1 pvc 1
interface serial 1
encapsulation x25
x25 ltc 5
x25 pvc 1 interface serial 0 pvc 1
```

x25 pvc (tunnel)

To connect two permanent virtual circuits (PVCs) across a TCP/IP LAN, use the tunnel version of the **x25 pvc** interface configuration command.

x25 pvc *number1* **tunnel** *address* **interface serial** *string* **pvc** *number2* [*option*]

Syntax Description

number1	PVC number of the connecting device.
tunnel	Indicates two PVCs will be connected across a TCP/IP LAN.
address	IP address of the router to which you are connecting.
interface serial	Indicates the interface is serial.
string	Serial interface specification that accepts either a number or a string in model 7000 format (number/number) to denote the serial interface.
pvc	Indicates a PVC.
number2	Remote PVC number on the target interface.
option	(Optional) Adds certain features for the connection; can be either option listed in Table 12-20.

Default

No PVCs are connected across a TCP/IP LAN. The PVC window and packet sizes default to the interface default values.

Command Mode

Interface configuration

Usage Guidelines

Use the PVC tunnel commands to tell the router to what the far end of the PVC is connected. The incoming and outgoing packet sizes and window sizes must match the remote PVC outgoing and incoming sizes.

Table 12-20 lists the PVC tunnel options supported by X.25.

Table 12-20 X.25 PVC Tunnel Options

Option	Description
packetsize *in-size out-size*	Maximum input packet size (*in-size*) and output packet size (*out-size*) for the PVC. Both values must be one of the following values: 16, 32, 64, 128, 256, 512, 1024, 2048, or 4096.
windowsize *in-size out-size*	Packet count for input window (*in-size*) and output window (*out-size*) for the PVC. Both values should be the same, must be in the range 1 to 127, and must not be greater than or equal to the value set for the **x25 modulo** command.

Examples

The following example enters the parameters for one side of a connection destined for a router platform other than the Cisco 7000 series:

```
interface serial 0
x25 pvc 1 tunnel 131.108.1.2 interface serial 1 pvc 2
```

The following example enters the parameters for one side of a connection destined for the Cisco 7000 series:

```
interface serial 0
x25 pvc 1 tunnel 131.108.1.2 interface serial 1/1 pvc 2
```

See the section “LAPB and X.25 Configuration Examples” in the *Router Products Configuration Guide* for more complete configuration examples.

x25 remote-red

To set up the table that lists the Blacker Front End (BFE) nodes (host or gateways) to which the router will send packets, use the **x25 remote-red** interface configuration command.

x25 remote-red *host-ip-address* **remote-black** *blacker-ip-address*

Syntax Description

host-ip-address	IP address of the host or a router that the packets are being sent to.
remote-black	Delimits the addresses for the table being built.
blacker-ip-address	IP address of the remote BFE device in front of the host to which the packet is being sent.

Default

No table is set up.

Command Mode

Interface configuration

Usage Guidelines

The table that results from this command provides the address translation information the router sends to the BFE when it is in emergency mode.

Example

The following example sets up a short table of BFE nodes for interface serial 0:

```
interface serial 0
x25 remote-red 131.108.9.3 remote-black 131.108.9.13
x25 remote-red 192.108.15.1 remote-black 192.108.15.26
```

Related Commands

x25 bfe-decision
show x25 remote-red

x25 route

To create an entry in the X.25 routing table, use the **x25 route** global configuration command. To remove an entry from the table, use a **no** form of the command.

x25 route [**#***position*] *x.121-address* [**cud** *pattern*] **interface** *type number*
no x25 route [**#***position*] *x.121-address* [**cud** *pattern*] **interface** *type number*

x25 route [**#***position*] *x.121-address* [**cud** *pattern*] **ip** *ip-address* [*ip-address2 ... ip-address6*]
no x25 route [**#** *position*] *x.121-address* [**cud** *pattern*] **ip** *ip-address*

x25 route [**#***position*] *x.121-address* [**cud** *pattern*] **alias** *type number*
no x25 route [**#***position*] *x.121-address* [**cud** *pattern*] **alias** *type number*

x25 route [**#***position*] *x.121-address* [**substitute-source** *rewrite-pattern*]
[**substitute-dest** *rewrite-pattern*] [**cud** *pattern*] **interface** *type number*
no x25 route [**#***position*] *x.121-address* [**substitute-source** *rewrite-pattern*]
[**substitute-dest** *rewrite-pattern*] [**cud** *pattern*] **interface** *type number*

Note For typographical reasons, the last two commands are shown on two lines. When using the optional keywords in this variation of the **x25 route** command, the **substitute-source** keyword must precede the **substitute-dest** keyword, and both must precede the **cud** keyword. The entire command must be on one line.

Syntax Description

#*position*	(Optional) A pound sign (#) followed by a number to designate a positional parameter at which to insert the new entry. If no *position* parameter is given, the entry is appended to the end of the routing table.
x.121-address	Called X.121 address pattern. This argument can be either an actual X.121 destination address or a regular expression such as 1111*, representing a group of X.121 addresses.
cud *pattern*	(Optional) Call User Data pattern, which is specified as a printable ASCII string. The Call User Data field may be present in a call packet and is commonly 4 bytes long.
interface *type number*	Keyword and destination interface type and unit or port number; for example, interface Ethernet 0.
ip *address*	Keyword and IP address of the network interface or DTE for connections routed through a LAN. Optionally, up to five alternate IP addresses can be listed and each in turn will be tried in the event that the first destination fails, thus allowing alternate routes and decreasing the likelihood of failure.

alias *type number*	Keyword and interface type and the unit or port number of the interface alias. Encapsulation calls are normally accepted when the destination address is that of the interface (or the zero-length X.121 address). Aliases allow the specified interface to accept calls with other destination addresses.
substitute-source *rewrite-pattern*	(Optional) See Table 12-21 and Table 12-22 for summaries of pattern and character matching, respectively.
substitute-dest *rewrite-pattern*	(Optional) Specifies the called X.121 address to replace in locally routed X.25 calls. (For backwards compatibility, the **substitute** keyword will be accepted as **substitute-dest** and written to nonvolatile memory in the new format.) The backslash (\) character is treated specially in the argument *rewrite-pattern*; it indicates that the digit immediately following it selects a portion of the original called address to be inserted in the new called address. The characters \0 are replaced with the entire original address. The characters \1 through \9 are replaced with the strings that matched the first through ninth parenthesized parts of *X.121-pattern.* See Table 12-23 for a summary of pattern rewrite elements.

Default

No entry is created in the X.25 routing table.

Command Mode

Global configuration

Usage Guidelines

The X.25 routing table is consulted when an incoming call is received that should be forwarded to its destination. Two fields are used to determine the route: the called X.121 network interface address (or destination host address), and the X.25 packet's Called User Data (CUD) field. When the destination address and the CUD of the incoming packet fit the X.121 and CUD patterns in the routing table, the call is forwarded.

The order in which X.25 routing table entries are specified is significant; the list is scanned for the first match. The optional argument *# position* (# followed by a number) designates the line number at which to insert the new router. If no *position* parameter is given, the entry is appended to the end of the routing table.

The argument *X.121-address* can be either an actual X.121 destination address or a regular expression such as 1111*, representing a group of X.121 addresses.

The optional Call User Data pattern can be specified as a printable ASCII string. Both the X.121 address and Call User Data can be written using UNIX-style, regular expressions. The Call User Data field is matched against any data in the call, which is commonly 4 bytes long.

X.121 address and Call User Data are used to find a matching routing table entry. The list is scanned from the beginning to the end and each entry is pattern-matched with the incoming X.121 address and Call User Data to the X.121 and Call User Data in the routing table entry. If the pattern match for both entries succeeds, then that route is used. If the incoming call does not have any Call User

Data, then only the X.121 address pattern match need succeed with an entry that only contains an X.121 pattern. If Call User Data is present, and while scanning, a route is found that matches the X.121 address but does not have a Call User Data pattern, then that route is used when a dual match cannot be found. Regular expressions are used to allow pattern-matching operations on the X.121 addresses and Call User Data. A common operation is to do prefix matching on the X.121 DNIC field and route accordingly. For example, the pattern ^3306 will match all X.121 addresses with a DNIC of 3306. The caret (^) is a special regular expression character that anchors the match at the beginning of the pattern.

If a matching route is found, the incoming call is forwarded to the *next hop* depending on the routing entry. If no match is found, the call is cleared. If the route specifies a serial interface running X.25, the router will attempt to forward the call over that interface. If the interface is not operational the remaining routes will be checked for forwarding to an operational interface. If the interface is operational but out of available virtual circuits, the call will be cleared. Otherwise, the expected Clear Request or Call Accepted message will be forwarded back toward the originator. The “null 0” interface can be used as the destination to refuse calls to specific locations. A call cannot be forwarded out the interface it arrived on.

If the matching route specifies an IP address, a TCP connection will be established to port 1998 at the specified IP address, which must be another Cisco router. The Call Request packet will be forwarded to the remote router, where it will be processed in a similar fashion. If a routing table entry is not present or the serial interface is down or out of virtual circuits, a Clear Request will be sent back and the TCP connection will be closed. Otherwise, the call will be forwarded over the serial interface and the expected Clear Request or Call Accepted packet will be returned. Incoming calls received via TCP connections that match a routing entry specifying an IP address will be cleared. This restriction prevents Cisco routers from establishing a TCP connection to another router that would establish yet another TCP connection. A router must always connect to the remote router with the destination DTE directly attached.

See Table 12-23, Table 12-21 and Table 12-22 for summaries of pattern matching,, character matching, and pattern rewrite elements. A more complete description of the pattern-matching characters is found in the “Regular Expressions” appendix.

Note that address substitution is only performed on routes to an interface. When running X.25 over IP, address substitution can be performed on the destination IP system if the destination system is configured with the appropriate X.25 routing commands.

Use the **show x25 route** command to display the X.25 routing table. The interface routes will show up after any routes used for translation commands. Because the interface routes are expected to be less specific, they should come last. This is done automatically.

Table 12-21 Pattern Matching

Pattern	Description
*	Matches 0 or more sequences of the regular expressions.
+	Matches 1 or more sequences of the regular expressions.
?	Matches the regular expression of the null string.

Table 12-22 Character Matching

Character	Description
^	Matches the null string at the beginning of the input string.
$	Matches the null string at the end of the input string.
char	Matches *char.*
.	Matches any single character.

Table 12-23 Pattern Rewrite Elements

Pattern	Description
\0	Replaces the entire original address.
\1...9	Replaces strings that match the first through ninth parenthesized part of the X.121 address.

Examples

The following example uses regular expression pattern matching characters to match just the initial portion of the complete X.25 address:

```
x25 route ^3107 interface serial 0
```

In the following example, if a call comes in on interface serial 0 and matches any X.121-address pattern, the call will be accepted for encapsulating traffic configured for the interface using x25 map commands:

```
x25 route .* alias serial 0
```

In the following example, a call will be accepted if destined for either the VAX X.121 address or the address given in the **x25 address** interface command:

```
x25 route vax-x121-address alias serial 0
```

The following example configures alternate IP addresses for the routing entry. In the event the first address listed is not available, the next address is tried, and so on until a connection is made:

```
x25 route ^3106 ip 131.08.2.5  131.08.7.10  131.08.7.9
```

Related Command

show x25 route

x25 routing

To enable X.25 switching or tunneling, use the **x25 routing** global configuration command. To disable the forwarding of X.25 calls, used the **no** form of this command.

x25 routing [use-tcp-if-defs]
no x25 routing

Syntax Description

use-tcp-if-defs	(Optional) May be used to modify the acceptance of calls received over TCP.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

The **x25 routing** command enables local switching and remote switching (also called tunneling which routes X.25 traffic between two routers via a TCP connection). X.25 calls will not be forwarded until this command is issued.

The **use-tcp-if-defs** keyword may be needed when receiving remotely routed calls from routers using older software versions. Normally calls received over a TCP connection (remote routing reception) will have the flow control parameters (window sizes and maximum packet sizes) indicated, because proper operation of routed X.25 requires that these values match at both ends of the connection.

Some previous versions of our software, however, do not ensure that these values are present in all calls. In this case the router normally forces universally acceptable flow control values (window sizes of 2 and maximum packet sizes of 128) on the connection. Because some equipment disallows modification of the flow control values in the call confirm, the **use-tcp-if-defs** keyword will cause the router to use the default flow control values of the outgoing interface and indicate the resulting values in the call confirm. This modified behavior may allow easier migration to newer versions of the router code.

Example

The following example enables X.25 switching:

```
x25 routing
```

x25 rpoa

To specify a sequence of packet network carriers, use the **x25 rpoa** global configuration command. To remove the specified name, use the **no** form of this command.

x25 rpoa *name number*
no x25 rpoa *name*

Syntax Description

name	Recognized Private Operating Agency (RPOA), which must be unique with respect to all other RPOA names. It is used in the **x25 facility** and **x25 map** interface configuration commands.
number	A sequence of 1 or more numbers used to describe an RPOA; up to 10 numbers are accepted.

Default

No packet network carriers are specified.

Command Mode

Global configuration

Usage Guidelines

This command specifies a list of transit RPOAs to use, referenced by name.

Example

The following example sets an RPOA name and then send the list via the X.25 user facilities:

```
x25 rpoa green_list 23 35 36
interface serial 0
x25 facility rpoa green_list
x25 map ip 131.108.170.26 10 rpoa green_list
```

Related Commands

x25 facility
x25 map

x25 suppress-called-address

To omit the destination address in outgoing calls, use the **x25 suppress-called-address** interface configuration command. To reset this command to the default state, use the **no** form of this command.

x25 suppress-called-address
no x25 suppress-called-address

Syntax Description

This command has no arguments or keywords.

Default

The called address is sent.

Command Mode

Interface configuration

Usage Guidelines

This command omits the called (destination) X.121 address in Call Request packets and is required for networks that expect only subaddresses in the called address field.

Example

The following example suppresses or omits the called address in Call Request packets:

```
interface serial 0
x25 suppress-called-address
```

x25 suppress-calling-address

To omit the source address in outgoing calls, use the **x25 suppress-calling-address** interface configuration command. To reset this command to the default state, use the **no** form of this command.

x25 suppress-calling-address
no x25 suppress-calling-address

Syntax Description

This command has no arguments or keywords.

Default

The calling address is sent.

Command Mode

Interface configuration

Usage Guidelines

This command omits the calling (source) X.121 address in Call Request packets and is required for networks that expect only subaddresses in the calling address field.

Example

The following example suppresses or omits the calling address in Call Request packets:

```
interface serial 0
x25 suppress-calling-address
```

x25 t10

Use the **x25 t10** interface configuration command to set the value of the Restart Indication retransmission timer (T10) on DCE devices.

x25 t10 *seconds*

Syntax Description

seconds	Time in seconds. The default is 60 seconds.

Default

60 seconds

Command Mode

Interface configuration

Example

The following example sets the T10 timer to 30 seconds:

```
interface serial 0
x25 t10 30
```

x25 t11

To set the value of the Incoming Call timer (T11) on DCE devices, use the **x25 t11** interface configuration command.

x25 t11 *seconds*

Syntax Description

seconds	Time in seconds. The default is 180 seconds.

Default

180 seconds

Command Mode

Interface configuration

Example

The following example sets the T11 timer to 90 seconds:

```
interface serial 0
x25 t11 90
```

x25 t12

To set the value of the Reset Indication retransmission timer (T12) on DCE devices, use the **x25 t12** interface configuration command.

x25 t12 *seconds*

Syntax Description

seconds	Time in seconds. The default is 60 seconds.

Default

60 seconds

Command Mode

Interface configuration

Example

The following example sets the T12 timer to 30 seconds:

```
interface serial 0
x25 t12 30
```

x25 t13

To set the value of the Clear Indication retransmission timer (T13) on DCE devices, use the **x25 t13** interface configuration command.

x25 t13 *seconds*

Syntax Description

seconds	Time in seconds. The default is 60 seconds.

Default

60 seconds

Command Mode

Interface configuration

Example

The following example sets the T13 timer to 30 seconds:

```
interface serial 0
x25 t13 30
```

x25 t20

To set the value of the Restart Request retransmission timer (T20) on DTE devices, use the **x25 t20** interface configuration command.

x25 t20 *seconds*

Syntax Description

seconds	Time in seconds. The default is 180 seconds.

Default

180 seconds

Command Mode

Interface configuration

Example

The following example sets the T20 timer to 90 seconds:

```
interface serial 0
x25 t20 90
```

x25 t21

To set the value of the Call Request timer (T21) on DTE devices, use the **x25 t21** interface configuration command.

x25 t21 *seconds*

Syntax Description

seconds	Time in seconds. The default is 200 seconds.

Default

200 seconds

Command Mode

Interface configuration

Example

The following example sets the T21 timer to 100 seconds:

```
interface serial 0
x25 t21 100
```

x25 t22

To set the value of the Reset Request retransmission timer (T22) on DTE devices, use the **x25 t22** interface configuration command.

x25 t22 *seconds*

Syntax Description

seconds	Time in seconds. The default is 180 seconds.

Default

180 seconds

Command Mode

Interface configuration

Example

The following example sets the T22 timer to 90 seconds:

```
interface serial 0
x25 t22 90
```

x25 t23

To set the value of the Clear Request retransmission timer (T23) on DTE devices, use the **x25 t23** interface configuration command.

x25 t23 *seconds*

Syntax Description

seconds	Time in seconds. The default is 180 seconds.

Default

180 seconds

Command Mode

Interface configuration

Example

The following example sets the T23 timer to 90 seconds:

```
interface serial 0
x25 t23 90
```

x25 th

To set the data packet acknowledgment threshold, use the **x25 th** interface configuration command.

x25 th *delay-count*

Syntax Description

delay-count	Value between zero and the input window size. A value of 1 sends one Receiver Ready acknowledgment per packet. The default is 0, which disables the acknowledgment threshold.

Default

0 (which disables the acknowledgment threshold)

Command Mode

Interface configuration

Usage Guidelines

This command instructs the router to send acknowledgment packets when it is not busy sending other packets, even if the number of input packets has not reached the input window size count.

The router sends an acknowledgment packet when the number of input packets reaches the count you specify, providing there are no other packets to send. For example, if you specify a count of 1, the router will send an acknowledgment per input packet if unable to "piggyback" the acknowledgment of an outgoing data packet. This command improves line responsiveness at the expense of bandwidth.

Example

The following example sends an explicit Receiver Ready acknowledgment when it has received five data packets that it has not acknowledged:

```
interface serial 1
x25 th 5
```

Related Commands

x25 win
x25 wout

x25 use-source-address

To override the X.121 addresses of outgoing calls forwarded over a specific interface, use the **x25 use-source-address** interface configuration command. Use the **no** form of this command to prevent updating the source addresses of outgoing calls.

x25 use-source-address
no x25 use-source-address

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

Some X.25 calls, when forwarded by the X.25 switching support, need the calling (source) X.121 address updated to that of the outgoing interface. This is necessary when forwarding calls from private data networks to public data networks.

Example

The following example shows how to prevent updating the source addresses of outgoing X.25 calls on interface serial 0 once calls have been forwarded:

```
interface serial 0
no x25 use-source-address
```

x25 win

To change the default incoming window size to match that of the network, use the **x25 win** interface configuration command.

x25 win *packets*

Syntax Description

packets	Packet count that can range from 1 to one less than the window modulus.

Default

2 packets

Command Mode

Interface configuration

Usage Guidelines

This command determines the default number of packets a virtual circuit can receive before sending an X.25 acknowledgment. To maintain high bandwidth utilization, assign this limit the largest number that the network allows.

Note Set **x25 win** and **x25 wout** to the same value unless your network supports asymmetric input and output window sizes.

Example

The following example specifies that five packets may be received before sending an X.25 acknowledgment:

```
interface serial 1
x25 win 5
```

Related Commands

x25 modulo
x25 th
x25 wout

x25 wout

To change the default outgoing window size to match that of the network, use the **x25 wout** interface configuration command.

x25 wout *packets*

Syntax Description

packets	Packet count that can range from 1 to one less than the window modulus.

Default

2 packets

Command Mode

Interface configuration

Usage Guidelines

This command determines the default number of packets a virtual circuit can send before waiting for an X.25 acknowledgment. To maintain high bandwidth utilization, assign this limit the largest number that the network allows.

Note Set **x25 win** and **x25 wout** to the same value unless your network supports asymmetric input and output window sizes.

Example

The following example specifies a default limit of five for the number of outstanding unacknowledged packets for virtual circuits:

```
interface serial 1
x25 wout 5
```

Related Commands

x25 modulo
x25 th
x25 win

PART 4

Routing Protocols

CHAPTER 13

Apollo Domain Commands

The Apollo Domain routing protocol is the native-mode networking protocol for Apollo workstations. This chapter describes how to configure Apollo Domain routing. It also describes how to control access to the Apollo Domain network, optimize Apollo Domain network performance, and monitor the Apollo Domain network. For a complete description of the commands discussed in this chapter, refer to the "Configuring Apollo Domain" chapter in the *Router Products Configuration Guide*.

apollo access-group

To apply an access list to an interface, use the **apollo access-group** interface configuration command. To remove the access list, use the **no** form of this command.

apollo access-group *access-list-name*
no apollo access-group

Syntax Description

access-list-name	Name of an access list to apply to the interface.

Default

None

Command Mode

Interface configuration

Usage Guidelines

The **apollo access-group** command applies an access list to an interface. You use the **apollo access-list** command to specify the filtering conditions.

You can apply only one access list to an interface.

Example

In the following example, the access list named “eng” is assigned to the first Ethernet interface:

```
interface ethernet 0
apollo access-group eng
```

Related Commands

apollo access-list
show apollo interface

apollo access-list

To define an Apollo Domain access list, use the **access-list** global configuration command. To remove an access list, use the **no** form of this command.

apollo access-list *access-list-name* {**deny** | **permit**} [*firstnet*-]*lastnet.host* [*wildcard-mask*]
no apollo access-list *access-list-name*

Syntax

access-list-name	Name of the access list.
deny	Denies access if the conditions are matched.
permit	Permits access if the conditions are matched.
firstnet	(Optional) Number that specifies the lower limit of a selected Apollo network range.
lastnet.host	Number that specifies the upper limit of a selected Apollo network range. This is a 32-bit Apollo address consisting of a network number and a host number separated by a period. To specify all networks, use a value of –1.
wildcard-mask	(Optional) A wildcard mask that uses the one bits to ignore the host part of the network address. Host bits corresponding to wildcard mask bits set to zero are used in comparisons.

Default

None

Command Mode

Global configuration

Usage Guidelines

Use this command in conjunction with the **apollo access-group** command to restrict access to the Apollo network. Apollo Domain access lists are collections of permit and deny conditions that apply to defined Apollo network and host numbers. The router sequentially tests the network and host numbers against conditions set in the access lists. The first match determines whether the router accepts or rejects the network and host number. Because the router stops testing conditions after the first match, the order of the conditions is critical. If no conditions match, the software rejects the network and host number.

Apollo Domain access lists are identified by a name, not by a number.

You can define Apollo access lists for a single network or for a range of Apollo networks. An access list can contain an indefinite number of actual and wildcard addresses. A wildcard address has a nonzero mask and thus potentially matches more than one actual address. The software examines the actual addresses, then the wildcard addresses. The order of the wildcard addresses is important because the software stops examining access list entries once it finds a match.

After creating an access list, apply the list restrictions to specific interfaces with the **apollo access-group** command.

Example

In the following example, the first line denies access to networks 3a to 3f, the second line denies access to a specific host, and the third line permits everyone else.

```
apollo access-list eng deny 3a-3f.0 fffff
apollo access-list eng deny 5fe.1293c
apollo access-list eng permit -1.0 ffff
```

Related Commands

apollo access-group
show apollo interface

apollo maximum-paths

To set the maximum number of paths the router uses when sending packets, use the **apollo maximum-paths** global configuration command. To restore the default value, use the **no** form of this command.

apollo maximum-paths *paths*
no apollo maximum-paths

Syntax Description

paths	Maximum number of equal-cost paths from which the router chooses. The argument *paths* can be a value from 1 to 512. The default is 1.

Default

1 path

Command Mode

Global configuration

Usage Guidelines

A router can use multiple paths to reach an Apollo Domain destination in order to increase throughput in the network. By default, the router will pick one best path and send all traffic on this path, but you can configure it to remember two or more paths that have equal costs and to balance the traffic load across all the available paths. (Note that when paths have differing costs, the router chooses lower-cost routes in preference to higher-cost routes.) Packets are distributed over the multiple paths in round-robin fashion on a packet-by-packet basis. That is, the first packet is sent along the first path, the second packet along the second path, and so on. If the final path is reached before all packets are sent, the next packet is sent to the first path, the next to the second path, and so on.

Limiting the number of equal-cost paths can save memory on routers with limited memory or very large configurations. Additionally, in networks with a large number of multiple paths and systems with limited ability to cache out-of-sequence packets, performance might suffer when traffic is split between many paths.

Example

The following command sets a maximum of three equal-cost paths:

```
apollo maximum-paths 3
```

Related Command

show apollo route

apollo network

To enable Apollo Domain routing on a particular interface, use the **apollo network** interface configuration command. To disable Apollo Domain routing on an interface, use the **no** form of this command.

apollo network *number*
no apollo network *number*

Syntax Description

number	Network number. This is an eight-digit hexadecimal number consisting of the network address followed by the host address.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

You must enable Apollo routing on the router with the **apollo routing** command before issuing the **apollo network** command.

Example

The following example enables Apollo Domain routing, specifying that Apollo networks 5f and 4e are connected to two of the router's Ethernet interfaces:

```
apollo routing 23d5a
interface ethernet 0
apollo network 5f
interface ethernet 1
apollo network 4e
```

Related Commands

apollo routing
show apollo interface

apollo route

To add a static route to the Apollo Domain routing table, use the **apollo route** global configuration command. To remove a route from the routing table, use the **no** form of this command.

apollo route *destination-network network.host*
no apollo route *destination-network network.host*

Syntax Description

destination-network	Network to which you want to establish a static route. This is a 12-bit hexadecimal number. You can omit leading zeros.
network.host	Network address of the router to which to forward packets destined for *destination-network*. The argument *network* is a 12-bit hexadecimal number. You can omit leading zeros. The argument *host* is the host number of the target router. This is a 20-bit hexadecimal value.

Default

No routes are predefined in the routing table.

Command Mode

Global configuration

Usage Guidelines

Static routes always override any paths determined by metrics.

Be careful when assigning static routes. When links associated with static routes are lost, traffic may stop being forwarded even though alternative paths might be available.

Example

In the following example, all packets addressed to network 33 will be forwarded to the router at the address of 45.91ac6:

```
apollo route 33 45.91ac6
```

Related Command

show apollo route

apollo routing

To enable Apollo routing, use the **apollo routing** global configuration command. To disable Apollo routing, use the **no** form of this command.

apollo routing *host*
no apollo routing *host*

Syntax Description

host	Host number of the router. This is a five-digit hexadecimal host address that is unique across the Apollo internet.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

This command must be used in conjunction with the **apollo network command.**

Example

In the following example, Apollo Domain routing is enabled on the router whose host address is 23d5a:

```
apollo routing 23d5a
```

Related Commands

apollo network
show apollo interface

apollo update-time

To set the interval between Apollo Domain routing updates, use the **apollo update-time** interface configuration command. To restore the default value, use the **no** form of this command.

apollo update-time *interval*
no apollo update-time

Syntax Description

interval	Interval, in seconds, at which Apollo Domain routing updates are sent. The minimum interval is 10 seconds, and the maximum is 2493644 seconds. The default is 30 seconds.

Default

30 seconds

Command Mode

Interface configuration

Usage Guidelines

The **apollo update-time** command sets the routing update timer on a per-interface basis. To display the current value, use the **show apollo route** command.

Routers exchange information about routes by sending broadcast messages when they are brought up and shut down, and periodically while they are running. The **apollo update-time** command lets you modify the periodic update interval.

You can set RIP timers only in a configuration in which all routers are our routers. The timers should be the same for all routers connected to the network.

The update interval you choose affects the internal Apollo Domain timers as follows:

- Apollo Domain routes are marked invalid if no routing updates for those routes are heard within six times the value of the update interval ($6 \times$ *interval*).
- Apollo Domain routes are removed from the routing table if no routing updates are heard within eight times the value of the update interval ($8 \times$ *interval*).
- If you define an update timer for more than one interface in a router, the granularity of the update timer is determined by the lowest value defined for one of the interfaces in the router. The router "wakes up" at this granularity interval and sends out updates.

  The concept of granularity is best explained by an example. (This is illustrated in the "Example" section below.) If you have two interfaces in the router and you set the update timer on one to 20 seconds and the second to 30 seconds, the router wakes up every 20 seconds to try to send routing updates. So at time 0:00:20, the router sends an update out the first interface only, and at time 0:00:40 it sends updates out the first and second interfaces. The router does not wake up at 0:00:30 to see if it needs to send an update out the second interface. This means that routing updates are sent out the second interface at N:NN:40 and N:NN:00. That is, the interval alternates between 40 seconds and 20 seconds; it is never 30 seconds. The interval on the first interface is always 20 seconds.

Ensure that all timers are the same for all routers attached to the same network segment.

Do not use the **apollo update-time** command in a multivendor router environment.

Example

The following example sets the update timers on three interfaces in the router. The update timer granularity would be 20 seconds because this is the lowest value specified.

```
interface serial 0
apollo update-time 40
interface ethernet 0
apollo update-time 20
interface ethernet 1
apollo update-time 25
```

Related Command

show apollo interface

show apollo arp

To list the entries in the Apollo Domain ARP table, use the **show apollo arp** EXEC command.

show apollo arp

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show apollo arp** command:

```
Router# show apollo arp
Protocol  Address          Age (min)     Hardware Addr   Type      Interface
Apollo    123A.CAFE              -     0000.0c00.62e6  ARPA      Ethernet0
```

Table 13-1 describes the fields shown in the display.

Table 13-1 Show Apollo ARP Field Descriptions

Field	Description
Protocol	Protocol for which the interface has been configured. This should be Apollo.
Address	Apollo address of the interface.
Age (min)	Time, in minutes, that this entry has been in the ARP table. A hyphen indicates that this is a new entry.
Hardware Addr	MAC address of this interface.
Type	Encapsulation type.
Interface	Type and number of the interface.

show apollo interface

To display the status of the Apollo Domain interfaces configured in the router and the parameters configured on each interface, use the **show apollo interface** EXEC command.

show apollo interface [*type number*]

Syntax Description

type	(Optional) Interface type. It can be one of the following types: asynchronous, dialer, Ethernet (IEEE 802.3), loopback, null, serial, or tunnel.
number	(Optional) Interface number.

Command Mode

EXEC

Sample Display

The following is sample output from the **show apollo interface** command:

```
Router# show apollo interface ethernet0
Ethernet 0 is up, line protocol is up
    Apollo address is 123A.CAFE
    Update time is 30 seconds
    Outgoing access list is not set
```

Table 13-2 describes the fields shown in the display.

Table 13-2 Show Apollo Interface Field Descriptions

Field	Description
Ethernet 0 is ...	The interface is currently active and inserted into the network (up) or is inactive and not inserted (down).
line protocol is ...	Indicates whether the software processes that handle the line protocol believe that the interface is usable (that is, whether keepalives are successful).
Apollo address is 123A.CAFE	Address of the Apollo interface, followed by its subnet mask, if any.
Update time is 30 seconds	How often the router sends RIP updates, as configured with the **apollo update-time** command.
Outgoing access list is not set	Indicates whether an access list has been enabled with the **apollo access-list** command.

Related Commands

apollo access-group
apollo access-list
apollo update-time

show apollo route

To display the contents of the Apollo Domain routing table, use the **show apollo route** EXEC command.

show apollo route [*network*]

Syntax Description

network	(Optional) Number of the network that the route is to. This is a 12-bit hexadecimal number.

Command Mode

EXEC

Sample Display

The following is sample output from the **show apollo route** command:

```
Router# show apollo route
Codes: R - RIP derived, C - connected, S - static, 1 learned routes

Maximum allowed path(s) are/is 1
C Net 123A is directly connected, 0 uses, Ethernet0
C Net 123B is directly connected, 0 uses, Ethernet1
R Net 123C [1/0] via 123A.CAFB,  4 sec, 0 uses, Ethernet0
```

Table 13-3 describes the fields shown in the display.

Table 13-3 Show Apollo Route Field Descriptions

Field	Description
Codes:	Codes defining source of route.
R	Route learned from a RIP update.
C	Directly connected network.
S	Statically defined route via the **apollo route** command.
l learned routes	Number of routes learned from RIP updates.
Maximum allowed path(s) are/is 1	Maximum number of paths for which the router has been configured with the **apollo maximum-paths** command.
Net 123A	Apollo network number.
is directly connected	Indicates that this network is directly connected to the router.
uses	Fair estimate of the number of times a route gets used. It actually indicates the number of times the route has been selected for use prior to operations such as access list filtering.
Ethernet 0	Possible interface through which you can reach the remote network via the specified router.
[1/0]	Delay/Metric. The delay is the delay between sending routing updates. The metric is the Apollo Domain metric used in making routing decisions.
via	Address of a router that is the next hop to the remote network.

Field	Description
sec	Number of seconds since information about this network was last heard.

Related Commands

apollo maximum-paths
apollo route

show apollo traffic

To display information about the number and type of Apollo Domain packets transmitted and received by the router, use the **show apollo traffic** EXEC command.

show apollo traffic

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show apollo traffic** command output:

```
Router# show apollo traffic
Rcvd:  8 total, 0 format errors, 0 checksum errors, 0 bad hop count,
       8 local destination, 0 multicast
Bcast: 8 received, 0 sent
Sent:  16 generated, 0 forwarded
       0 encapsulation failed, 0 no route
       0 unknown
```

Table 13-4 describes the fields shown in the display.

Table 13-4 Show Apollo Traffic Field Descriptions

Field	Description
Rcvd:	Description of the Apollo Domain packets the router has received.
8 total	Total number of packets the router has received.
0 format errors	Number of bad packets discarded (for example, packets with a corrupted header).
0 checksum errors	Number of packets discarded because they contained checksum errors. This field should always have a value of 0, because Apollo Domain does not use a checksum.
0 bad hop count	Number of packets discarded because their hop count exceeded 16 (that is, the packets timed out).
8 local destination	Number of packets sent to the local broadcast address or specifically to the router.
0 multicast	Number of packets received that were addressed to multiple destinations.
Bcast:	Number of broadcast packets received and sent.
Sent:	Description of the Apollo Domain packets the router has sent.
16 generated	Number of packets the router transmitted that it generated itself.
0 forwarded	Number of packets the router transmitted that it forwarded from other sources.
0 encapsulation failed	Number of packets the router was unable to encapsulate.
0 no route	Number of times the router could not locate in the routing table a route to the destination.

Field	Description
Unknown:	Number of packets the router was unable to forward, for example, because of a misconfigured helper address or because no route was available.

CHAPTER 14

AppleTalk Commands

AppleTalk is a local-area network system that was designed and developed by Apple Computer, Inc. It can run over Ethernet, Token Ring, and FDDI networks, and over Apple's proprietary twisted-pair media access system (LocalTalk). AppleTalk specifies a protocol stack comprising several protocols that direct the flow of traffic over the network.

Apple Computer uses the name *AppleTalk* to refer to the Apple networking architecture. Apple refers to the actual transmission media used in an AppleTalk network as LocalTalk (Apple's proprietary twisted-pair transmission medium for AppleTalk), TokenTalk (AppleTalk over Token Ring), EtherTalk (AppleTalk over Ethernet), and FDDITalk (AppleTalk over Fiber Distributed Data Interface).

Use the commands in this chapter to configure and monitor AppleTalk networks. For AppleTalk configuration information and examples, refer to the "Configuring AppleTalk" chapter in the *Router Products Configuration Guide*.

access-list additional-zones

To define the default action to take for access checks that apply to zones, use the **access-list additional-zones** global configuration command.

access-list *access-list-number* {**deny** | **permit**} **additional-zones**

Syntax Description

access-list-number	Number of the access list. This is a decimal number from 600 to 699.
deny	Denies access if the conditions are matched.
permit	Permits access if the conditions are matched.

Default

No access lists are predefined.

Command Mode

Global configuration

Usage Guidelines

The **access-list additional-zones** command defines the action to take for access checks not explicitly defined with the **access-list zone** command. If you do not specify this command, the default action is to deny other access.

You apply access lists defined with the **access-list additional-zones** command to outgoing routing updates and GZL filters (using the **appletalk distribute-list out**, and **appletalk getzonelist-filter** commands). You cannot apply them to data-packet filters (using the **appletalk access-group** command) or to incoming routing update filters (using the **appletalk distribute-list in** command).

Example

The following example creates an access list based on AppleTalk zones:

```
access-list 610 deny zone Twilight
access-list 610 permit additional-zones
```

Related Commands

access-list cable-range
access-list includes
access-list network
access-list other-access
access-list within
access-list zone
appletalk access-group
appletalk distribute-list in

appletalk distribute-list out
appletalk getzonelist-filter
appletalk permit-partial-zones

access-list cable-range

To define an AppleTalk access list for a cable range (for extended networks only), use the **access-list cable-range** global configuration command. To remove an access list, use the **no** form of this command.

access-list *access-list-number* {**deny** | **permit**} **cable-range** *cable-range*
no access-list *access-list-number* [{**deny** | **permit**} **cable-range** *cable-range*]

Syntax Description

access-list-number	Number of the access list. This is a decimal number from 600 to 699.
deny	Denies access if the conditions are matched.
permit	Permits access if the conditions are matched.
cable-range	Cable range value. The argument specifies the start and end of the cable range, separated by a hyphen. These values are decimal numbers from 1 to 65279. The starting network number must be less than or equal to the ending network number.

Default

No access lists are predefined.

Command Mode

Global configuration

Usage Guidelines

When used as a routing update filter, the **access-list cable-range** command affects matching on extended networks only. The conditions defined by this access list are used only when a cable range in a routing update exactly matches that specified in the **access-list cable-range** command. The conditions are never used to match a network number (for a nonextended network).

When used as a data-packet filter, the **access-list cable-range** command affects matching on any type of network number. The conditions defined by this access list are used only when the packet's source network lies in the range defined by the access list.

You apply access lists defined with the **access-list cable-range** command to data-packet and routing-update filters (using the **appletalk access-group, appletalk distribute-list in**, and **appletalk distribute-list out**). You cannot apply them to GZL filters (using the **appletalk getzonelist-filter** command).

To delete an access list, specify the minimum number of keywords and arguments needed to delete the proper access list. For example, to delete the entire access list, use the following command:

no access-list *access-list-number*

To delete the access list for a specific network, use the following command:

no access-list *access-list-number* {**deny** | **permit**} **cable-range** *cable-range*

Priority queuing for AppleTalk operates on the destination network number, not the source network number.

Example

The following access list forwards all packets except those destined to cable range 10 to 20:

```
access-list 600 deny cable-range 10-20
access-list 600 permit other-access
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

access-list additional-zones
access-list network
access-list includes
access-list other-access
access-list within
access-list zone
appletalk access-group
appletalk distribute-list in
appletalk distribute-list out
appletalk getzonelist-filter
priority-list protocol †

access-list includes

To define an AppleTalk access list that overlaps any part of a range of network numbers or cable ranges (for both extended and nonextended networks), use the **access-list includes** global configuration command. To remove an access list, use the **no** form of this command.

access-list *access-list-number* {**deny** | **permit**} **includes** *cable-range*
no access-list *access-list-number* [{**deny** | **permit**} **includes** *cable-range*]

Syntax Description

access-list-number	Number of the access list. This is a decimal number from 600 to 699.
deny	Denies access if the conditions are matched.
permit	Permits access if the conditions are matched.
cable-range	Cable range or network number. The argument specifies the start and end of the cable range, separated by a hyphen. These values are decimal numbers from 1 to 65279. The starting network number must be less than or equal to the ending network number. To specify a network number, set the starting and ending network numbers to the same value.

Default

No access lists are predefined.

Command Mode

Global configuration

Usage Guidelines

When used as a routing update filter, the **access-list includes** command affects matching on extended and nonextended AppleTalk networks. The conditions defined by this access list are used when a cable range or network number overlaps, either partially or completely, one (or more) of those specified in the **access-list includes** command.

When used as a data-packet filter, the conditions defined by this access list are used when the packet's source network lies in the range defined in the **access-list includes** command.

You apply access lists defined with the **access-list includes** command to data-packet and routing-update filters (using the **appletalk access-group, appletalk distribute-list in**, and **appletalk distribute-list out**). You cannot apply them to GZL filters (using the **appletalk getzonelist-filter** command).

To delete an access list, specify the minimum number of keywords and arguments needed to delete the proper access list. For example, to delete the entire access list, use the following command:

no access-list *access-list-number*

To delete the access list for a specific network, use the following command:

no access-list *access-list-number* {**deny** | **permit**} **includes** *cable-range*

Priority queuing for AppleTalk operates on the destination network number, not the source network number.

Example

The following example defines an access list that permits access to any network or cable range that overlaps any part of the range 10 to 20. This means, for example, that cable ranges 13 to 16 and 17 to 25 will be permitted. This access list also permits all other ranges.

```
access-list 600 permit includes 10-20
access-list 600 permit other-access
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

access-list additional-zones
access-list cable-range
access-list network
access-list other-access
access-list within
access-list zone
appletalk access-group
appletalk distribute-list in
appletalk distribute-list out
appletalk getzonelist-filter
priority-list protocol †

access-list network

To define an AppleTalk access list for a single network number (that is, for a nonextended network), use the **access-list network** global configuration command. To remove an access list, use the **no** form of this command.

access-list *access-list-number* {**deny** | **permit**} **network** *network*
no access-list *access-list-number* [{**deny** | **permit**} **network** *network*]

Syntax Description

access-list-number	Number of the access list. This is a decimal number from 600 to 699.
deny	Denies access if the conditions are matched.
permit	Permits access if the conditions are matched.
network	AppleTalk network number.

Default

No access lists are predefined.

Command Mode

Global configuration

Usage Guidelines

When used as a routing-update filter, the **access-list network** command affects matching on nonextended networks only. The conditions defined by this access list are used only when the a nonextended number in a routing update matches a network number specified in one of the **access-list network** commands. The conditions are never used to match a cable range (for an extended network) even if the cable range has the same starting and ending number.

When used as a data-packet filter, the conditions defined by this access list are used only when the packet's source network matches the network number specified in the **access-list network** command.

You apply access lists defined with the **access-list network** command to data-packet and routing-update filters (using the **appletalk access-group, appletalk distribute-list in**, and **appletalk distribute-list out**). You cannot apply them to GZL filters (using the **appletalk getzonelist-filter** command).

In software releases before 9.0, the syntax of this command was **access-list** *access-list-number* {**deny** | **permit**} *network*. The current version of the software is still able to interpret commands in this format if it finds them in a configuration or boot file. However, it is recommended that you update the commands in your configuration or boot files to match the current syntax.

Use the **no access-list** command with the *access-list-number argument* only to remove an entire access list from the configuration. Specify the optional arguments to remove a particular clause.

To delete an access list, specify the minimum number of keywords and arguments needed to delete the proper access list. For example, to delete the entire access list, use the following command:

no access-list *access-list-number*

To delete the access list for a specific network, use the following command:

no access-list *access-list-number* {**deny** | **permit**} **network** *network*

Priority queuing for AppleTalk operates on the destination network number, not the source network number.

Example

The following example defines an access list that forwards all packets except those destined for networks 1 and 2:

```
access-list 650 deny network 1
access-list 650 deny network 2
access-list 650 permit other-access
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

access-list additional-zones
access-list cable-range
access-list includes
access-list other-access
access-list within
access-list zone
appletalk access-group
appletalk distribute-list in
appletalk distribute-list out
appletalk getzonelist-filter
priority-list protocol †

access-list other-access

To define the default action to take for access checks that apply to networks or cable ranges, use the **access-list other-access** global configuration command.

access-list *access-list-number* {**deny** | **permit**} **other-access**

Syntax Description

access-list-number	Number of the access list. This is a decimal number from 600 to 699.
deny	Denies access if the conditions are matched.
permit	Permits access if the conditions are matched.

Default

No access lists are predefined.

Command Mode

Global configuration

Usage Guidelines

The **access-list other-access** command defines the action to take for access checks not explicitly defined with an **access-list network, access-list cable-range, access-list includes,** or **access-list within** command. If you do not specify this command, the default action is to deny other access.

You apply access lists defined with the **access-list other-access** command to data-packet and routing-update filters (using the **appletalk access-group, appletalk distribute-list in**, and **appletalk distribute-list out**). You cannot apply them to GZL filters (using the **appletalk getzonelist-filter** command).

In software releases before 9.0, the syntax of this command was **access-list** *access-list-number* {**deny** | **permit**} **–1**. The current version of the software is still able to interpret commands in this format if it finds them in a configuration or boot file. However, it is recommended that you update the commands in your configuration or boot files to match the current syntax.

Priority queuing for AppleTalk operates on the destination network number, not the source network number.

Example

The following example defines an access list that forwards all packets except those destined for networks 1 and 2:

```
access-list 650 deny network 1
access-list 650 deny network 2
access-list 650 permit other-access
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

access-list additional-zones
access-list cable-range
access-list includes
access-list network
access-list within
access-list zone
appletalk access-group
appletalk distribute-list in
appletalk distribute-list out
priority-list protocol †

access-list within

To define an AppleTalk access list for an extended or a nonextended network whose network number or cable range is included entirely within the specified cable range, use the **access-list within** global configuration command. To remove this access list, use the **no** form of this command.

access-list *access-list-number* {**deny** | **permit**} **within** *cable-range*
no access-list *access-list-number* [{**deny** | **permit**} **within** *cable-range*]

Syntax Description

access-list-number	Number of the access list. This is a decimal number from 600 to 699.
deny	Denies access if the conditions are matched.
permit	Permits access if the conditions are matched.
cable-range	Cable range or network number. The argument specifies the start and end of the cable range, separated by a hyphen. These values are decimal numbers from 1 to 65279. The starting network number must be less than or equal to the ending network number. To specify a network number, set the starting and ending network numbers to the same value.

Default

No access lists are predefined.

Command Mode

Global configuration

Usage Guidelines

When used as a routing update filter, the **access-list within** command affects matching on extended and nonextended AppleTalk networks. The conditions defined by this access list are used when a cable range or network number overlaps, either partially or completely, one (or more) of those specified in the **access-list within** command.

When used as a data-packet filter, the conditions defined by this access list are used when the packet's source network lies in the range defined in the **access-list within** command.

You apply access lists defined with the **access-list within** command to data-packet and routing-update (using the **appletalk access-group, appletalk distribute-list in**, and **appletalk distribute-list out**). You cannot apply them to GZL filters (using the **appletalk getzonelist-filter** command).

To delete an access list, specify the minimum number of keywords and arguments needed to delete the proper access list. For example, to delete the entire access list, use the following command:

no access-list *access-list-number*

To delete the access list for a specific network, use the following command:

no access-list *access-list-number* {**deny** | **permit**} **within** *cable-range*

Priority queuing for AppleTalk operates on the destination network number, not the source network number.

Example

The following example defines an access list that permits access to any network or cable range that is completely included in the range 10 to 20. This means, for example, that cable range 13 to 16 will be permitted, but cable range 17 to 25 will not be. The second line of the access list permits all other packets.

```
access-list 600 permit within 10-20
access-list 600 permit other-access
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

access-list additional-zones
access-list cable-range
access-list includes
access-list network
access-list other-access
access-list zone
appletalk access-group
appletalk distribute-list in
appletalk distribute-list out
appletalk getzonelist-filter
priority-list protocol †

access-list zone

To define an AppleTalk access list that applies to a zone, use the **access-list zone** global configuration command. To remove an access list, use the **no** form of this command.

access-list *access-list-number* {**deny** | **permit**} **zone** *zone-name*
no access-list *access-list-number* [{**deny** | **permit**} **zone** *zone-name*]

Syntax Description

access-list number	Number of the access list. This is a decimal number from 600 to 699.
deny	Denies access if the conditions are matched.
permit	Permits access if the conditions are matched.
zone-name	Name of the zone. The name can include special characters from the Apple Macintosh character set. To include a special character, type a colon followed by two hexadecimal characters. For zone names with a leading space character, enter the first character as the special sequence :20.

Default

No access lists are predefined.

Command Mode

Global configuration

Usage Guidelines

You apply access lists defined with the **access-list zones** command to outgoing routing update and GZL filters (using the **appletalk distribute-list out**, and **appletalk getzonelist-filter**commands). You cannot apply them to data-packet filters (using the **appletalk access-group** command) or to incoming routing update filters (using the **appletalk distribute-list in** command).

To delete an access list, specify the minimum number of keywords and arguments needed to delete the proper access list. For example, to delete the entire access list, use the following command:

no access-list *access-list-number*

To delete the access list for a specific network, use the following command:

no access-list *access-list-number* {**deny** | **permit**} **zone** *zone-name*

Use the **access-list additional-zones** command to define the action to take for access checks not explicitly defined with the **access-list zone** command.

Example

The following example creates an access list based on AppleTalk zones:

```
access-list 610 deny zone Twilight
access-list 610 permit additional-zones
```

Related Commands

access-list additional-zones
access-list cable-range
access-list includes
access-list network
access-list other-access
access-list within
appletalk access-group
appletalk distribute-list in
appletalk distribute-list out
appletalk getzonelist-filter
appletalk permit-partial-zones

appletalk access-group

To assign an access list to an interface, use the **appletalk access-group** interface configuration command. To remove the access list use the **no** form of this command.

appletalk access-group *access-list-number*
no appletalk access-group [*access-list-number*]

Syntax Description

access-list-number	Number of the access list. This is a decimal number from 600 to 699.

Default

No access lists are predefined.

Command Mode

Interface configuration

Usage Guidelines

The **appletalk access-group** command applies data-packets filter to an interface. These filters check data packets being sent out an interface. If the packets' source network has access denied, these packets are not transmitted but rather are discarded.

Data-packet filters use access lists that define conditions for networks and cable ranges only. They ignore any zone information that may be in the access list.

When you apply a data-packet filter to an interface, you should ensure that all networks or cable ranges within a zone are governed by the same filters.

Example

The following example applies access list 601 to Ethernet interface 0:

```
access-list 601 deny cable-range 1-10
access-list 601 permit other-access
interface ethernet 0
appletalk access-group 601
```

Related Commands

access-list cable-range
access-list includes
access-list network
access-list other-access
access-list within
appletalk distribute-list in
appletalk distribute-list out

appletalk address

To enable nonextended AppleTalk routing on an interface, use the **appletalk address** interface configuration command. To disable nonextended AppleTalk routing, use the **no** form of this command.

appletalk address *network.node*
no appletalk address [*network.node*]

Syntax Description

network.node	AppleTalk network address assigned to the interface. The argument *network* is the 16-bit network number in the range 0 to 65279. The argument *node* is the 8-bit node number in the range 0 to 254. Both numbers are decimal.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

You must enable routing on the interface before assigning zone names.

Specifying an address of 0.0, or 0.*node* places the interface into *discovery mode*. When in this mode, the router attempts to determine network address information from another router on the network. You also can enable discovery mode with the **appletalk discovery** command. Discovery mode does not run over serial lines.

Example

The following example enables nonextended AppleTalk routing on Ethernet interface 0:

```
appletalk routing
interface ethernet 0
appletalk address 1.129
```

Related Commands

access-list cable-range
appletalk discovery
appletalk zone

appletalk alternate-addressing

To display network numbers in a two-octet format, use the **appletalk alternate-addressing** global configuration command. To return to displaying network numbers in the format *network.node*, use the **no** form of this command.

appletalk alternate-addressing
no appletalk alternate-addressing

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

The **appletalk alternate-addressing** command displays cable ranges in the alternate format wherever applicable. This format consists of printing the upper and lower bytes of a network number as 8-bit decimal values separated by a decimal point. For example, the cable range 511-512 would be printed as 1.255-2.0.

Example

The following example enables the display of network numbers in a two-octet format:

```
appletalk alternate-addressing
```

appletalk arp interval

To specify the time interval between the retransmission of ARP packets, use the **appletalk arp interval** global configuration command. To restore both default intervals, use the **no** form of this command.

appletalk arp [**probe** | **request**] **interval** *interval*
no appletalk arp [**probe** | **request**] **interval** *interval*

Syntax Description

probe	(Optional) Indicates that the interval specified is to be used with AARP requests that are trying to determined the address of the local router when the router is being configured. If you omit probe and request, probe is the default.
request	(Optional) Indicates that the interval specified is to be used when AARP is attempting to determine the hardware address of another node so that AARP can deliver a packet.
interval	Interval, in milliseconds, between AARP transmissions. The minimum value is 33 milliseconds. When used with the **probe** keyword, the default interval is 200 milliseconds. When used with the **request** keyword, the default interval is 1000 milliseconds.

Defaults

If you omit all keywords, probe is the default.

probe: 200 milliseconds
request: 1000 milliseconds

Command Mode

Global configuration

Usage Guidelines

The time interval you specify takes effect immediately.

Lengthening the interval between AARP transmissions permits responses from devices that respond slowly, such as printers and overloaded file servers, to be received.

AARP uses the **appletalk arp probe interval** value when obtaining the address of the local router. This is done when the router is being configured. You should not change the default value of this interval unless absolutely necessary, because this value directly modifies the AppleTalk dynamic node assignment algorithm.

AARP uses the **appletalk arp request interval** value when attempting to determine the hardware address of another node so that it can deliver a packet. You can change this interval as desired, although the default value is optimal for most sites.

The **no appletalk arp** command restores both the **probe** and **request** intervals specified in the **appletalk arp interval** and **appletalk arp retransmit-count** commands to their default values.

Example

In the following example, the AppleTalk ARP retry interval is lengthened to 2000 milliseconds:

```
appletalk arp request interval 2000
```

Related Commands

appletalk arp retransmit-count
appletalk arp-timeout
appletalk glean-packets
show appletalk globals

appletalk arp retransmit-count

To specify the number of AARP probe or request transmissions, use the **appletalk arp retransmit-count** global configuration command. To restore both default values, use the **no** form of this command.

appletalk arp [**probe** | **request**] **retransmit-count** *number*
no appletalk arp [**probe** | **request**] **retransmit-count** *number*

Syntax Description

probe	(Optional) Indicates that the number specified is to be used with AARP requests that are trying to determined the address of the local router when the router is being configured. If you omit probe and request, probe is the default.
request	(Optional) Indicates that the number specified is to be used when AARP is attempting to determine the hardware address of another node so that AARP can deliver a packet.
number	Number of AARP retransmissions that will occur. The minimum number is 1. When used with the **probe** keyword, the default value is 10 retransmissions. When used with the **request** keyword, the default value is 5 retransmissions. Specifying 0 selects the default value.

Defaults

If you omit the keyword, probe is the default.

probe: 10
request: 5

Command Mode

Global configuration

Usage Guidelines

The value you specify takes effect immediately.

Increasing the number of retransmissions permits responses from devices that respond slowly, such as printers and overloaded file servers, to be received.

AARP uses the **appletalk arp probe retransmit-count** value when obtaining the address of the local router. This is done when the router is being configured. You should not change the default value unless absolutely necessary, because this value directly modifies the AppleTalk dynamic node assignment algorithm.

AARP uses the **appletalk arp request retransmit-count** value when attempting to determine the hardware address of another node so that it can deliver a packet. You can change this interval as desired, although the default value is optimal for most sites.

The **no appletalk arp** command restores both the **probe** and **request** intervals specified in the **appletalk arp interval** and **appletalk arp retransmit-count** commands to their default values.

Example

The following example specifies an AARP retransmission count of 10 for AARP packets that are requesting the hardware address of another node on the network:

```
appletalk arp request retransmit-count 10
```

Related Commands

appletalk arp interval
appletalk arp-timeout
appletalk glean-packets
show appletalk globals

appletalk arp-timeout

To specify the interval at which entries are aged out of the ARP table, use the **appletalk arp-timeout** interface configuration command. To return to the default timeout, use the **no** form of this command.

appletalk arp-timeout *interval*
no appletalk arp-timeout [*interval*]

Syntax Description

interval	Time, in minutes, after which an entry is removed from the AppleTalk ARP table. The default is 240 minutes, or 4 hours.

Default

240 minutes (4 hours)

Command Mode

Interface configuration

Example

The following example changes the ARP timeout interval on Ethernet interface 0 to 2 hours:

```
interface ethernet 0
appletalk cable-range 2-2
appletalk arp-timeout 120
```

Related Commands

appletalk arp interval
appletalk arp retransmit-count
appletalk glean-packets

appletalk aurp tickle-time

To set the AURP last-heard-from timer value, use the **appletalk aurp tickle-time** interface configuration command. To return to the default last-heard-from timer value, use the **no** form of this command.

appletalk aurp tickle-time *seconds*
no appletalk aurp tickle-time [*seconds*]

Syntax Description

seconds	Time-out value, in seconds. This value can be a number in the range 30 to infinity. The default is 90 seconds.

Default

90 seconds

Command Mode

Interface configuration

Usage Guidelines

If the tunnel peer has not been heard from with the time specified by the least-heard-from timer value, the router sends tickle packets to check that the tunnel peer is still up.

You can use this command only on tunnel interfaces.

Example

The following example changes the AURP last-heard-from timer value on tunnel interface 0 to 120 seconds:

```
interface tunnel 0
appletalk aurp tickle-time 120
```

Related Command

show appletalk interface tunnel

appletalk aurp update-interval

To set the minimum interval between AURP routing updates, use the **appletalk aurp update-interval** global configuration command. To return to the default interval, use the **no** form of this command.

appletalk aurp update-interval *seconds*
no appletalk aurp update-interval [*seconds*]

Syntax Description

seconds	AURP routing update interval, in seconds. This interval must be a multiple of 10. The default is 30 seconds.

Default

30 seconds

Command Mode

Global configuration

Usage Guidelines

The AURP routing update interval applies only to tunnel interfaces.

Example

The following example changes the AURP routing update interval on tunnel interface 0 to 40 seconds:

```
interface tunnel 0
appletalk aurp update-interval 40
```

Related Command

show appletalk globals

appletalk cable-range

To enable an extended AppleTalk network, use the **appletalk cable-range** interface configuration command. To disable an extended AppleTalk network, use the **no** form of this command.

appletalk cable-range *cable-range* [*network.node*]
no appletalk cable-range *cable-range* [*network.node*]

Syntax Description

cable-range	Cable range value. The argument specifies the start and end of the cable range, separated by a hyphen. These values are decimal number from 0 to 65279. The starting network number must be less than or equal to the ending network number.
network.node	(Optional) Suggested AppleTalk address for the interface. The argument *network* is the 16-bit network number, and the argument *node* is the 8-bit node number. Both numbers are decimal. The suggested network number must fall within the specified range of network numbers.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

You must enable routing on the interface before assigning zone names.

Specifying a cable range value of 0-0 places the interface into *discovery mode*. When in this mode, the router attempts to determine cable range information from another router on the network. You also can enable discovery mode with the **appletalk discovery** command. Discovery mode does not run over serial lines.

Example

The following example assigns a cable range of 3 to 3 to the interface:

```
interface ethernet 0
appletalk cable-range 3-3
```

Related Commands

appletalk address
appletalk discovery
appletalk zone

appletalk checksum

To enable the generation and verification of checksums for all AppleTalk packets (except routed packets), use the **appletalk checksum** global configuration command. To disable checksum generation and verification, use the **no** form of this command.

appletalk checksum
no appletalk checksum

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Global configuration

Usage Guidelines

When the **appletalk checksum** command is enabled, the router discards incoming DDP packets when the checksum is nonzero and is incorrect, and when the router is the final destination for the packet.

You might want to disable checksum generation and verification if you have very early devices, such as LaserWriter printers, that cannot receive packets that contain checksums.

Our routers do not check checksums on routed packets, thereby eliminating the need to disable checksum to allow operation of some networking applications.

Example

The following example disables the generation and verification of checksums:

```
no appletalk checksum
```

Related Command

show appletalk globals

appletalk client-mode

To allow users to access an AppleTalk zone when dialing into an asychronous line via the router's auxiliary port, use the **appletalk client-mode** interface configuration command. To disable this function, use the **no** form of this command.

appletalk client-mode
no appletalk client-mode

Syntax Description

This command has no arguments or keywords.

Default

Client mode is disabled.

Command Mode

Interface configuration

Usage Guidelines

The **appletalk client-mode** command allows a remote client to use an asynchronous interface to access AppleTalk zones, use networked peripherals, and share files with other Macintosh users.

This commands works only an asynchronous interfaces on which PPP encapsulation is enabled. Also, you must first create an internal network for the Macintosh client using the **appletalk virtual-net** global configuration command.

An interface configured with the **appletalk client-mode** and **appletalk virtual-net** global commands does not support routing.

Example

The following example allows a user to access AppleTalk functionality on an asynchronous line using PPP:

```
interface asynchronous 1
appletalk client-mode
```

Related Commands

appletalk virtual-net
encapsulation
interface async
ppp

appletalk discovery

To place an interface into discovery mode, use the **appletalk discovery** interface configuration command. To disable discovery mode, use the **no** form of this command.

appletalk discovery
no appletalk discovery

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

If an interface is connected to a network that has at least one other operational AppleTalk router, you can dynamically configure the interface using *discovery mode*. In discovery mode, an interface acquires network address information about the attached network from an operational router and then uses this information to configure itself.

If you enable discovery mode on an interface, then when the router is starting up, that interface must acquire information to configure itself from another operational router on the attached network. If no operational router is present on the connected network, the interface will not start up.

If you do not enable discovery mode, then when the router is starting up, the interface must acquire its configuration from memory. If the stored configuration is not complete, the interface will not start up. If there is another operational router on the connected network, the router will verify the interface's stored configuration with that router. If there is any discrepancy, the interface will not start up. If there are no neighboring operational routers, the router will assume the interface's stored configuration is correct and will start up.

Once an interface is operational, it can seed the configurations of other routers on the connected network regardless of whether you have enabled discovery mode on any of the routers.

If you enable **appletalk discovery** and the interface is restarted, another operational router must still be present on the directly connected network in order for the interface to start up.

It is not advisable to have all routers on a network configured with discovery mode enabled. If all routers were to restart simultaneously (for instance, after a power failure), the network would become inaccessible until at least one router were restarted with discovery mode disabled.

You also can enable discovery mode by specifying an address of 0.0. in the **appletalk address** command or a cable range of 0-0 in the **appletalk cable-range** command.

Discovery mode is useful when you are changing a network configuration or when you are adding a router to an existing network.

Discovery mode does not run over serial lines.

Use the **no appletalk discovery** command to disable discovery mode. If the interface is not operational when you issue this command (that is, if you have not issued an **appletalk zone** command on the interface), you must configure the zone name next. If the interface is operational when you issue the **no appletalk discovery** command, you can save the current configuration (in running memory) in nonvolatile memory by issuing the **write memory** EXEC command.

Example

The following example enables discovery mode on Ethernet interface 0:

```
interface ethernet 0
appletalk discovery
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

appletalk address
appletalk cable-range
appletalk zone
show appletalk interface
write memory †

appletalk distribute-list in

To filter routing updates received from other routers over a specified interface, use the **appletalk distribute-list in** interface configuration command. To remove the routing table update filter, use the **no** form of this command.

appletalk distribute-list *access-list-number* **in**
no appletalk distribute-list [*access-list-number* **in**]

Syntax Description

access-list-number	Number of the access list. This is a decimal number from 600 to 699.

Default

No routing filters are preconfigured.

Command Mode

Interface configuration

Usage Guidelines

The **appletalk distribute-list in** command controls which networks and cable ranges in routing updates will be entered into the local routing table.

Filters for incoming routing updates use access lists that define conditions for networks and cable ranges only. They cannot use access lists that define conditions for zones. All zone information in an access list assigned to the interface with the **appletalk distribute-list in** command is ignored.

An input distribution list filters network numbers received in an incoming routing update. When AppleTalk routing updates are received on the specified interface, each network number and cable range in the update is checked against the access list. Only network numbers and cable ranges that are permitted by the access list are inserted into the router's AppleTalk routing table.

Example

The following example prevents the router from accepting routing table updates received from network 10 and on Ethernet interface 3:

```
access-list 601 deny network 10
access-list 601 permit other-access
interface ethernet 3
appletalk distribute-list 601 in
```

Related Commands

access-list cable-range
access-list includes
access-list network
access-list other-access
access-list within
appletalk distribute-list out

appletalk distribute-list out

To filter routing updates transmitted to other routers, use the **appletalk distribute-list out** interface configuration command. To remove the routing table update filter, use the **no** form of this command.

appletalk distribute-list *access-list-number* **out**
no appletalk distribute-list [*access-list-number* **out**]

Syntax Description

access-list-number	Number of the access list. This is a decimal number from 600 to 699.

Default

No routing filters are preconfigured.

Command Mode

Interface configuration

Usage Guidelines

The **appletalk distribute-list out** command controls which network numbers and cable ranges are included in routing updates and which zones the local router includes in its GetZoneList replies.

When an AppleTalk routing update is generated on the specified interface, each network number and cable range in the routing table is checked against the access list. If an undefined access list is used, all network numbers and cable ranges are added to the routing update. Otherwise, if an access list is defined, only network numbers and cable ranges that satisfy the following conditions are added to the routing update:

- The network number or cable range is not explicitly or implicitly denied.
- The network number or cable range is not a member of a zone that is explicitly or implicitly denied.
- If **appletalk permit-partial-zones** is disabled (the default), the network number or cable range is not a member of a zone that is partially obscured.

A zone is considered partially obscured when one or more network numbers or cable ranges that are members of the zone is explicitly or implicitly denied.

When a ZIP GetZoneList reply is generated, only zones that satisfy the following conditions are included:

- If **appletalk permit-partial-zones** is enabled, at least one network number or cable range that is a member of the zone is explicitly or implicitly permitted.
- If **appletalk permit-partial-zones** is disabled, all network numbers or cable ranges are explicitly or implicitly permitted.
- The zone is explicitly or implicitly permitted.

Example

The following example prevents routing updates sent on Ethernet 0 from mentioning any networks in zone Admin:

```
access-list 601 deny zone Admin
access-list 601 permit other-access
interface Ethernet 0
appletalk distribute-list 601 out
```

Related Commands

access-list additional-zones
access-list zone
appletalk distribute-list in
appletalk getzonelist-filter
appletalk permit-partial-zones

appletalk domain-group

To assign a predefined domain number to an interface, use the **appletalk domain-group** interface configuration command. To remove an interface from a domain, use the **no** form of this command.

appletalk domain-group *domain-number*
no appletalk domain-group [*domain-number*]

Syntax Description

domain-number	Number of an AppleTalk domain. It can be a decimal integer from 1 through 1000000.

Default

No domain number is assigned to the interface.

Command Mode

Interface configuration

Usage Guidelines

Before you can assign a domain number to an interface, you must create a domain with that domain number using the **appletalk domain name** global configuration command.

One or more interfaces on a router can be members of the same domain. However, a given interface can be in only one domain.

Example

The following example assigns domain group 1 to Ethernet interface 0:

```
interface ethernet 0
appletalk domain-group 1
```

Related Command

appletalk domain name
show appletalk domain

appletalk domain hop-reduction

To reduce the hop-count value in packets traveling between segments of a domains, use the **appletalk domain hop-reduction** global configuration command. To disable the reduction of hop-count values, use the **no** form of this command.

appletalk domain *domain-number* **hop-reduction**
no appletalk domain *domain-number* **hop-reduction**

Syntax Description

domain-number	Number of an AppleTalk domain. It can be a decimal integer from 1 through 1000000.

Default

Reduction of hop-count values is disabled.

Command Mode

Global configuration

Usage Guidelines

Before you can specify the **appletalk domain hop-reduction** global configuration command, you must have created a domain with that domain number using the **appletalk domain name** global configuration command.

DDP and RTMP both impose a 15-hop limit when forwarding packets. A packet ages out and is no longer forwarded when its hop count reaches 16. To overcome RTMP's 15-hop limit, the domain router represents all networks accessible to routers on its local network as one hop away. This allows routers to maintain and send routing information about networks beyond the 15-hop limit and achieve full connectivity.

When you enable hop-count reduction, delivery of packets from networks that are farther than 15 hops apart is guaranteed.

Example

The following example enables hop-count reduction for domain number 1:

```
appletalk domain 1 name Delta
appletalk domain 1 hop-reduction
```

Related Command

appletalk domain name
show appletalk domain

appletalk domain name

To create a domain and assign it a name and number, use the **appletalk domain name** global configuration command. To remove a domain, use the **no** form of this command.

appletalk domain *domain-number* **name** *domain-name*
no appletalk domain *domain-number* **name** *domain-name*

Syntax Description

domain-number	Number of an AppleTalk domain. It can be a decimal integer from 1 through 1000000.
domain-name	Name of an AppleTalk domain. The name must be unique across the AppleTalk internetwork. It can be up to 32 characters long and can include special characters from the Apple Macintosh character set. To include a special character, type a colon followed by two hexadecimal characters. For zone names with a leading space character, enter the first character as the special sequence :20.

Default

No domain is created.

Command Mode

Global configuration

Example

The following example creates domain number 1 and assigns it the name Delta:

```
appletalk domain 1 name Delta
```

Related Command

appletalk routing
show appletalk domain

appletalk domain remap-range

To remap ranges of AppleTalk network numbers or cable ranges between two segments of a domain, use the **appletalk domain remap-range** global configuration command. To disable remapping, use the **no** form of this command.

appletalk domain *domain-number* **remap-range** {**in** | **out**} *start-range-end-range*
no appletalk domain *domain-number* **remap-range** {**in** | **out**} [*start-range-end-range*]

Syntax Description

domain-number	Number of an AppleTalk domain. It can be a decimal integer from 1 through 1000000.
in	Specifies that the remapping is performed on inbound packets, that is, on packets arriving into the local interenterprise network. All network numbers or cable ranges coming from the domain are remapped into the specified range.
out	Specifies that the remapping is performed on outbound packets, that is, on packets exiting from the local interenterprise network. All network numbers or cable ranges going to the domain are remapped into the specified range.
start-range	First AppleTalk network number or beginning of cable range to remap. The number must be immediately followed by a hyphen.
end-range	Last AppleTalk network number or end of cable range to remap. The number must be immediately preceded by a hyphen.

Default

No remapping is performed.

Command Mode

Global configuration

Usage Guidelines

Before you can specify the **appletalk domain remap-range** command, you must create a domain with that domain number using the **appletalk domain name** global configuration command.

Inbound and outbound are relative to the domain router.

Ensure that the domain range you specify does not overlap any network addresses or cable ranges that already exist in the AppleTalk interenterprise network.

Each domain can have two domain mapping ranges to which to remap all incoming or outgoing network numbers or cable ranges. Incoming remapping ranges cannot overlap. However, outbound remapping ranges can overlap.

When an AppleTalk network in a domain becomes inactive, its remapped entry is removed from the remapping table. This frees the space for another network to be remapped.

If there are more remote domains than available remapping range numbers, the router displays an error message and shuts down domains.

Example

The following example remaps all network addresses and cable ranges for packets inbound from domain 1 into the address range 1000 to 1999. It also remaps packets inbound from domain 2.

```
appletalk domain 1 name Delta
appletalk domain 2 name Echo
appletalk domain 1 remap-range in 10000-10999
appletalk domain 2 remap-range in 20000-20999
```

Related Commands

appletalk domain name
show appletalk remap

appletalk eigrp-splithorizon

To configure split horizon, use the **appletalk eigrp-splithorizon** interface configuration command. To disable split horizon, use the **no** form of this command.

appletalk eigrp-splithorizon
no appletalk eigrp-splithorizon

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

Usage Guidelines

If you enable split horizon on an interface, AppleTalk Enhanced IGRP update and query packets are not sent if this interface is the next hop to that destination. This reduces the number of Enhanced IGRP packets of the network.

Split horizon blocks information about routes from being advertised by a router out any interface from which that information originated. This behavior usually optimizes communication among multiple routers, particularly when links are broken. However, with nonbroadcast networks, such as Frame Relay and SMDS, situations can arise for which this behavior is less than ideal. For these situations, you may wish to disable split horizon.

Example

The following example disables split horizon on serial interface 0:

```
interface serial 0
no appletalk eigrp-splithorizon
```

appletalk eigrp-timers

To configure the AppleTalk Enhanced IGRP hello packet interval and the route hold time, use the **appletalk eigrp-timers** interface configuration command. To return to the default values for these timers, use the **no** form of this command.

appletalk eigrp-timers *hello-interval hold-time*
no appletalk eigrp-timers *hello-interval hold-time*

Syntax Description

hello-interval	Interval between hello packets, in seconds. The default interval is 5 seconds. It can be a maximum of 30 seconds.
hold-time	Hold time, in seconds. The hold time is advertised in hello packets and indicates to neighbors the length of time they should consider the sender valid. The hold time can be in the range of 15 to 90 seconds. The default is 45 seconds.

Default

hello-interval: 5 seconds
hold-time: 45 seconds

Command Mode

Interface configuration

Usage Guidelines

If the current value for the hold time is less than two times the hello interval, the hold time is reset to three time the hello interval.

If a router does not receive a hello packet within the specified hold time, routes through the router are considered available.

Increasing the hold time delays route convergence across the network.

Note Do not adjust the hold time without advising technical support.

Example

The following example changes the hello interval to 10 seconds:

```
interface ethernet 0
appletalk eigrp-timers 10 45
```

appletalk event-logging

To log significant network events, use the **appletalk event-logging** global configuration command. To disable this function, use the **no** form of this command.

appletalk event-logging
no appletalk event-logging

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

The **appletalk event-logging** command logs a subset of messages produced by **debug appletalk** command. This includes routing changes, zone creation, port status, and address.

Example

The following example shows the use of the **appletalk event-logging** command:

```
appletalk routing
appletalk event-logging
```

Related Command

show appletalk globals

appletalk free-trade-zone

To establish a free-trade zone, use the **appletalk free-trade-zone** interface configuration command. To disable a free-trade zone, use the **no** form of this command.

appletalk free-trade-zone
no appletalk free-trade-zone

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

A free-trade zone is a part of an AppleTalk internet that is accessible by two other parts of the internet, neither of which can access the other. You might want to create a free-trade zone to allow the exchange of information between two organizations that otherwise want to keep their internets isolated from each other or that do not have physical connectivity with one another.

You apply the **appletalk free-trade-zone** command to each interface attached to the common-access network. This command has the following effect on the interface:

- All incoming RTMP updates are ignored.
- All outgoing RTMP updates contain no information.
- NBP conversion of BrRq packets to FwdReq packets is not performed.

The GZL for free-trade zone nodes will be empty.

Example

The following example establishes a free-trade zone on Ethernet interface 0:

```
interface ethernet 0
appletalk cable-range 5-5
appletalk zone FreeAccessZone
appletalk free-trade-zone
```

appletalk getzonelist-filter

To filter GetZoneList (GZL) replies, use the **appletalk getzonelist-filter** interface configuration command. To remove a filter, use the **no** form of this command.

appletalk getzonelist-filter *access-list-number*
no appletalk getzonelist-filter [*access-list-number*]

Syntax Description

access-list-number	Number of the access list. This is a decimal number from 600 to 699.

Default

No filters are preconfigured.

Command Mode

Interface configuration

Usage Guidelines

GZL filters define conditions for zones only. They cannot use access lists that define conditions for network numbers or cable ranges. All network number and cable range information in the access list assigned to an interface with the **appletalk getzonelist-filter** command is ignored.

Using a GZL filter is not a complete replacement for anonymous network numbers. In order to prevent users from seeing a zone, all routers must implement the GZL filter. If there are any routers from other vendors on the network, the GZL filter will not have a consistent effect.

The Macintosh Chooser uses ZIP GZL requests to compile a list of zones from which the user can select services. Any router on the same network as the Macintosh can respond to these requests with a GZL reply. You can create a GZL filter on the router to control which zones the router mentions in its GZL replies. This has the effect of controlling the list of zones that are displayed by the Chooser.

When defining GZL filters, you should ensure that all routers on the same internetwork filter GZL reply identically. Otherwise, the Chooser will list different zone depending upon which router responded to the request. Also, inconsistent filters can result in zones appearing and disappearing every few seconds when the user remains in the Chooser. Because of these inconsistencies, you should normally use the **appletalk getzonelist-filter** command only when all routers in the internetwork are our routers, unless the other vendors' routers have a similar feature.

Replies to GZL requests are also filtered by any**appletalk distribute-list out** filter that has been applied to the same interface. You need to specify an **appletalk getzonelist-filter** command only if you want additional filtering to be applied to GZL replies. This filter is rarely needed except to eliminate zones that do not contain user services.

Example

The following example does not include the zone Engineering in GZL replies sent out Ethernet interface 0:

```
access-list 600 deny zone Engineering
interface Ethernet 0
appletalk getzonelist-filter 600
```

Related Commands

access-list additional-zones
access-list zone
appletalk distribute-list out
appletalk permit-partial-zones

appletalk glean-packets

To derive AARP table entries from incoming packets, use the **appletalk glean-packets** interface configuration command. To disable this function, use the **no** form of this command.

appletalk glean-packets
no appletalk glean-packets

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

Usage Guidelines

The router automatically derives AARP table entries from incoming packets. This process is referred to as “gleaning.” Gleaning speeds up the process of populating the AARP table.

Our implementation of AppleTalk does not forward packets with local source and destination network addresses. This behavior does not conform with the definition of AppleTalk in Apple Computer’s *Inside AppleTalk* publication. However, this behavior is designed to prevent any possible corruption of the AARP table in any AppleTalk node that is performing MAC-address gleaning.

Example

The following example disables the building of the AARP table using information derived from incoming packets:

```
interface ethernet 0
appletalk address 33
no appletalk glean-packets
```

appletalk ignore-verify-errors

To allow a router to start functioning even if the network is misconfigured, use the **appletalk ignore-verify-errors** global configuration command. To disable this function, use the **no** form of this command.

appletalk ignore-verify-errors
no appletalk ignore-verify-errors

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Example

The following example allows a router to start functioning without verifying network misconfiguration:

```
no appletalk ignore-verify-errors 0
```

Usage Guidelines

Use this command only under the guidance of a customer engineer or other service representative. A router that starts routing in a misconfigured network will serve only to make a bad situation worse; it will not correct other misconfigured routers.

appletalk iptalk

To enable IPTalk encapsulation on an interface that already has a configured IP address, use the **appletalk iptalk** interface configuration command. To disable IPTalk encapsulation, use the **no** form of this command.

appletalk iptalk *network.node zone*
no appletalk iptalk[*network.node zone*]

Syntax Description

network.node	AppleTalk network address assigned to the interface. The argument *network* is the 16-bit network number, and the argument *node* is the 8-bit node number. Both numbers are decimal.
zone	Name of the zone for the connected AppleTalk network.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

Use the **appletalk iptalk** interface subcommand to enable IPTalk encapsulation on an interface that already has a configured IP address. This command encapsulates AppleTalk in IP packets in a manner compatible with the Columbia AppleTalk Package (CAP) IPTalk and the Kinetics IPTalk (KIP) implementations.

This command allows AppleTalk communication with UNIX hosts running older versions of CAP that do not support native AppleTalk EtherTalk encapsulations. Typically, Apple Macintosh users wishing to communicate with these servers would have their connections routed through a Kinetics FastPath router running KIP (Kinetics IP) software.

This command is provided as a migration command; newer versions of CAP provide native AppleTalk EtherTalk encapsulations, and the IPTalk encapsulation is no longer required. Our implementation of IPTalk assumes that AppleTalk is already being routed on the backbone, because there is currently no LocalTalk hardware interface for our routers.

Our implementation of IPTalk does not support manually configured AppleTalk-to-IP address mapping (atab). The address mapping provided is the same as the Kinetics IPTalk implementation when the atab facility is not enabled. This address mapping functions as follows: The IP subnet mask used on the router Ethernet interface on which IPTalk is enabled is inverted (ones complement). This result is then masked against 255 (0xFF hexadecimal). This is then masked against the low-order 8 bits of the IP address to obtain the AppleTalk node number.

Example

The following example configuration illustrates how to configure IPTalk:

```
interface Ethernet 0
ip address 131.108.1.118 255.255.255.0
appletalk address 20.129
appletalk zone Native AppleTalk
appletalk iptalk 30.0 UDPZone
```

In this configuration, the IP subnet mask would be inverted:

```
255.255.255.0 inverted yields: 0.0.0.255
```

Masked with 255 it yields 255, and masked with the low-order 8 bits of the interface IP address it yields 118.

This means that the AppleTalk address of the Ethernet 0 interface seen in the UDPZone zone is 30.118. This caveat should be noted, however: Should the host field of an IP subnet mask for an interface be more than 8 bits wide, it will be possible to obtain conflicting AppleTalk node numbers. For instance, consider a situation where the subnet mask for the Ethernet 0 interface above is 255.255.240.0, meaning that the host field is 12 bits wide.

Related Command

appletalk iptalk-baseport

appletalk iptalk-baseport

To specify the UDP port number when configuring IPTalk, use the **appletalk iptalk-baseport** global configuration command. To return to the default UDP port number, use the **no** form of this command.

appletalk iptalk-baseport *port-number*
no appletalk iptalk-baseport [*port-number*]

Syntax Description

port-number	First UDP port number in the range of UDP ports used in mapping AppleTalk well-known DDP socket numbers to UDP ports.

Default

768

Command Mode

Global configuration

Usage Guidelines

Implementations of IPTalk prior to April 1988 mapped well-known DDP socket numbers to privileged UDP ports starting at port number 768. In April 1988, the NIC assigned a range of UDP ports for the defined DDP well-known sockets starting at UDP port number 200 and assigned these ports the names at-nbp, at-rtmp, at-echo, and at-zis. Release 6 and later of the CAP program dynamically decides which port mapping to use. If there are no AppleTalk service entries in the UNIX system's */etc/services* file, CAP uses the older mapping starting at UDP port number 768.

The default UDP port mapping supported by our implementation of IPTalk is 768. If there are AppleTalk service entries in the UNIX system's */etc/services* file, you should specify the beginning of the UDP port mapping range with the **appletalk iptalk-baseport** command.

Example

The following example sets the base UDP port number to 200, which is the official NIC port number, and configures IPTalk on Ethernet interface 0:

```
appletalk routing
appletalk iptalk-baseport 200
!
interface Ethernet 0
ip address 131.108.1.118 255.255.255.0
appletalk address 20.129
appletalk zone Native AppleTalk
appletalk iptalk 30.0 UDPZone
```

Related Command

appletalk iptalk

appletalk lookup-type

To specify which NBP service types are retained in the name cache, use the **appletalk lookup-type** global configuration command. To disable the caching of services, use the **no** form of this command.

appletalk lookup-type *service-type*
no appletalk lookup-type [*service-type*]

Syntax Description

service-type	AppleTalk service types. The name of a service type can include special characters from the Apple Macintosh character set. To include a special character, type a colon followed by two hexadecimal numbers. For zone names with a leading space character, enter the first character as the special sequence :20. For a list of possible types, see Table 14-1 in the "Usage Guidelines" section.

Default

The ciscoRouter entries are retained in the name cache.

Command Mode

Global configuration

Usage Guidelines

You can issue multiple **appletalk lookup-type** commands. The router does not query the entire zone, but instead polls only the connected networks. This reduces network overhead and means that the name cache contains entries only for selected services that are in a directly connected network or zone, not for all the selected services in a network or zone.

Table 14-1 lists some AppleTalk service types.

Table 14-1 AppleTalk Service Types

Service Type[1]	Description
Services for Cisco Routers	
ciscoRouter	Active adjacent Cisco routers; this service type is initially enabled by default
IPADDRESS	Addresses of active MacIP server
IPGATEWAY	Names of active MacIP server
SNMP Agent	Active SNMP agents in Cisco routers
Services for Other Vendors' Routers	
AppleRouter	Apple internet router
FastPath	Shiva LocalTalk gateway
GatorBox	Cayman LocalTalk gateway
systemRouter	Cisco's OEM router name

Service Type[1]	Description
Workstation	Macintosh running System 7; the machine type also is defined, so it is possible to easily identify all user nodes

1. Type all entries exactly as shown. Spaces are valid. Do not use leading or trailing spaces when entering service names.

If you omit the *service-type* argument **from the no appletalk lookup-type** command, no service types except those relating to our routers are cached.

To display information that is stored in the name cache about the services being used by our routers and other vendors' routers, use the **show appletalk name-cache** command.

If a neighboring router is not our router or is running our software that is earlier than Release 9.0, it is possible the router will be unable to determine the name of the neighbor. This is normal behavior, and there is no workaround.

If AppleTalk routing is enabled, enabling SNMP will automatically enable SNMP over DDP.

Name cache entries are deleted after several interval periods expire without being refreshed. (You set the interval with the **appletalk name-lookup-interval**command.) At each interval, a single request is sent via each interface that has valid addresses.

Example

The following example caches information about GatorBox services, Apple internet routers, MacIP services, and workstations. Information about our routers is automatically cached.

```
appletalk lookup GatorBox
appletalk lookup AppleRouter
appletalk lookup IPGATEWAY
appletalk lookup Workstation
```

Related Commands

appletalk name-lookup-interval
show appletalk name-cache
show appletalk nbp

appletalk macip dynamic

To allocate IP addresses to dynamic MacIP clients, use the **appletalk macip dynamic** global configuration command. To delete a MacIP dynamic address assignment, use the **no** form of this command.

appletalk macip dynamic *ip-address* [*ip-address*] **zone** *server-zone*
no appletalk macip [**dynamic** *ip-address* [*ip-address*] **zone** *server-zone*]

Syntax Description

ip-address	IP address, in four-part dotted decimal notation. To specify a range, enter two IP addresses, which represent the first and last addresses in the range.
zone *server-zone*	Zone in which the MacIP server resides. The argument *server-zone* can include special characters from the Apple Macintosh character set. To include a special character, specify a colon followed by two hexadecimal characters. For zone names with a leading space character, enter the first character as the special sequence :20. For a list of Macintosh characters, refer to the Apple Computer, Inc. specification *Inside AppleTalk*.

Default

No IP addresses are allocated.

Command Mode

Global configuration

Usage Guidelines

Use the **appletalk macip dynamic** command when configuring MacIP.

Dynamic clients are those that accept *any* IP address assignment within the dynamic range specified.

In general, it is recommended that you do not use fragmented address ranges in configuring ranges for MacIP. However, if this is unavoidable, use the **appletalk macip dynamic** command to specify as many addresses or ranges as required and use the **appletalk macip static** command to assign a specific address or address range.

To shut down all running MacIP services, use the following command:

no appletalk macip

To delete a particular dynamic address assignment from the configuration, use the following command:

no appletalk macip dynamic *ip-address* [*ip-address*] **zone** *server-zone*

Example

The following example illustrates MacIP support for dynamically addressed MacIP clients with IP addresses in the range 131.108.1.28 to 131.108.1.44.

```
!This global statement specifies the MacIP server address and zone:
appletalk macip server 131.108.1.27 zone Engineering
!
!This global statement identifies the dynamically addressed clients:
appletalk macip dynamic 131.108.1.28 131.108.1.44 zone Engineering
!
!These statements assign the IP address and subnet mask for Ethernet interface 0:
interface ethernet 0
ip address 131.108.1.27 255.255.255.0
!
!This global statement enables AppleTalk routing on the router.
appletalk routing
!
!These statements enable AppleTalk routing on the interface and
!set the zone name for the interface
interface ethernet 0
appletalk cable-range 69-69 69.128
appletalk zone Engineering
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

appletalk macip server
appletalk macip static
ip address †
show appletalk macip-servers

appletalk macip server

To establish a MacIP server for a zone, use the **appletalk macip server** global configuration command. To shut down a MACIP server, use the **no** form of this command.

appletalk macip server *ip-address* **zone** *server-zone*
no appletalk macip [**server** *ip-address* **zone** *server-zone*]

Syntax Description

ip-address	IP address, in four-part dotted decimal notation. It is suggested that this address match the address of an existing IP interface.
zone *server-zone*	Zone in which the MacIP server resides. The argument *server-zone* can include special characters from the Apple Macintosh character set. To include a special character, specify a colon followed by two hexadecimal characters. For zone names with a leading space character, enter the first character as the special sequence :20. For a list of Macintosh characters, refer to the Apple Computer, Inc. specification *Inside AppleTalk*.

Default

No MacIP server is established.

Command Mode

Global configuration

Usage Guidelines

Use the **appletalk macip server** command when configuring MacIP.

You can configure only one MacIP server per AppleTalk zone, and the server must reside in the default zone. A server is not registered via NBP until at least one MacIP resource is configured.

You can configure multiple MacIP servers for a router, but you can assign only one MacIP server to a particular zone and only one IP interface to each MacIP server. In general, you must be able to establish an alias between the IP address you assign with the **appletalk macip server** command and an existing IP interface. For implementation simplicity, it is suggested that the address specified in this command match an existing IP interface address.

To shut down all active MacIP servers, use the following command:

no appletalk macip

To delete a specific MacIP server from the MacIP configuration, use the following command:

no appletalk macip server *ip-address* **zone** *server-zone*

Example

The following example establishes a MacIP server on Ethernet interface 0 in AppleTalk zone Engineering. It then assigns an IP address to the Ethernet interface and enables AppleTalk routing on the router and the Ethernet interface.

```
appletalk macip server 131.108.1.27 zone Engineering
ip address 131.108.1.27 255.255.255.0
appletalk routing
interface ethernet 0
appletalk cable-range 69-69 69.128
appletalk zone Engineering
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

appletalk macip dynamic
appletalk macip static
ip address †
show appletalk macip-servers

appletalk macip static

To allocate an IP address to be used by a MacIP client that has reserved a static IP address, use the **appletalk macip static** global configuration command. To delete a MacIP static address assignment, use the **no** form of this command.

appletalk macip static *ip-address* [*ip-address*] **zone** *server-zone*
no appletalk macip [**static** *ip-address* [*ip-address*] **zone** *server-zone*]

Syntax Description

ip-address	IP address, in four-part dotted decimal format.
ip-address	(Optional) To specify a range, enter two IP addresses, which represent the first and last addresses in the range.
zone *server-zone*	Zone in which the MacIP server resides. The argument *server-zone* can include special characters from the Apple Macintosh character set. To include a special character, specify a colon followed by two hexadecimal characters. For zone names with a leading space character, enter the first character as the special sequence :20. For a list of Macintosh characters, refer to Apple Computer, Inc. specification *Inside AppleTalk*.

Default

No IP address is allocated.

Command Mode

Global configuration

Usage Guidelines

Use the **appletalk macip static** command when configuring MacIP.

Static addresses are for users who require fixed addresses for IP name domain name service and for administrators who do want addresses to change so they can always know who has what IP address.

In general, it is recommended that you do not use fragmented address ranges in configuring ranges for MacIP. However, if this is unavoidable, use the **appletalk macip dynamic** command to specify as many addresses or ranges as required, and then use the **appletalk macip static** command to assign a specific address or address range.

To shut down all running MacIP services, use the following command:

no appletalk macip

To delete a particular static address assignment from the configuration, use the following command:

no appletalk macip static *ip-address* [*ip-address*] **zone** *server-zone*

Example

The following example illustrates MacIP support for MacIP clients with statically allocated IP addresses. The IP addresses range is from 131.108.1.50 to 131.108.1.66. The three nodes that have the specific addresses are 131.108.1.81, 131.108.1.92, and 131.108.1.101.

```
!This global statement specifies the MacIP server address and zone:
appletalk macip server 131.108.1.27 zone Engineering
!
!These global statements identify the statically addressed clients:
appletalk macip static 131.108.1.50 131.108.1.66 zone Engineering
appletalk macip static 131.108.1.81 zone Engineering
appletalk macip static 131.108.1.92 zone Engineering
appletalk macip static 131.108.1.101 zone Engineering
!
!These statements assign the IP address and subnet mask for Ethernet interface 0:
interface ethernet 0
ip address 131.108.1.27 255.255.255.0
!
!This global statement enables AppleTalk routing on the router.
appletalk routing
!
!These statements enable AppleTalk routing on the interface and
!set the zone name for the interface
interface ethernet 0
appletalk cable-range 69-69 69.128
appletalk zone Engineering
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

appletalk macip dynamic
appletalk macip server
ip address †
show appletalk macip-servers

appletalk name-lookup-interval

To set the interval between service pollings by the router on its AppleTalk interfaces, use the **appletalk name-lookup-interval** global configuration command. To purge the name cache and return to the default polling interval, use the **no** form of this command.

appletalk name-lookup-interval *seconds*
no appletalk name-lookup-interval [*seconds*]

Syntax Description

seconds	Interval, in seconds, between NBP lookup pollings. This can be any positive integer; there is no upper limit. It is recommended that you use an interval between 300 seconds (5 minutes) and 1200 seconds (20 minutes). The smaller the interval, the more packets are generated to handle the names. Specifying an interval of 0 purges all entries from the name cache and disables the caching of service type information that is controlled by the **appletalk lookup-type** command, including the caching of information about our routers.

Default

0, which purges all entries from the name cache and disables the caching of service type information

Command Mode

Global configuration

Usage Guidelines

The router collects name information only for entities on connected AppleTalk networks.This reduces overhead.

If you enter an interval of 0, all polling for services (except ciscoRouter) is disabled. If you reenter a nonzero value, the configuration specified by the **appletalk lookup-type** command is reinstated. You cannot disable the lookup of ciscoRouter.

Example

The following example sets the lookup interval to 20 minutes:

```
appletalk name-lookup-interval 1200
```

Related Commands

appletalk lookup-type
show appletalk name-cache

appletalk permit-partial-zones

To permit access to the other networks in a zone when access to one of those networks is denied, use the **appletalk permit-partial-zones** global command. To return to the default behavior, which is to deny access to all networks in a zone if access to one of those networks is denied, use the **no** form of this command.

appletalk permit-partial-zones
no appletalk permit-partial-zones

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

The permitting of partial zones provides IP-style access control.

When you enable the use of partial zones, the NBP protocol cannot ensure the consistency and uniqueness of name bindings.

If you enable the use of partial zones, access control behavior is compatible with that of software Release 8.3.

Example

The following example allows partial zones:

```
appletalk permit-partial-zones
```

Related Commands

access-list additional-zones
access-list zone
appletalk distribute-list out
appletalk getzonelist-filter

appletalk pre-fdditalk

To enable the recognition of pre-FDDITalk packets, use the **appletalk pre-fdditalk** global configuration command. To disable this function, use the **no** form of this command.

appletalk pre-fdditalk
no appletalk pre-fdditalk

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

Use this command to have the router recognize AppleTalk packets sent on the FDDI ring from routers running Cisco software releases prior to Release 9.0(3) or Release 9.1(2).

Example

The following example disables the recognition of pre-FDDITalk packets:

```
no appletalk pre-fdditalk
```

appletalk protocol

To specify the routing protocol to use on an interface, use the **appletalk protocol** interface configuration command. To disable a routing protocol, use the **no** form of this command.

appletalk protocol {**aurp** | **eigrp** | **rtmp**}
no appletalk protocol {**aurp** | **eigrp** | **rtmp**}

Syntax Description

aurp	Specifies that the routing protocol to use is AURP. You can enable AURP only on tunnel interfaces.
eigrp	Specifies that the routing protocol to use is Enhanced IGRP.
rtmp	Specifies that the routing protocol to use is RTMP. RTMP is enabled by default.

Default

RTMP

Command Mode

Interface configuration

Usage Guidelines

You can configure an interface to use both RTMP and Enhanced IGRP. If you do so, route information learned from Enhanced IGRP will take precedence over information learned from RTMP. The router will, however, continue to send out RTMP routing updates.

You cannot disable RTMP without first enabling AURP or Enhanced IGRP.

Enabling AURP automatically disables RTMP.

You can enable AURP only on tunnel interfaces.

Examples

The following example enables AppleTalk Enhanced IGRP on serial interface 0:

```
interface serial 0
appletalk protocol eigrp
```

The following example disables RTMP on serial interface 0:

```
interface serial 0
no appletalk protocol rtmp
```

The following example enables AURP on tunnel interface 1:

```
interface tunnel 1
appletalk protocol aurp
```

Related Command

appletalk routing

appletalk proxy-nbp

To assign a proxy network number for each zone in which there is a router that supports only nonextended AppleTalk, use the **appletalk proxy-nbp** global configuration command. To delete the proxy, use the **no** form of this command.

appletalk proxy-nbp *network-number zone-name*
no appletalk proxy-nbp [*network-number zone-name*]

Syntax Description

network-number	Network number of the proxy. It is a 16-bit decimal number and must be unique on the network. This is the network number that will be advertised by the router as if it were a real network number.
zone-name	Name of the zone that contains the routers that support only nonextended AppleTalk. The name can include special characters from the Apple Macintosh character set. To include a special character, type a colon followed by two hexadecimal characters. For zone names with a leading space character, enter the first character as the special sequence :20.

Default

No proxy network number is assigned.

Command Mode

Global configuration

Usage Guidelines

The **appletalk proxy-nbp** command provides compatibility between AppleTalk Phase 1 and AppleTalk Phase 2 networks.

Proxy routes are included in outgoing RTMP updates as if they were directly connected routes, although they are not really directly connected, since they are not associated with any interface. Whenever an NBQ BrRq for the zone in question is generated by anyone anywhere in the Internet, an NBP FwdReq is directed to any router connected to the proxy route. The Phase 2 router which is the only router directly connected converts the FwdReq to LkUps, which are understood by Phase 1 routers, and sends them to every network in the zone.

In an environment in which there are Phase 1 and Phase 2 networks, you must specify at least one **appletalk proxy-nbp** command for each zone that has a nonextended-only AppleTalk router.

The proxy network number you assign with the **appletalk proxy-nbp** command cannot also be assigned to a router, nor can it also be associated with a physical network.

You need to assign only one proxy network number for each zone. However, you can define additional proxies with different network numbers to provide redundancy. Each proxy generates one or more packets for each forward request it receives. All other packets sent to the proxy network address are discarded. Defining redundant proxy network numbers increases the NBP traffic linearly.

Example

The following example defines network number 60 as an NBP proxy for the zone Twilight:

```
appletalk proxy-nbp 60 Twilight
```

Related Command

show appletalk route

appletalk require-route-zones

To prevent the advertisement of routes (network numbers or cable ranges) that have no assigned zone, use the **appletalk require-route-zones** global configuration command. To disable this option and allow the router to advertise to its neighbors routes that have no network-zone association, use the **no** form of this command.

appletalk require-route-zones
no appletalk require-route-zones

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Global configuration

Usage Guidelines

The **appletalk require-route-zones** command ensures that all networks have zone names prior to advertisement to neighbors.

The **no appletalk require-route-zones** command enables router behavior compatible with software Release 8.3.

Using this command helps prevent ZIP protocol storms. ZIP protocol storms can arise when corrupt routes are propagated and routers broadcast ZIP requests to determine the network/zone associations.

When the **appletalk require-route-zones** command is enabled, the router will not advertise a route to its neighboring routers until it has obtained the network/zone associations. This effectively limits the storms to a single network rather than the entire internet.

As an alternative to disabling this option, use the **appletalk getzonelist-filter** interface configuration command to filter *empty* zones from the list presented to users.

You can configure different zone lists on different interfaces. However, you are discouraged from doing this because AppleTalk users expect to have the same user zone lists at any end node in the internet.

The filtering provided by the **appletalk require-route-zones** command does not prevent explicit access via programmatic methods, but should be considered a user optimization to suppress unused zones. You should use other forms of AppleTalk access control lists to actually *secure* a zone or network.

Example

The following example configures a router to prevent the advertisement of routes that have no assigned zone:

```
appletalk require-route-zones
```

appletalk route-cache

To enable fast switching on all supported interfaces, use the **appletalk route-cache** interface configuration command. To disable fast switching, use the **no** form of this command.

appletalk route-cache
no appletalk route-cache

Syntax Description

This command has no arguments or keywords.

Default

Enabled on all interfaces that support fast switching

Command Mode

Interface configuration

Usage Guidelines

Fast switching allows higher throughput by switching a packet using a cache created by previous packets. Fast switching is enabled by default on all interfaces that support fast switching, including Token Ring, Frame Relay, PPP, and ATM. Note that fast switching is not supported over X.25 and LAPB encapsulations, or on the CSC-R16, CSC-1R, or CSC-2R STR Token Ring adapters.

Packet transfer performance is generally better when fast switching is enabled. However, you may want to disable fast switching in order to save memory space on interface cards and to help avoid congestion when high-bandwidth interfaces are writing large amounts of information to low-bandwidth interfaces.

For serial lines, fast switching is supported on extended serial lines with HDLC encapsulation only. It is not supported on nonextended serial lines.

Example

The following example disables fast switching on an interface:

```
interface ethernet 0
appletalk cable-range 10-20
appletalk zone Twilight
no appletalk route-cache
```

Related Command

show appletalk cache

appletalk route-redistribution

To redistribute RTMP routes into AppleTalk Enhanced IGRP and vice versa, use the **appletalk route-redistribution** global configuration command. To keep Enhanced IGRP and RTMP routes separate, use the **no** form of this command.

appletalk route-redistribution
no appletalk route-redistribution

Syntax Description

This command has no arguments or keywords.

Default

Enabled when Enhanced IGRP is enabled

Command Mode

Global configuration

Usage Guidelines

Redistribution allows routing information generated by one protocol to be advertised in another.

In the automatic redistribution of routes between Enhanced IGRP and RTMP, an RTMP hop is treated as having a slightly worse metric than an equivalent Enhanced IGRP hop on a 9.6-kilobit link. This allows Enhanced IGRP to be preferred over RTMP except in the most extreme of circumstances. Typically, you will see this only when using tunnels. If you want an Enhanced IGRP path in a tunnel to be preferred over an alternate RTMP path, you should set the interface delay and bandwidth parameters on the tunnel to bring the metric of the tunnel down to being better than a 9.6-kilobit link.

Example

In the following example, RTMP routing information is not redistributed:

```
appletalk routing eigrp 23
no appletalk route-redistribution
```

appletalk routing

To enable AppleTalk routing, use the **appletalk routing** global configuration command. To disable AppleTalk routing, use the **no** form of this command.

appletalk routing [**eigrp** *router-number*]
no appletalk routing [**eigrp** *router-number*]

Syntax Description

eigrp *router-number*	(Optional) Specifies the Enhanced IGRP routing protocol. The argument *router-number* is the router ID. It can be a decimal integer from 1 to 65535. It must be unique in your AppleTalk Enhanced IGRP internetwork.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

If you do not specify the optional keywork and argument, this command enables AppleTalk routing using the RTMP routing protocol.

You can configure multiple AppleTalk Enhanced IGRP processes on a router. To do so, assign each a different router ID number. (Note that IP and IPX Enhanced IGRP use an autonomous system number to enable Enhanced IGRP, while AppleTalk Enhanced IGRP uses a router ID.)

If you configure a router with a router number that is the same as that of a neighboring router, the router will refuse to start AppleTalk Enhanced IGRP on interfaces that connect with that neighboring router.

Examples

The following example enables AppleTalk protocol processing on the router:

```
appletalk routing
```

The following example enables AppleTalk Enhanced IGRP routing on router number 22:

```
appletalk routing eigrp 22
```

Related Commands

appletalk address
appletalk cable-range
appletalk protocol
appletalk zone

appletalk send-rtmps

To allow a router to send routing updates to its neighbors, use the **appletalk send-rtmps** interface configuration command. To block updates from being sent, use the **no** form of this command.

appletalk send-rtmps
no appletalk send-rtmps

Syntax Description

This command has no arguments or keywords.

Default

Send routing updates.

Command Mode

Interface configuration

Usage Guidelines

If you block the sending of routing updates, an interface on the network that has AppleTalk enabled is not “visible” to other routers on the network.

Example

The following example prevents a router from sending routing updates to its neighbors:

```
no appletalk send-rtmps
```

Related Commands

appletalk require-route-zones
appletalk strict-rtmp-checking
appletalk timers

appletalk static cable-range

To define a static route on an extended network, use the **appletalk static cable-range** global configuration command. To remove a static route, use the **no** form of this command.

appletalk static cable-range *cable-range* **to** *network.node* **zone** *zone-name*
no appletalk static cable-range *cable-range* **to** *network.node* [**zone** *zone-name*]

Syntax Description

cable-range	Cable range value. The argument specifies the start and end of the cable range, separated by a hyphen. These values are decimal number from 0 to 65279. The starting network number must be less than or equal to the ending network number.
to *network.node*	AppleTalk network address of the remote router. The argument *network* is the 16-bit network number in the range 0 to 65279. The argument *node* is the 8-bit node number in the range 0 to 254. Both numbers are decimal.
zone *zone-name*	Name of the zone on the remote network. The name can include special characters from the Apple Macintosh character set. To include a special character, type a colon followed by two hexadecimal characters. For zone names with a leading space character, enter the first character as the special sequence :20.

Default

No static routes are defined.

Command Mode

Global configuration

Usage Guidelines

You cannot delete a particular zone from the zone list without first deleting the static route.

Example

The following example creates a static route to the remote router whose address is 1.2 on the remote network 100-110 that is in the remote zone Remote:

```
appletalk static cable 100-110 to 1.2 zone Remote
```

Related Commands

appletalk static network
show appletalk route
show appletalk static

appletalk static network

To define a static route on a nonextended network, use the **appletalk static network** global configuration command. To remove a static route, use the **no** form of this command.

appletalk static network *network-number* **to** *network.node* **zone** *zone-name*
no appletalk static network *network-number* **to** *network.node* [**zone** *zone-name]*

Syntax Description

network-number	AppleTalk network number assigned to the interface. It is a 16-bit decimal number and must be unique on the network. This is the network number that will be advertised by the router as if it were a real network number.
to *network.node*	AppleTalk network address of the remote router. The argument *network* is the 16-bit network number in the range 0 to 65279. The argument *node* is the 8-bit node number in the range 0 to 254. Both numbers are decimal.
zone *zone-name*	Name of the zone on the remote network. The name can include special characters from the Apple Macintosh character set. To include a special character, type a colon followed by two hexadecimal characters. For zone names with a leading space character, enter the first character as the special sequence :20.

Default

No static routes are defined.

Command Mode

Global configuration

Usage Guidelines

You cannot delete a particular zone from the zone list without first deleting the static route.

Example

The following example creates a static route to the remote router whose address is 1.2 on the remote network 200 that is in the remote zone Remote:

```
appletalk static network 200 to 1.2 zone Remote
```

Related Commands

appletalk static cable-range
show appletalk route
show appletalk static

appletalk strict-rtmp-checking

To perform maximum checking of routing updates to ensure their validity, use the **appletalk strict-rtmp-checking** global configuration command. To disable the maximum checking, use the **no** form of this command.

appletalk strict-rtmp-checking
no appletalk strict-rtmp-checking

Syntax Description

This command has no arguments or keywords.

Default

Provide maximum checking

Command Mode

Global configuration

Usage Guidelines

Strict RTMP checking discards any RTMP packets arriving from routers that are not directly connected to the local router. This means that the local router does not accept any routed RTMP packets. Note that RTMP packets that need to be forwarded by the router are not discarded.

Example

The following example disables strict checking of RTMP routing updates:

```
no appletalk strict-rtmp-checking
```

Related Commands

appletalk require-route-zones
appletalk send-rtmps
appletalk timers

appletalk timers

To change the routing update timers, use the **appletalk timers** global configuration command. To return to the default routing update timers, use the **no** form of this command.

appletalk timers *update-interval valid-interval invalid-interval*
no appletalk timers [*update-interval valid-interval invalid-interval*]

Syntax Description

update-interval	Time, in seconds, between routing updates sent to other routers on the network. The default is 10 seconds.
valid-interval	Time, in seconds, that the router will consider a route valid without having heard a routing update for that route. The default is 20 seconds (two times the update interval).
invalid-interval	Time, in seconds, that the route is retained after the last update. The default is 60 seconds (three times the valid interval).

Defaults

update-interval: 10 seconds
valid-interval: 20 seconds
invalid-interval: 60 seconds

Command Mode

Global configuration

Usage Guidelines

Routes older than the time specified by *update-interval* are considered suspect. Once the period of time specified by *valid-interval* has elapsed without having heard a routing update for a route, the route becomes bad and is eligible for replacement by a path with a higher (less favorable) metric. During the *invalid-interval* period, routing updates include this route with a special "*notify neighbor*" metric. If this timer expires, the route is deleted from the routing table.

Note that you should not attempt to modify the routing timers without fully understanding the ramifications of doing so. Many other AppleTalk router vendors provide no facility for modifying their routing timers; should you adjust our router's AppleTalk timers such that routing updates do not arrive at these other routers within the normal interval, it is possible to degrade or destroy AppleTalk network connectivity.

If you change the routing update interval, be sure to do so for *all* routers on the network.

In rare instances, you might want to change this interval, such as when a router is busy and cannot send routing updates every 10 seconds or when slower routers are incapable of processing received routing updates in a large network.

Example

The following example increases the update interval to 20 seconds and the route-valid interval to 40 seconds:

```
appletalk timers 20 40 60
```

appletalk virtual-net

To add AppleTalk users logging in on an asynchronous line and using PPP encapsulation to an internal network, use the **appletalk virtual-net global configuration** command. To remove an internal network, use the **no** form of this command.

appletalk virtual-net *network-number zone-name*
no appletalk virtual-net *network-number zone-name*

Syntax Description

network-number	AppleTalk network address assigned to the interface. This is a 16-bit decimal network number in the range 0 to 65279. The network address must be unique across your AppleTalk internetwork.
zone-name	Name of a new or existing zone to which the AppleTalk user will belong.

Default

No virtual networks are predefined.

Command Mode

Global configuration

Usage Guidelines

A virtual network is a logical network that exists only within the router. It enables you—and by extension anyone who dials into the routeron the auxiliary port—to add an asynchronous interfaceto either a new or an existing AppleTalk zone.

Virtual networks work with both extended and nonextended AppleTalk networks.

If you issue the **appletalk virtual-net** command and specify a new AppleTalk zone name, the network number you specify is the only one associated with this zone. If you issue this command and specify an existing AppleTalk zone, the network number you specify is added to the existing zone.

The selected AppleTalk zone (either new or existing) is highlighted when you open the Macintosh Chooser window. From this window, you can access all available zones.

Example

The following example adds a user to the virtual network number 3 and and specifies the zone name *renegade*:

```
apple virtual-net 3 renegade
```

Related Commands

appletalk address
appletalk cable-range
appletalk client-mode
appletalk zone
show appletalk zone

appletalk zip-query-interval

To specify the interval at which the router sends ZIP queries, use the **appletalk zip-query-interval** global configuration command. To return to the default interval, use the **no** form of this command.

appletalk zip-query-interval *interval*
no zip-query-interval [*interval*]

Syntax Description

interval	Interval, in seconds, at which the router sends ZIP queries. It can be any positive integer. The default is 10 seconds.

Default

10 seconds

Command Mode

Global configuration

Usage Guidelines

The router uses the information received in response to its ZIP queries to update its zone table.

Example

The following example changes the ZIP query interval to 40 seconds:

```
appletalk zip-query-interval 40
```

appletalk zip-reply-filter

To configure a ZIP reply filter, use the **appletalk zip-reply-filter** interface configuration command. To remove a filter, use the **no** form of this command.

appletalk zip-reply-filter *access-list-number*
no appletalk zip-reply-filter [*access-list-number*]

Syntax Description

access-list-number	Number of the access list. This is a decimal number from 600 to 699.

Default

No access lists are predefined.

Command Mode

Interface configuration

Usage Guidelines

ZIP reply filters limit the visibility of zones from routers in unprivileged regions throughout the internetwork. These filters filter the zone list for each network provided by a router to neighboring routers to remove restricted zones.

ZIP reply filters apply to downstream routers, not to end stations on networks attached to the local router. With ZIP reply filters, when downstream routers request the names of zones in a network, the local router replies with the names of visible zones only. It does not reply with the names of zones that have been hidden with a ZIP reply filter. To filter zones from end stations, use GZL filters.

Example

The following example assigns a ZIP reply filter to Ethernet interface 0:

```
interface ethernet 0
appletalk zip-reply-filter 600
```

Related Commands

access-list additional-zones
access-list zone
show appletalk interface

appletalk zone

To set the zone name for the connected AppleTalk network, use the **appletalk zone** interface configuration command. To delete a zone, use the **no** form of this command.

appletalk zone *zone-name*
no appletalk zone [*zone-name*]

Syntax Description

zone-name	Name of the zone. The name can include special characters from the Apple Macintosh character set. To include a special character, type a colon followed by two hexadecimal characters. For zone names with a leading space character, enter the first character as the special sequence :20.

Default

No zone name is set.

Command Mode

Interface configuration

Usage Guidelines

If discovery mode is not enabled, you can specify this command only after an **appletalk address** or **appletalk cable-range** command. You can issue it multiple times if it follows the **appletalk cable-range** command.

On interfaces that have discovery mode disabled, you must assign a zone name in order for AppleTalk routing to begin.

If an interface is using extended AppleTalk, the first zone specified in the list is the default zone. The router always uses the default zone when registering NBP names for interfaces. Nodes in the network will select the zone in which they will operate from the list of zone names valid on the cable to which they are connected.

If an interface is using nonextended AppleTalk, repeated execution of the **appletalk zone** command will replace the interface's zone name with the newly specified zone name.

The **no** form of the command deletes a zone name from a zone list or deletes the entire zone list if you do not specify a zone name. For nonextended AppleTalk interfaces, the zone name argument is ignored. You should delete any existing zone-name list using the **no appletalk zone** interface subcommand before configuring a new zone list.

The zone list is cleared automatically when you issue an **appletalk address** or **appletalk cable-range** command. The list also is cleared if you issue the **appletalk zone** command on an *existing* network; this can occur when adding zones to a set of routers until all routers are in agreement.

Examples

The following example assigns the zone name Twilight to an interface:

```
interface Ethernet 0
appletalk cable-range 10-20
appletalk zone Twilight
```

The following example uses AppleTalk special characters to set the zone name to *Cisco•Zone*.

```
appletalk zone Cisco:A5Zone
```

Related Commands

appletalk address
appletalk cable-range
show appletalk zone

clear appletalk arp

To delete all entries or a specified entry from the AARP table, use the **clear appletalk arp** EXEC command.

clear appletalk arp [*network.node*]

Syntax Description

network.node	(Optional) AppleTalk network address to be deleted from the router's AARP table. The argument *network* is the 16-bit network number in the range 0 to 65279. The argument *node* is the 8-bit node number in the range 0 to 254. Both numbers are decimal.

Command Mode

EXEC

Example

The following example deletes all entries from the router's AARP table:

```
clear appletalk arp
```

Related Command

show appletalk arp

clear appletalk neighbor

To delete all entries or a specified entry from the neighbor table, use the **clear appletalk neighbor** EXEC command.

clear appletalk neighbor [*neighbor-address*]

Syntax Description

neighbor-address	(Optional) Network address of the neighboring router to be deleted from the neighbor table. The address is in the format *network.node*. The argument *network* is the 16-bit network number in the range 1 to 65279. The argument *node* is the 8-bit node number in the range 0 to 254. Both numbers are decimal.

Command Mode

EXEC

Usage Guidelines

You cannot clear the entry for an active neighbor, that is, for a neighbor that still has RTMP connectivity.

Example

The following example deletes the neighboring router 1.129 from the neighbor table:

```
clear appletalk neighbor 1.129
```

Related Command

show appletalk neighbors

clear appletalk route

To delete entries from the routing table, use the **clear appletalk route** EXEC command.

clear appletalk route [*network*]

Syntax Description

network	(Optional) Number of the network the route is to.

Command Mode

EXEC

Example

The following example deletes the route to network 1:

```
clear appletalk route 1
```

Related Command

show appletalk route

clear appletalk traffic

To reset AppleTalk traffic counters, use the **clear appletalk traffic** EXEC command.

clear appletalk traffic

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output after a **clear appletalk traffic** command was executed.

```
Router# clear appletalk traffic
Router# show appletalk traffic

AppleTalk statistics:
  Rcvd:  0 total, 0 checksum errors, 0 bad hop count
         0 local destination, 0 access denied
         0 for MacIP, 0 bad MacIP, 0 no client
         0 port disabled, 0 no listener
         0 ignored, 0 martians
  Bcast: 0 received, 0 sent
  Sent:  0 generated, 0 forwarded, 0 fast forwarded, 0 loopback
         0 forwarded from MacIP, 0 MacIP failures
         0 encapsulation failed, 0 no route, 0 no source
  DDP:   0 long, 0 short, 0 macip, 0 bad size
  NBP:   0 received, 0 invalid, 0 proxies
         0 replies sent, 0 forwards, 0 lookups, 0 failures
  RTMP:  0 received, 0 requests, 0 invalid, 0 ignored
         0 sent, 0 replies
  EIGRP: 0 received, 0 hellos, 0 updates, 0 replies, 0 queries
         0 sent,     0 hellos, 0 updates, 0 replies, 0 queries
         0 invalid, 0 ignored
  ATP:   0 received
  ZIP:   0 received, 0 sent, 0 netinfo
  Echo:  0 received, 0 discarded, 0 illegal
         0 generated, 0 replies sent
  Responder:  0 received, 0 illegal, 0 unknown
AppleTalk statistics:
         0 replies sent, 0 failures
  AARP:  0 requests, 0 replies, 0 probes
         0 martians, 0 bad encapsulation, 0 unknown
         0 sent, 0 failures, 0 delays, 0 drops
  Lost: 0 no buffers
  Unknown: 0 packets
  Discarded: 0 wrong encapsulation, 0 bad SNAP discriminator
```

Table 14-37 describes the fields shown in the **show appletalk traffic** display.

Related Commands

show appletalk macip-traffic
show appletalk traffic

ping (user)

To check host reachability and network connectivity, use the **ping** user EXEC command.

ping appletalk *network.node*

Syntax Description

appletalk	Specifies the AppleTalk protocol.
network.node	AppleTalk address of the system to ping.

Command Mode

EXEC

Usage Guidelines

The user **ping** (packet internet groper function) command provides a basic ping facility for users who do not have system privileges. This command is equivalent to the nonverbose form of the privileged **ping** command. It sends five 100-byte ping packets. The **ping** command sends Apple Echo Protocol (AEP) datagrams to other AppleTalk nodes to verify connectivity and measure round-trip times.

Only an interface that supports *HearSelf* can respond to packets generated at a local console and directed to an interface on the same router. Our routers support only *HearSelf* on Ethernet.

If the system cannot map an address for a host name, it will return an "%Unrecognized host or address" error message.

To abort a **ping** session, type the escape sequence. By default, this is Ctrl-^ X. You enter this by simultaneously pressing the Ctrl, Shift, and 6 keys, letting go, and then pressing the X key.

Table 14-2 describes the test characters displayed in **ping** responses.

Table 14-2 AppleTalk Ping Characters

Character	Meaning
!	Each exclamation point indicates the receipt of a reply from the target address.
.	Each period indicates the network server timed out while waiting for a reply from the target address.
B	A bad or malformed echo was received from the target address.
C	An echo with a bad DDP checksum was received.
E	Transmission of an echo packet to the target address failed.
R	Transmission of the echo packet to the target address failed due to lack of a route to the target address.

Sample Display

The following display shows input to and output from the user **ping** command.

```
Router> ping appletalk 1024.128
Type escape sequence to abort.
Sending 5, 100-byte AppleTalk Echoes to 1024.128, timeout is 2 seconds:
!!!!!
Success rate is 100 percent, round-trip min/avg/max = 4/4/8 ms
```

Related Command

ping (privileged)

ping (privileged)

To check host reachability and network connectivity, use the **ping** privileged EXEC command.

ping [**appletalk**] [*network.node*]

Syntax Description

appletalk	(Optional) Specifies the AppleTalk protocol.
network.node	(Optional) AppleTalk address of the system to ping.

Command Mode

Privileged EXEC

Usage Guidelines

The privileged **ping** (packet internet groper function) command provides a complete **ping** facility for users who have system privileges. The **ping** command sends Apple Echo Protocol (AEP) datagrams to other AppleTalk nodes to verify connectivity and measure round-trip times.

Only an interface that supports *HearSelf* can respond to packets generated at a local console and directed to an interface on the same router. Our routers only support *HearSelf* on Ethernet.

If the system cannot map an address for a host name, it will return an "%Unrecognized host or address" error message.

To abort a **ping** session, type the escape sequence. By default, this is Ctrl-^ X. You enter this by simultaneously pressing the Ctrl, Shift, and 6 keys, letting go, and then pressing the X key.

Table 14-3 describes the test characters displayed in **ping** responses.

Table 14-3 AppleTalk Ping Characters

Character	Meaning
!	Each exclamation point indicates the receipt of a reply (echo) from the target address.
.	Each period indicates the network server timed out while waiting for a reply from the target address.
B	The echo received from the target address was bad or malformed.
C	An echo with a bad DDP checksum was received.
E	Transmission of an echo packet to the target address failed.
R	Transmission of the echo packet to the target address failed due to lack of a route to the target address.

Sample Display of a Standard Ping

The following display shows a sample standard **appletalk ping** session:

```
Router# ping
Protocol [ip]: appletalk
Target Appletalk address: 1024.128
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Verbose [n]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte AppleTalk Echos to 1024.128, timeout is 2 seconds:
!!!!!
Success rate is 100 percent, round-trip min/avg/max = 4/4/8 ms
```

Sample Display Using Ping in Verbose Mode

When you answer `y` in response to the prompt `Verbose [n]`, **ping** runs in verbose mode. The following display shows a sample **appletalk ping** session when verbose mode is enabled:

```
Router# ping
Protocol [ip]: appletalk
Target AppleTalk address: 4.129
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Verbose [n]: y
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte AppleTalk Echos to 4.129, timeout is 2 seconds:
0 in 4 ms from 4.129 via 1 hop
1 in 8 ms from 4.129 via 1 hop
2 in 4 ms from 4.129 via 1 hop
3 in 8 ms from 4.129 via 1 hop
4 in 8 ms from 4.129 via 1 hop
Success rate is 100 percent, round-trip min/avg/max = 4/6/8 ms
```

Table 14-4 describes the fields in the verbose mode portion of the display.

Table 14-4 AppleTalk Ping Fields

Field	Meaning
0	Sequential number identifying the packet's relative position in the group of ping packets sent.
in 4 ms	Round-trip travel time of the ping packet, in milliseconds.
from 4.129	Source address of the ping packet.
via 1 hop	Number of hops the ping packet traveled to the destination.

Sample Display of NBP Ping and the Nbptest Facility

The AppleTalk **ping** command allows testing and informational lookup of NBP-registered entities. Use the NBP option when you find that AppleTalk zones are listed in the Chooser, but services in these zones are unavailable. When you enter **nbp** in response to the `Target AppleTalk address` prompt, **ping** starts the **nbptest** facility, which is an interactive, menu-driven facility. Type **help** or **?** to see the command list. Type **quit** to return to the EXEC prompt.

The following display shows how to initialize the AppleTalk **nbptest** utility:

```
Router# ping
Protocol [ip]: appletalk
Target AppleTalk address: nbp
nbptest>
```

Type **help** to display the following list of available commands:

```
nbptest> help
Tests are:
lookup:     lookup an NVE. prompt for name, type and zone
parms:      display/change lookup parms (ntimes, ncecs, interval)
zones:      display zones
poll:       for every zone, lookup all devices, using default
help|?:     print command list
quit:       exit nbptest
```

The following paragraphs summarize the **nbptest** tests that you can perform:

- **lookup**—Searches for NBP entities in a specific zone.
- **parms**—Sets the parameters used in subsequent lookup and pool tests.
- **zones**—Displays the router's current zone list. It is equivalent to the **show appletalk zone** command.
- **poll**—Searches for all devices in all zones.
- **help** or **?**—Displays the list of **nbptest** tests.
- **quit**—exit from the **nbptest** facility.

The remainder of this section shows and explains the output of the various **nbptest** commands.

When running any of the **nbptest** tests, you specify a nonprinting character by entering a three-character string that is the hexadecimal equivalent of the character. For example, type **:c5** to specify the NBP truncation wildcard.

The following display shows sample output of the **nbptest lookup** command:

```
nbptest> lookup
Entity name [=]:
Type of Service [ipgateway]: macintosh:c5
Zone [bldg-17]: engineering
(100n,50a,253s)[1]: 'userA:Macintosh IIcx@engineering'
(100n,16a,251s)[1]: 'userB:Macintosh II@engineering'
(200n,24a,253s)[1]: 'userC:Macintosh IIci@engineering'
(200n,36a,251s)[1]: 'userD:Macintosh II@engineering'
(300n,21a,252s)[1]: 'userE:Macintosh SE/30@engineering'
NBP lookup request timed out
Processed 6 replies, 7 events
```

Table 14-5 describes the fields shown in the display.

Table 14-5 AppleTalk Ping Nbptest Lookup Field Descriptions

Field	Description
Entity name [=]:	Name of NBP entity to display. The default is to display entries for all NBP entities. This is the same as typing =.
Type of Service	NBP service. The default is ipgateway. An = indicates any type of service.

Field	Description
Zone	Zone to search. The default is the zone of the current interface.
(100n,50a,253s) [1]	AppleTalk DDP address of the registered entity, in the format network, node address, and socket number. The number in brackets is either the current value of the field (if this is the first time you have invoked **nbptest**) or the value the field had the last time you invoked **nbptest**.
'userA:Macintosh IIcx@engineering'	NBP enumerator:NBP entity string of the registered entity.
NBP lookup request timed out	Indicates whether replies were heard within the timeout interval.
Processed 6 replies, 7 events	Number of NBP replies the router has received.

The following display shows sample output of the **nbptest parms** command:

```
nbptest> parms
maxrequests [5]:1
maxreplies [1]:100
interval [5]:10
```

Table 14-6 describes the fields shown in the display.

Table 14-6 AppleTalk Ping Nbptest Parms Field Descriptions

Field	Description
maxrequests	Maximum number of lookup retries. This is a number in the range 1 to 5. The default value is 5.
maxreplies	Maximum number of replies to accept for each lookup. This is a number in the range 1 to 500. The default is 1.
interval	Interval, in seconds, between each retry. This is in the range 1 to 60. The default is 5.

The following display shows sample output from the **nbptest zones** command:

```
nbptest> zones
Name                  Network(s)
UDP                   17 11
Heavenly              1161 6
Hostipal              55
Bldg-17               82 81 14 13
CSL EtherTalk         22
Twilight              1554 254 36 33 4
EtherTalk             22
LocalTalk             80
Total of 9 zones
```

Table 14-7 describes the fields shown in the display.

Table 14-7 AppleTalk Ping Nbptest Zones Field Descriptions

Field	Description
Name	Zone name.
Network(s)	Number or numbers of the AppleTalk networks assigned to the zone.

The following display shows sample output from the **nbptest poll** command:

```
nbptest> poll
poll: sent 2 lookups
(100n,82a,252s)[1]: ‘userA:Macintosh IIci@Zone one’
(200n,75a,254s)[1]: ‘userB:Macintosh IIcx@Zone two’
NBP polling completed.
Processed 2 replies, 2 events
```

Table 14-8 describes the fields shown in the display.

Table 14-8 AppleTalk Ping Nbptest Poll Field Descriptions

Field	Description
poll	Number of lookups the command sent.
(100n,82,252s) [1]	AppleTalk DDP address of the registered entity, in the format network, node address, and socket number. The number in brackets is either the current value of the field (if this is the first time you have invoked **nbptest**) or the value the field had the last time you invoked **nbptest**.
‘userA:Macintosh IIci@Zone one’	NBP enumerator:NBP entity string of the registered entity.
NBP polling completed.	Indicates that the polling completed successfully.
Processed 2 replies, 2 events	Number of NBP replies the router has received.

Related Commands

ping (user)
show appletalk zone

show appletalk access-lists

To display the AppleTalk access lists currently defined, use the **show appletalk access-lists** user EXEC command.

show appletalk access-lists

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show appletalk access-lists** command:

```
Router> show appletalk access-lists

AppleTalk access list 601:
         permit zone ZoneA
         permit zone ZoneB
         deny additional-zones
         permit network 55
         permit network 500
         permit cable-range 900-950
         deny includes 970-990
         permit within 991-995
         deny other-access
```

Table 14-9 describes fields shown in the display.

Table 14-9 Show AppleTalk Access-Lists Field Descriptions

Field	Description
AppleTalk access list 601:	Number of the AppleTalk access lists.
permit zone deny zone	Indicates whether access to an AppleTalk zone has been explicitly permitted or denied with the**access-list zone** command.
permit additional-zones deny additional-zones	Indicates whether additional zones have been permitted or denied with the **access-list additional-zones** command.
permit network deny network	Indicates whether access to an AppleTalk network has been explicitly permitted or denied with the**access-list network** command.
permit cable-range deny cable-range	Indicates the cable ranges to which access has been permitted or denied with the **access-list cable-range** command.
permit includes deny includes	Indicates the cable ranges to which access has been permitted or denied with the **access-list includes** command.
permit within deny within	Indicates the additional cable ranges to which access has been permitted or denied with the **access-list within** command.
permit other-access deny other-access	Indicates whether additional networks or cable ranges have been permitted or denied with the **access-list other-access** command.

Related Commands

access-list additional-zones
access-list cable-range
access-list includes
access-list network
access-list other-access
access-list within
access-list zone
appletalk access-group
appletalk distribute-list in
appletalk distribute-list out
appletalk getzonelist-filter

show appletalk adjacent-routes

To display routes to networks that are directly connected or that are one hop away, use the **show appletalk adjacent-routes** privileged EXEC command.

show appletalk adjacent-routes

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Usage Guidelines

The **show appletalk adjacent-routes** command provides a quick overview of the local environment that is especially useful when an AppleTalk internet consists of a large number of networks (typically, more then 600 networks).

You can use information provided by this command to determine if any local routes are missing or are misconfigured.

Sample Display

The following is sample output from the **show appletalk adjacent-routes** command:

```
Router# show appletalk adjacent-routes

Codes: R - RTMP derived, E - EIGRP derived, C - connected, S - static, P - proxy, 67
routes in internet

R Net 29-29 [1/G] via gatekeeper, 0 sec, Ethernet0, zone Engineering
C Net 2501-2501 directly connected, Ethernet1, no zone set
C Net 4160-4160 directly connected, Ethernet0, zone Low End SW Lab
C Net 4172-4172 directly connected, TokenRing0, zone Low End SW Lab
R Net 6160 [1/G] via urk, 0 sec, TokenRing0, zone Low End SW Lab
```

Table 14-10 describes the fields shown in the display.

Table 14-10 Show AppleTalk Adjacent-Routes Field Descriptions

Field	Description
Codes:	Codes defining source of route.
R	Route derived from an RTMP update.
E	Route derived from an EIGRP.
C	Directly connected network.RTMP update.
S	Static route.
P	Proxy route.
67 routes in internet	Total number of known routes in the AppleTalk network.
Net 29-29	Cable range or network to which the route goes.

Field	Description
[1/G]	Hop count, followed by the state of the route. Possible values for state include the following: • G—good (update has been received within the last 10 seconds) • S—suspect (update has been received more than 10 seconds ago but less than 20 seconds ago) • B—bad (update was received more than 20 seconds ago)
via	NBP registered name or address of the router that sent the routing information.
directly connected	Indicates that the network or cable range is directly connected to the router.
0 sec	Time, in seconds, since information about this network cable range was last received.
Ethernet0	Possible interface through which updates to this NBP registered name or address will be sent.
zone	Zone name assigned to the network or cable range sending this update.

show appletalk arp

To display the entries in the AARP cache, use the **show appletalk arp** privileged EXEC command.

show appletalk arp

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Usage Guidelines

ARP establishes associates between network addresses and hardware (MAC) addresses. This information is maintained in the router's ARP cache.

Sample Display

The following is sample output from the **show appletalk arp** command:

```
Router# show appletalk arp

Address      Age (min)  Type      Hardware Addr   Encap     Interface
2000.1               -  Hardware  0000.0c04.1111  SNAP      Ethernet1
2000.2               0  Dynamic   0000.0c04.2222  SNAP      Ethernet1
2000.3               0  Dynamic   0000.0c04.3333  SNAP      Ethernet3
2000.4               -  Hardware  0000.0c04.4444  SNAP      Ethernet3
```

Table 14-11 describes the fields shown in the display.

Table 14-11 Show AppleTalk ARP Field Descriptions

Field	Description
Address	AppleTalk network address of the interface.
Age (min)	Time, in minutes, that this entry has been in the ARP table. Entries are purged after they have been in the table for 240 minutes (4 hours). A hyphen indicates that this is a new entry.
Type	Indicates how the ARP table entry was learned. It can be one of the following: • Dynamic—Entry was learned via AARP. • Hardware—Entry was learned from an adapter in the router. • Pending—Entry for a destination for which the router does not yet know the address. When a packet requests to be sent to an address for which the router does not yet have the MAC-level address, the router creates an AARP entry for that AppleTalk address, then sends an AARP Resolve packet to get the MAC-level address for that node. When the router gets the response, the entry is marked "Dynamic." A pending AARP entry times out after 1 minute.
Hardware Addr	MAC address of this interface.

Field	Description
Encap	Encapsulation type. It can be one of the following: • ARPA—Ethernet-type encapsulation • SNAP—IEEE 802.3 encapsulation.
Interface	Type and number of the interface.

show appletalk aurp events

To display the pending events in the AURP update-events queue, use the **show appletalk aurp events** privileged EXEC command.

show appletalk aurp events

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Sample Display

The following is sample output from the **show appletalk aurp events** command:

```
Router> show appletalk aurp events

100-100, NDC EVENT pending
17043-17043, ND EVENT pending
```

Table 14-12 explains the fields shown in the display.

Table 14-12 Show AppleTalk AURP Events Fields

Field	Description
100-100	Network number or cable range.
NCD EVENT pending	Type of update event that is pending.

show appletalk aurp topology

To display entries in the AURP private path database, which consists of all paths learned from exterior routers, use the **show appletalk aurp topology** privileged EXEC command.

show appletalk aurp topology

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Sample Display

The following is sample output from the **show appletalk aurp topology** command:

```
Router# show appletalk aurp topology

30
             via Tunnel0, 3 hops
80
             via Tunnel0, 3 hops
101-101
             via Tunnel0, 8 hops
102-102
             via Tunnel0, 8 hops
103-103
             via Tunnel0, 8 hops
104-104
             via Tunnel0, 8 hops
105-105
             via Tunnel0, 8 hops
108-108
             via Tunnel0, 8 hops
109-109
             via Tunnel0, 9 hops
120-120
             via Tunnel0, 10 hops
125-125
             via Tunnel0, 8 hops
169-169
             via Tunnel0, 7 hops
201-205
             via Tunnel0, 4 hops
```

Table 14-13 explains the field shown in the display.

Table 14-13 Show AppleTalk AURP Topology Fields

Field	Description
30	AppleTalk network number or cable range.
via Tunnel0	Interface used to reach the network.
3 hops	Number of hops to the network.

show appletalk cache

To display the routes in the AppleTalk fast-switching table on an extended AppleTalk network, use the **show appletalk cache** EXEC command.

show appletalk cache

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

The **show appletalk cache** command displays information for all fast-switching route cache entries, whether or not they are valid.

Route entries are removed from the fast-switching cache if one of the following occurs:

- A route that was used has been deleted but has not yet been marked bad.
- A route that was used has gone bad.
- A route that was used has been replaced with a new route with a better metric.
- The state of route to a neighbor has changed from suspect to bad.
- The hardware address corresponding to a node address in the AARP cache has changed.
- The node address corresponding to a hardware address has changed.
- The ARP cache has been flushed.
- An ARP cache entry has been deleted.
- You have entered a **no appletalk routing**, an **appletalk route-cache**, or an **access-list** command.
- The encapsulation on the line has changed.
- An interface has become operational or nonoperational.

Sample Display

The following is sample output from the **show appletalk cache** command:

```
Router> show appletalk cache

AppleTalk Routing Cache, * = active entry, cache version is 227
Destination     Interface     MAC Header
*      29.0     Ethernet0     00000C00008200000C00D8DD
*  1544.000     Ethernet1     AA000400013400000C000E8C809B84BE02
*    33.000     Ethernet1     AA000400013400000C000E8C809B84BE02
```

Table 14-14 describes the fields shown in the display.

Table 14-14 Show AppleTalk Cache Field Descriptions

Field	Description
*	Indicates the entry is valid.
cache version is	Version number of the AppleTalk fast-switching cache.
Destination	Destination network for this packet.
Interface	Router interface through which this packet is transmitted.
MAC Header	First bytes of this packet's MAC header.

Related Command

appletalk route-cache

show appletalk domain

To display all domain-related information, use the **show appletalk domain** EXEC command.

show appletalk domain [*domain-number*]

Syntax Description

domain-number	(Optional) Number of an AppleTalk domain about which to display information. It can be a decimal integer from 1 through 1000000.

Command Mode

EXEC

Usage Guidelines

If you omit the argument *domain-number*, the **show appletalk domain** command displays information about all domains.

Sample Displays

The following is sample output from the **show appletalk domain** command:

```
Router# show appletalk domain

        AppleTalk   Domain   Information:

        Domain 1        Name : Xerxes
      ---------------------------------------
        State               : Active
        Inbound remap range  : 100-199
        Outbound remap range : 200-299
        Hop reduction        : OFF
        Interfaces in domain :
               Ethernet1     : Enabled

        Domain 2        Name : Desdemona
      ---------------------------------------
        Statue              : Active
        Inbound remap range  : 300-399
        Outbound remap range : 400-499
        Hop reduction        : OFF
        Interfaces in domain :
               Ethernet3     : Enabled
```

The following is sample output from the **show appletalk domain** command when you specify a domain number:

```
Router# show appletalk domain 1

        AppleTalk   Domain   Information:

        Domain 1        Name : Xerxes
       ---------------------------------------
        Statue              : Active
        Inbound remap range  : 100-199
        Outbound remap range : 200-299
        Hop reduction        : OFF
        Interfaces in domain :
               Ethernet1     : Enabled
```

Table 14-15 explains the fields shown in the displays.

Table 14-15 Show AppleTalk Domain Field Descriptions

Field	Description
Domain	Number of the domain as specified with the **appletalk domain name** global configuration command.
Name	Name of the domain as specified with the **appletalk domain name** global configuration command.
Status	Status of the domain. It can be either Active or Nonactive.
Inbound remap range	Inbound mapping range as specified with the **appletalk domain remap-range in** global configuration command.
Outbound remap range	Outbound mapping range as specified with the **appletalk domain remap-range out** global configuration command.
Hop reduction	Indicates whether hop reduction has been enabled with the **appletalk domain hop-reduction** global configuration command. It can be either OFF or ON.
Interfaces in domain	Indicates which interfaces are in the domain as specified with the **appletalk domain-group** interface configuration command and whether they are enabled.

Related Commands

appletalk domain-group
appletalk domain hop-reduction
appletalk domain name
appletalk domain remap-range

show appletalk eigrp neighbors

To display the neighbors discovered by Enhanced IGRP, use the **show appletalk eigrp neighbors** EXEC command.

show appletalk eigrp neighbors [*interface*]

Syntax Description

interface	(Optional) Displays information about the specified neighbor router.

Command Mode

EXEC

Usage Guidelines

The **show appletalk eigrp neighbors** command lists only the neighbors running AppleTalk Enhanced IGRP. To list all neighboring AppleTalk routers, use the **show appletalk neighbors** command.

Sample Display

The following is sample output from the **show appletalk eigrp neighbors** command:

```
Router# show appletalk eigrp neighbors

AT/EIGRP Neighbors for process 1, router id 83
Address                 Interface     Holdtime Uptime   Q      Seq  SRTT  RTO
                                      (secs)   (h:m:s)  Count  Num  (ms)  (ms)
warp.Ethernet1          Ethernet2     41       0:02:48  0      282  4     20
master.Ethernet2        Ethernet2     40       1:16:46  0      333  4     20
```

Table 14-16 explains the fields shown in the display.

Table 14-16 Show AppleTalk EIGRP Neighbors Field Descriptions

Field	Description
process 1	Number of the Enhanced IGRP routing process.
router id 83	Autonomous system number specified in the **appletalk routing global** configuration command.
Address	AppleTalk address of the AppleTalk Enhanced IGRP peer.
Interface	Interface on which the router is receiving hello packets from the peer.
Holdtime	Length of time, in seconds, that the router will wait to hear from the peer before declaring it down. If the peer is using the default hold time, this number will be less than 15. If the peer configures a nondefault hold time, it will be reflected here.
Uptime	Elapsed time, in hours, minutes, and seconds, since the local router first heard from this neighbor.
Q Count	Number of AppleTalk Enhanced IGRP packets (update, query, and reply) that the router is waiting to send.

Field	Description
Seq Num	Sequence number of the last update, query, or reply packet that was received from this neighbor.
SRTT	Smooth round-trip time. This is the number of milliseconds it takes for an AppleTalk Enhanced IGRP packet to be sent to this neighbor and for the local router to receive an acknowledgment of that packet.
RTO	Retransmission timeout, in milliseconds. This is the amount of time the router waits before retransmitting a packet from the retransmission queue to a neighbor.

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

appletalk routing
show appletalk neighbors †

show appletalk eigrp topology

To display the AppleTalk Enhanced IGRP topology table, use the **show appletalk eigrp topology** EXEC command.

show appletalk eigrp topology [*network-number* | **active** | **zero-successors**]

Syntax Description

network-number	(Optional) Number of the AppleTalk network whose topology table entry you want to display.
active	(Optional) Displays the entries for all active routes.
zero-successors	(Optional) Displays the entries for destinations for which no successors exist. These entries are destinations that the router currently does not know how to reach via Enhanced IGRP. This option is useful for debugging network problems.

Command Mode

EXEC

Usage Guidelines

All Enhanced IGRP routes that are received for a destination, regardless of metric, are placed in the topology table. The route to a destination that is currently in use is the first route listed. Routes that are listed as "connected" take precedence over any routes learned from any other source.

Sample Display

The following is sample output from the **show appletalk eigrp topology** command:

```
Router# show appletalk eigrp topology

IPX EIGRP Topology Table for process 1, router id 1

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - Reply status

P 3165-0, 1 successors, FD is 0
          via Redistributed (25601/0),
          via 100.1 (2198016/2195456), Fddi0
          via 4080.67 (2198016/53760), Serial4
P 3161-0, 1 successors, FD is 307200
          via Redistributed (1025850/0),
          via 100.1 (2198016/2195456), Fddi0
          via 4080.67 (2198016/1028410), Serial4
P 100-100, 1 successors, FD is 0
          via Connected, Fddi0
          via 4080.67 (2198016/28160), Serial4
P 4080-4080, 1 successors, FD is 0
          via Connected, Serial4
          via 100.1 (2172416/2169856), Fddi0
```

Table 14-17 explains the fields that may be displayed in the output.

Table 14-17 Show AppleTalk EIGRP Topology Field Descriptions

Field	Description
Codes	State of this topology table entry. Passive and Active refer to the Enhanced IGRP state with respect to this destination; and Update, Query and Reply refer to the type of packet that is being sent.
P – Passive	No Enhanced IGRP computations are being performed for this destination.
A – Active	Enhanced IGRP computations are being performed for this destination.
U – Update	Indicates that an update packet was sent to this destination.
Q – Query	Indicates that a query packet was sent to this destination.
R – Reply	Indicates that a reply packet was sent to this destination.
r – Reply status	Flag that is set after the router has sent a query and is waiting for a reply.
3165, 3161, and so on	Destination AppleTalk network number.
successors	Number of successors. This number corresponds to the number of next hops in the AppleTalk routing table.
FD	Feasible distance. This value is used in the feasibility condition check. If the neighbor's reported distance (the metric after the slash) is less than the feasible distance, the feasibility condition is met and that path is a feasible successor. Once the router determines it has a feasible successor, it does not have to send a query for that destination.
replies	Number of replies that are still outstanding (have not been received) with respect to this destination. This information appears only when the destination is in the Active state.
state	Exact Enhanced IGRP state that this destination is in. It can be the number 0, 1, 2, or 3. This information appears only when the destination is Active.
via	AppleTalk address of the peer who told the router about this destination. The first *n* of these entries, where *n* is the number of successors, are the current successors. The remaining entries on the list are feasible successors.
(345088/319488)	The first number is the Enhanced IGRP metric that represents the cost to the destination, The second number is the Enhanced IGRP metric that this peer advertised to us.
Ethernet0	Interface from which this information was learned.

The following is sample output from the **show appletalk eigrp topology** command when you specify an AppleTalk network number:

```
Router# show appletalk eigrp topology 3165

AT-EIGRP topology entry for 3165-0
State is Passive, Query origin flag is 1, 1 Successor(s)
Routing Descriptor Blocks:
0.0, from 0.0
  Composite metric is (25601/0), Send flag is 0x0, Route is Internal
  Vector metric:
    Minimum bandwidth is 2560000000 Kbit
    Total delay is 1000000 nanoseconds
    Reliability is 255/255
    Load is 1/255
    Minimum MTU is 1500
    Hop count is 0
```

```
100.1 (Fddi0), from 100.1
  Composite metric is (2198016/2195456), Send flag is 0x0, Route is External
  Vector metric:
    Minimum bandwidth is 1544 Kbit
    Total delay is 21100000 nanoseconds
    Reliability is 255/255
    Load is 1/255
    Minimum MTU is 1500
    Hop count is 2
4080.83 (Serial4), from 4080.83
   Composite metric is (2198016/53760), Send flag is 0x0, Route is Internal
   Vector metric:
   Minimum bandwidth is 1544 Kbit
   Total delay is 21100000 nanoseconds
   Reliability is 255/255
   Load is 1/255
   Minimum MTU is 1500
   Hop count is 2
```

Table 14-18 explains the fields that may be in the output.

Table 14-18 Show AppleTalk EIGRP Topology Field Descriptions for a Specified Network

Field	Description
3165	AppleTalk network number of the destination.
State is ...	State of this entry. It can be either Passive or Active. Passive means that no Enhanced IGRP computations are being performed for this destination, and Active means that they are being performed.
Query origin flag	Exact Enhanced IGRP state that this destination is in. It can be the number 0, 1, 2, or 3. This information appears only when the destination is Active.
Successors	Number of successors. This number corresponds to the number of next hops in the IPX routing table.
Next hop is ...	Indicates how this destination was learned. It can be one of the following: • Connected—The destination is on a network directly connected to this router. • Redistributed—The destination was learned via RTMP or another routing protocol. • AppleTalk host address—The destination was learned from that peer via this Enhanced IGRP process.
Ethernet0	Interface from which this information was learned.
from	Peer from whom the information was learned. For connected and redistributed routers, this is 0.0. For information learned via Enhanced IGRP, this is the peer's address. Currently, for information learned via Enhanced IGRP, the peer's AppleTalk address always matches the address in the "Next hop is" field.
Composite metric is	Enhanced IGRP composite metric. The first number is this router's metric to the destination, and the second is the peer's metric to the destination.
Send flag	Numeric representation of the "flags" field. It is 0 when nothing is being sent, 1 when an Update is being sent, 3 when a Query is being sent, and 4 when a Reply is being sent. Currently, 2 is not used.
Route is ...	Type of router. It can be either internal or external. Internal routes are those that originated in an Enhanced IGRP autonomous system, and external routes are those that did not. Routes learned via RTMP are always external.
Vector metric:	This section describes the components of the Enhanced IGRP metric.
Minimum bandwidth	Minimum bandwidth of the network used to reach the next hop.

Field	Description
Total delay	Delay time to reach the next hop.
Reliability	Reliability value used to reach the next hop.
Load	Load value used to reach the next hop.
Minimum MTU	Minimum MTU size of the network used to reach the next hop.
Hop count	Number of hops to the next hop.
External data	This section describes the original protocol from which this route was redistributed. It appears only for external routes.
Originating router	Network address of the router that first distributed this route into AppleTalk Enhanced IGRP.
External protocol..metric..delay	External protocol from which this route was learned. The metric will match the external hop count displayed by the **show appletalk route** command for this destination. The delay is the external delay.
Administrator tag	Currently not used.
Flag	Currently not used.

Related Command

show appletalk route

show appletalk globals

To display information and settings about the router's AppleTalk internetwork and other parameters, use the **show appletalk globals** EXEC command.

show appletalk globals

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show appletalk globals** command:

```
Router# show appletalk globals

AppleTalk global information:
      The router is a domain router.
      Internet is compatible with older, AT Phase1, routers.
      There are 67 routes in the internet.
      There are 25 zones defined.
      All significant events will be logged.
      ZIP resends queries every 10 seconds.
      RTMP updates are sent every 10 seconds.
      RTMP entries are considered BAD after 20 seconds.
      RTMP entries are discarded after 60 seconds.
      AARP probe retransmit count: 10, interval: 200.
      AARP request retransmit count: 5, interval: 1000.
      DDP datagrams will be checksummed.
      RTMP datagrams will be strictly checked.
      RTMP routes may not be propagated without zones.
      Alternate node address format will not be displayed.
```

Table 14-19 describes the fields shown in the display.

Table 14-19 Show AppleTalk Globals Field Descriptions

Field	Description
AppleTalk global information:	Heading for the command output.
The router is a domain router.	Indicates whether this router is a domain router.
Internet is compatible with older, AT Phase1, routers.	Indicates whether the AppleTalk internetwork meets the criteria for interoperation with Phase 1 routers.
There are 67 routes in the internet.	Total number of routes in the AppleTalk internet from which this router has heard in routing updates.
There are 25 zones defined.	Total number of valid zones in the current AppleTalk internet configuration.
All significant events will be logged.	Indicates whether the router has been configured with the **appletalk event-logging** command.
ZIP resends queries every 10 seconds.	Interval, in seconds, at which zone name queries are retried.

Field	Description
RTMP updates are sent every 10 seconds.	Interval, in seconds, at which the router sends routing updates.
RTMP entries are considered BAD after 20 seconds.	Time after which routes for which the router has not received an update will be marked as candidates for being deleted from the routing table.
RTMP entries are discarded after 60 seconds.	Time after which routes for which the router has not received an update will be deleted from the routing table.
AARP probe retransmit count: 10, interval: 200.	Number of AARP probe retransmissions that will be done before abandoning address negotiations and instead using the selected AppleTalk address, followed by the time, in milliseconds, between retransmission of ARP probe packets. You set these values with the **appletalk arp retransmit-count**and **appletalk arp interval** commands, respectively.
AARP request retransmit count: 5, interval: 1000.	Number of AARP request retransmissions that will be done before abandoning address negotiations and using the selected AppleTalk address, followed by the time, in milliseconds, between retransmission of ARP request packets. You set these values with the **appletalk arp retransmit-count** and **appletalk arp interval** commands, respectively.
DDP datagrams will be checksummed.	Indicates whether the **appletalk checksum** configuration command is enabled. When enabled, the router discards DDP packets when the checksum is incorrect and when the router is the final destination for the packet.
RTMP datagrams will be strictly checked.	Indicates whether the **appletalk strict-rtmp-checking** configuration command is enabled. When enabled, RTMP packets arriving from routers that are not directly connected to the router performing the check are discarded.
RTMP routes may not be propagated without zones.	Indicates whether the **appletalk require-route-zones** configuration command is enabled. When enabled, the router does not advertise a route to its neighboring routers until it has obtained a network/zone association for that route.
Alternate node address format will not be displayed.	Indicates whether AppleTalk addresses will be printed in numeric or name form. You configure this with the **appletalk lookup-type** and **appletalk name-lookup-interval** commands.

Related Commands

appletalk arp interval
appletalk arp retransmit-count
appletalk checksum
appletalk event-logging
appletalk lookup-type
appletalk name-lookup-interval
appletalk require-route-zones
appletalk strict-rtmp-checking

show appletalk interface

To display the status of the AppleTalk interfaces configured in the router and the parameters configured on each interface, use the **show appletalk interface** privileged EXEC command.

show appletalk interface [**brief**] [*type unit*]

Syntax Description

brief	(Optional) Displays a brief summary of the status of the AppleTalk interfaces.
type	(Optional) Interface type. It can be one of the following types: asynchronous, dialer, Ethernet (IEEE 802.3), Token Ring (IEEE 802.5), FDDI, HSSI, Virtual Interface, ISDN BRI, ATM interface, loopback, null, or serial.
number	(Optional) Interface number.

Command Mode

Privileged EXEC

Usage Guidelines

The **show appletalk interface** is particularly useful when you first enable AppleTalk on a router interface.

Sample Displays

The following is sample output from the **show appletalk interface** command for an extended AppleTalk network:

```
Router# show appletalk interface fddi 0

Fddi0 is up, line protocol is up
    AppleTalk cable range is 4199-4199
    AppleTalk address is 4199.82, Valid
    AppleTalk zone is "Low End SW Lab"
    AppleTalk address gleaning is enabled
    AppleTalk route cache is enabled
    AppleTalk domain is 1 (Domain 1)
    Interface will perform pre-FDDITalk compatibility
```

Table 14-20 describes the fields shown in the display as well as some fields not shown but that also may be displayed. Note that this command can show a node name in addition to the address, depending on how the router has been configured with the **appletalk lookup-type** and **appletalk name-lookup-interval** commands.

Table 14-20 Show AppleTalk Interface Field Descriptions for an Extended Network

Field	Description
FDDI is ...	Type of interface and whether it is currently active and inserted into the network (up) or inactive and not inserted (down).
line protocol	Indicates whether the software processes that handle the line protocol believe the interface is usable (that is, whether keepalives are successful).
AppleTalk cable range	Cable range of the interface.
AppleTalk address is ..., Valid	Address of the interface, and whether the address conflicts with any other address on the network ("Valid" means it does not).
AppleTalk zone	Name of the zone that this interface is in.
AppleTalk address gleaning	Indicates whether the interface is automatically deriving ARP table entries from incoming packets (referred to as "gleaning").
AppleTalk route cache	Indicates whether fast switching is enabled on the interface.
Interface will ...	Indicates that the AppleTalk interface will check to see if AppleTalk packets sent on the FDDI ring from routers running Cisco software releases prior to Release 9.0(3) or 9.1(2) are recognized.
AppleTalk domain	AppleTalk domain of which this interface is a member.

The following is sample output from the **show appletalk interface** command for a nonextended AppleTalk network:

```
Router# show appletalk interface ethernet 1

Ethernet 1 is up, line protocol is up
    AppleTalk address is 666.128, Valid
    AppleTalk zone is Underworld
    AppleTalk routing protocols enabled are RTMP
    AppleTalk address gleaning is enabled
    AppleTalk route cache is not initialized
```

Table 14-21 describes the fields shown in the display.

Table 14-21 Show AppleTalk Interface Field Descriptions for a Nonextended Network

Field	Description
Ethernet 1	Type of interface and whether it is currently active and inserted into the network (up) or inactive and not inserted (down).
line protocol	Indicates whether the software processes that handle the line protocol believe the interface is usable (that is, whether keepalives are successful).
AppleTalk address is ..., Valid	Address of the interface, and whether the address conflicts with any other address on the network ("Valid" means it does not).
AppleTalk zone	Name of the zone that this interface is in.
AppleTalk routing protocols enabled	AppleTalk routing protocols that are enabled on the interface.
AppleTalk address gleaning	Indicates whether the interface is automatically deriving ARP table entries from incoming packets (referred to as "gleaning").
AppleTalk route cache	Indicates whether fast switching is enabled on the interface.

The following is sample output from the **show appletalk interface brief** command:

```
Router# show appletalk interface brief

Interface   Address     Config        Status/Line Protocol   Atalk Protocol
TokenRing0  108.36      Extended      up                     down
TokenRing1  unassigned  not config'd  administratively down  n/a
Ethernet0   10.82       Extended      up                     up
Serial0     unassigned  not config'd  administratively down  n/a
Ethernet1   30.83       Extended      up                     up
Serial1     unassigned  not config'd  administratively down  n/a
Serial2     unassigned  not config'd  administratively down  n/a
Serial3     unassigned  not config'd  administratively down  n/a
Serial4     unassigned  not config'd  administratively down  n/a
Serial5     unassigned  not config'd  administratively down  n/a
Fddi0       50001.82    Extended      administratively down  down
Ethernet2   unassigned  not config'd  up                     n/a
Ethernet3   9993.137    Extended      up                     up
Ethernet4   40.82       Non-Extended  up                     up
Ethernet5   unassigned  not config'd  administratively down  n/a
Ethernet6   unassigned  not config'd  administratively down  n/a
Ethernet7   unassigned  not config'd  administratively down  n/a
```

Table 14-22 describes the fields shown in the display.

Table 14-22 Show AppleTalk Interface Brief Field Descriptions

Field	Description
Interface	Interface type and number.
Address	Address assigned to the interface.
Config	How the interface is configured. Possible values are extended, nonextended, and not configured.
Status/Line Protocol	Whether the software processes that handle the line protocol believe the interface is usable (that is, whether keepalives are successful).
Atalk Protocol	Whether AppleTalk routing is up and running on the interface.

Related Commands

appletalk discovery
appletalk lookup-type
appletalk name-lookup-interval

show appletalk macip-clients

To display status information about all known MacIP clients, use the **show appletalk macip-clients** EXEC command.

show appletalk macip-clients

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show appletalk macip-clients** command:

```
Router# show appletalk macip-clients

131.108.199.1@[27001n,69a,72s] 45 secs    'S/W Test Lab'
```

Table 14-23 describes the fields shown in the display.

Table 14-23 Show AppleTalk MacIP Clients Field Descriptions

Field	Description
131.108.199.1@	Client IP address.
[2700ln,69a,72s]	DDP address of the registered entity, showing the network number, node address, and socket number.
45 secs	Time, in seconds, since the last NBP confirmation was received.
'S/W Test Lab'	Name of the zone to which the MacIP client is attached.

Related Command

show appletalk traffic

show appletalk macip-servers

To display status information about a router's servers, use the **show appletalk macip-servers** EXEC command.

show appletalk macip-servers

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

The information in the **show appletalk macip-servers** display can help you quickly determine the status of your MacIP configuration. In particular, the STATE field can help identify problems in your AppleTalk environment.

Sample Display

The following is sample output from the **show appletalk macip-servers** command:

```
Router# show appletalk macip-servers

MACIP SERVER 1, IP 131.108.199.221,  ZONE 'S/W Test Lab' STATE is server_up
Resource #1 DYNAMIC 131.108.199.1-131.108.199.10, 1/10 IP in use
Resource #2 STATIC 131.108.199.11-131.108.199.20, 0/10 IP in use
```

Table 14-24 describes the fields shown in the display.

Table 14-24 Show AppleTalk MacIP Servers Field Descriptions

Field	Description
MACIP SERVER 1	Number of the MacIP server. This number is assigned arbitrarily.
IP 131.108.199.221	IP address of the MacIP server.
ZONE 'S/W Test Lab'	AppleTalk server zone specified with the **appletalk macip server** command.
STATE is server_up	State of the server. Table 14-26 lists the possible states. If the server remains in the "resource_wait" state, check that resources have been assigned to this server with either the **appletalk macip dynamic** or the **appletalk macip static** command.
Resource #1 DYNAMIC 131.108.199.1-131.108.199.10, 1/10 IP in use	Resource specifications defined in the **appletalk macip dynamic** and **appletalk macip static** commands. This list indicates whether the resource address was assigned dynamically or statically, identifies the IP address range associated with the resource specification, and indicates the number of active MacIP clients.

Use the **show appletalk macip-servers** command with **show appletalk interface** to identify AppleTalk network problems, as follows.

Step 1 Determine the state of the MacIP server using **show macip-servers**. If the STATE field continues to indicate an anomalous status (something other than "server_up," such as "resource_wait" or "zone_wait"), there is a problem.

Step 2 Determine the status of AppleTalk routing and the specific interface using the show appletalk interface command.

Step 3 If the protocol and interface are up, check the MacIP configuration commands for inconsistencies in the IP address and zone.

The STATE field of the **show appletalk macip-servers** command indicates the current state of each configured MacIP server. Each server operates according to the finite-state machine table described in Table 14-25. Table 14-26 describes the state functions listed in Table 14-25. These are the states that are displayed by the **show appletalk macip-servers** command.

Table 14-25 MacIP Finite-State Machine Table

State	Event	New State	Notes
initial	ADD_SERVER	resource_wait	Server configured
resource_wait	TIMEOUT	resource_wait	Wait for resources
resource_wait	ADD_RESOURCE	zone_wait	Wait for zone seeding
zone_wait	ZONE_SEEDED	server_start	Register server
zone_wait	TIMEOUT	zone_wait	Wait until seeded
server_start	START_OK	reg_wait	Wait for server register
server_start	START_FAIL	del_server	Could not start (possible configuration error)
reg_wait	REG_OK	server_up	Registration successful
reg_wait	REG_FAIL	del_server	Registration failed (possible duplicate IP address)
reg_wait	TIMEOUT	reg_wait	Wait until register
server_up	TIMEOUT	send_confirms	NBP confirm all clients
send_confirms	CONFIRM_OK	server_up	
send_confirms	ZONE_DOWN	zone_wait	Zone or IP interface down; restart
*	ADD_RESOURCE	*	Ignore, except resource_wait
*	DEL_SERVER	del_server	"No server" statement (HALT)
*	DEL_RESOURCE	ck_resource	Ignore
ck_resource	YES_RESOURCES	*	Return to previous state
ck_resource	NO_RESOURCES	resource_wait	Shut down and wait for resources

Table 14-26 Server States

State	Description
ck_resource	The server makes sure at least one client range is available. If not, it deregisters NBP names and returns to the resource_wait state.
del_server	State at which all servers end. In this state, the server deregisters all NBP names, purges all clients, and deallocates server resources.
initial	The state at which all servers start.
resource-wait	The server waits until a client range for the server has been configured.
send_confirms	The server tickles active clients every minute, deletes clients that have not responded within the last 5 minutes, and checks IP and AppleTalk interfaces used by MacIP server. If the interfaces are down or have been reconfigured, the server restarts.
server_start	The server registers configured IPADDRESS and registers as IPGATEWAY. It then opens an ATP socket to listen for IP address assignment requests, sends NBP lookup requests for existing IPADDRESSes, and automatically adds clients with addresses within one of the configured client ranges.
server_up	The server has registered. Being in this state enables routing to client ranges. The server now responds to IP address assignment requests.
zone_wait	The server waits until the configured AppleTalk zone name for the server is up. The server will remain in this state if no such zone has been configured or if AppleTalk routing is not enabled.
*	An asterisk in the first column represents any state. An asterisk in the second column represents a return to the previous state.

Related Commands

appletalk macip dynamic
appletalk macip server
appletalk macip static
show appletalk interface
show appletalk traffic

show appletalk macip-traffic

To display statistics about MacIP traffic through the router, use the **show appletalk macip-traffic** privileged EXEC command.

show appletalk macip-traffic

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Usage Guidelines

Use the **show appletalk macip-traffic** command to obtain a detailed breakdown of MacIP traffic that is sent through a router from an AppleTalk to an IP network. The output from this command differs from that of the **show appletalk traffic** command, which shows normal AppleTalk traffic generated, received, or routed by the router.

Sample Display

The following is sample output from the **show appletalk macip-traffic** command:

```
Router# show appletalk macip-traffic

 -- MACIP Statistics
                    MACIP_DDP_IN:    11062
                 MACIP_DDP_IP_OUT:    10984
   MACIP_DDP_NO_CLIENT_SERVICE:       78
                     MACIP_IP_IN:     7619
                 MACIP_IP_DDP_OUT:     7619
                  MACIP_SERVER_IN:       62
                 MACIP_SERVER_OUT:       52
            MACIP_SERVER_BAD_ATP:       10
          MACIP_SERVER_ASSIGN_IN:       26
         MACIP_SERVER_ASSIGN_OUT:       26
            MACIP_SERVER_INFO_IN:       26
           MACIP_SERVER_INFO_OUT:       26
```

Table 14-27 describes the fields shown in the display.

Table 14-27 Show AppleTalk MacIP Traffic Field Descriptions

Field	Description
MACIP_DDP_IN	Number of DDP packets received by the router.
MACIP_DDP_IP_OUT	Number of DDP packets received by the router that were sent to the IP network.
MACIP_DDP_NO_CLIENT_ SERVICE	Number of DDP packets received by the router for which there is no client.
MACIP_IP_IN	Number of IP packets received by the router.
MACIP_IP_DDP_OUT	Number of IP packets received by the router that were sent to the AppleTalk network.

Field	Description
MACIP_SERVER_IN	Number of packets destined for MacIP servers.
MACIP_SERVER_OUT	Number of packets sent by MacIP servers.
MACIP_SERVER_BAD_ATP	Number of MacIP allocation requests received with a bad request.
MACIP_SERVER_ASSIGN_IN	Number of MacIP allocation requests received asking for an IP address.
MACIP_SERVER_ASSIGN_ OUT	Number of IP addresses assigned.
MACIP_SERVER_INFO_IN	Number of MacIP packets received requesting server information.
MACIP_SERVER_INFO_OUT	Number of server information requests answered.

Related Command

show appletalk traffic

show appletalk name-cache

To display a list of NBP services offered by nearby routers and other devices that support NBP, use the **show appletalk name-cache** privileged EXEC command.

show appletalk name-cache

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Usage Guidelines

The **show appletalk name-cache** command displays the information currently in the NBP name cache.

Support for names allows you to easily identify and determine the status of any associated device. This can be important in AppleTalk internetworks where node numbers are dynamically generated.

You can authorize the **show appletalk name-cache** command to display any AppleTalk services of interest in local zones. This contrasts with the **show appletalk nbp** command, which you use to display services registered by the router.

Sample Display

The following is sample output from the **show appletalk name-cache** command:

```
Router# show appletalk name-cache

AppleTalk Name Cache:
Net     Adr  Skt  Name                Type            Zone
4160    19   8    gatekeeper          SNMP Agent      Underworld
4160    19   254  gatekeeper.Ether4   ciscoRouter     Underworld
4160    86   8    bones               SNMP Agent      Underworld
4160    86   72   131.108.160.78      IPADDRESS       Underworld
4160    86   254  bones.Ethernet0     IPGATEWAY       Underworld
```

Table 14-28 describes the fields shown in the display.

Table 14-28 Show AppleTalk Name-Cache Field Descriptions

Field	Description
Net	AppleTalk network number or cable range.
Adr	Node address.
Skt	DDP socket number.
Name	Name of the service.

Field	Description
Type	Device type. The possible types vary, depending on the service. The following are the Cisco server types: • ciscoRouter —Server is a Cisco router. • SNMP Agent —Server is an SNMP agent. • IPGATEWAY—Active MacIP server names. • IPADDRESS—Active MacIP server addresses.
Zone	Name of the AppleTalk zone to which this address belongs.

Related Command

show appletalk nbp

show appletalk nbp

To display the contents of the NBP name registration table, use the **show appletalk nbp** EXEC command.

show appletalk nbp

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

The **show appletalk nbp** command lets you identify specific AppleTalk nodes. It displays services registered by the router. In contrast, use the **show appletalk name-cache** command to display any AppleTalk services of interest in local zones.

Routers with active AppleTalk interfaces register each interface separately. The router generates a unique interface NBP name by appending the interface type name and unit number to the router name. For example, for the router named "router" that has AppleTalk enabled on Ethernet interface 0 in the zone Marketing, the NBP registered name is as follows:

```
router.Ethernet0:ciscoRouter@Marketing
```

Registering each interface on the router provides you with an indication that the router is configured and operating properly.

One name is registered for each interface. Other service types are registered once for each zone.

The router deregisters the NBP name if AppleTalk is disabled on the interface for any reason.

Sample Display

The following is sample output from the **show appletalk nbp** command:

```
Router# show appletalk nbp

Net  Adr Skt Name                    Type            Zone
4160 211 254 pag.Ethernet0           ciscoRouter     Low End SW Lab
4160 211   8 pag                     SNMP Agent      Low End SW Lab
4172  84 254 pag.TokenRing0          ciscoRouter     LES Tokenring
4172  84   8 pag                     SNMP Agent      LES Tokenring
200  75 254 myrouter. Ethernet1      ciscoRouter     Marketing    *
```

Table 14-29 describes the fields shown in the display as well as some fields not shown but that also may be displayed.

Table 14-29 Show AppleTalk NBP Field Descriptions

Field	Description
Net	AppleTalk network number.
Adr	Node address.

Field	Description
Skt	DDP socket number.
Name	Name of the service.
Type	Device type. The possible types vary, depending on the service. The following are the Cisco server types: • ciscoRouter—Cisco routers displayed by port. • SNMP Agent—SNMP agents displayed by zone if AppleTalk SNMP-over-DDP is enabled. • IPGATEWAY—Active MacIP server names. • IPADDRESS—Active MacIP server addresses.
Zone	Name of the AppleTalk zone to which this address belongs.
*	An asterisk in the right margin indicates that the name registration is pending confirmation.

Related Command

show appletalk name-cache

show appletalk neighbors

To display information about a AppleTalk routers that are directly connected to any of the networks to which this router is directly connected, use the **show appletalk neighbors** EXEC command.

show appletalk neighbors [*neighbor-address*]

Syntax Description

neighbor-address	(Optional) Displays information about the specified neighbor router.

Command Mode

EXEC

Usage Guidelines

If no neighbor address is specified, this command displays information about all AppleTalk routers.

The local router determines the AppleTalk network topology from its neighboring routers and learns from them most of the other information it needs to support the AppleTalk protocols.

Sample Displays

The following is sample output from the **show appletalk neighbors** command:

```
Router# show appletalk neighbors

AppleTalk neighbors:
  17037.2      anger.Ethernet0/0        Ethernet0/0, uptime 8:33:27, 2 secs
           Neighbor is reachable as a RTMP peer
  17037.108     Ethernet0/0, uptime 8:33:21, 7 secs
           Neighbor is reachable as a RTMP peer
  17037.248     Ethernet0/0, uptime 8:33:30, 4 secs
           Neighbor is reachable as a RTMP peer
  17046.2      anger.Ethernet0/1        Ethernet0/1, uptime 8:33:27, 2 secs
           Neighbor is reachable as a RTMP peer
  17435.87     firewall.Ethernet0/0     Ethernet0/3, uptime 8:33:27, 6 secs
           Neighbor is reachable as a RTMP peer
  17435.186    the-wall.Ethernet0       Ethernet0/3, uptime 8:33:24, 5 secs
           Neighbor is reachable as a RTMP peer
  17435.233    teach-gw.Ethernet0       Ethernet0/3, uptime 8:33:24, 7 secs
           Neighbor is reachable as a RTMP peer
  17036.1      other-gw.Ethernet5 Ethernet0/5, uptime 8:33:29, 9 secs
           Neighbor is reachable as a RTMP peer
  4021.5       boojum.Hssi4/0   Hssi1/0, uptime 10:49:02, 0 secs
         Neighbor has restarted 1 time in 8:33:11.
           Neighbor is reachable as a static peer
```

Table 14-30 describes the fields shown in this display. Depending on the configuration of the **appletalk lookup-type** and **appletalk name-lookup-interval**commands, a node name as well as a node address also may be shown in this display.

Table 14-30 Show AppleTalk Neighbors Field Descriptions

Field	Description
`31.86`	AppleTalk address of the neighbor router.
`Ethernet0/0`	Router interface through which the neighbor router can be reached.
`uptime 133:28:06`	Amount of time, in hours, minutes, and seconds, that the router has received this neighboring router's routing updates.
`2 secs`	Time, in seconds, since the router last received an update from the neighbor router.
Neighbor is reachable as a RTMP peer Neighbor is reachable as a static peer	Indicates how the route to this neighbor was learned.
`Neighbor is down.` `Neighbor has restarted 1 time`	Indicates whether neighbor is up or down, and number of times it has restarted in the specified time interval, displayed in the format hours:minutes:seconds.

The following is sample output from the **show appletalk neighbor** command when you specify the AppleTalk address of a particular neighbor:

```
Router# show appletalk neighbors 69.163

Neighbor 69.163, Ethernet0, uptime 268:00:52, last update 7 secs ago
  We have sent queries for 299 nets via 214 packets.
  Last query was sent 4061 secs ago.
  We received 152 replies and 0 extended replies.
  We have received queries for 14304 nets in 4835 packets.
  We sent 157 replies and 28 extended replies.
  We received 0 ZIP notifies.
  We received 0 obsolete ZIP commands.
  We received 4 miscellaneous ZIP commands.
  We received 0 unrecognized ZIP commands.
  We have received 92943 routing updates.
  Of the 92943 valid updates, 1320 entries were invalid.
  We received 1 routing update which were very late.
  Last update had 0 extended and 2 nonextended routes.
  Last update detail: 2 old
```

Table 14-31 describes the fields shown in this display. Depending on the configuration of the **appletalk lookup-type** and **appletalk name-lookup-interval** commands, a node name as well as a node address can be shown in this display.

Table 14-31 Show AppleTalk Neighbor Field Descriptions for a Specific Address

Field	Description
Neighbor 69.163	AppleTalk address of the neighbor.
Ethernet0	Interface through which the router receives this neighbor's routing updates.
uptime 268:00:52	Amount of time, in hours, minutes, and seconds, that the router has received this neighboring router's routing updates.
last update 7 secs ago	Time, in seconds, since the router last received an update from the neighbor router.
received queries	Number of RTMP queries that have been received from this neighbor.

Field	Description
Last query was sent	Time, in seconds, since last query was sent.
replies received	Number of RTMP replies the router has heard from this neighbor.
extended replies	Number of extended RTMP replies the router has received from this neighbor.
ZIP notifies	Number of ZIP notify packets the router has received from this neighbor.
obsolete ZIP commands	Number of nonextended-only (obsolete) ZIP commands the router has received from this neighbor.
miscellaneous ZIP commands	Number of ZIP commands (for example, GNI, GZI, and GMZ) the router received from end systems rather than from routers.
unrecognized ZIP commands	Number of bogus ZIP packets the router has received from this neighbor.
routing updates	Number of RMTP updates the router has received from this neighbor.
invalid entries	Of the routing update packets received from this neighbor, the number of invalid entries the router discarded.
Last update detail	Of the routing update packets received from this neighbor, the number the router already knew about.

Related Commands

appletalk lookup-type
appletalk name-lookup-interval

show appletalk remap

To display domain remapping information, use the **show appletalk remap** EXEC command.

show appletalk remap [**domain** *domain-number* [{**in** | **out**} [{**to** | **from**} *domain-network*]]]

Syntax Description

domain *domain-number*	(Optional) Number of an AppleTalk domain about which to display remapping information. It can be a decimal integer from 1 through 1000000.
in	(Optional) Displays remapping information about inbound packets, that is, on packets entering the local segment of the domain.
out	(Optional) Displays remapping information about outbound packets, that is on packets exiting from the local segment of the domain.
to	(Optional) Displays information about the network number or cable range to which an address has been remapped.
from	(Optional) Displays information about the original network number or cable range.
domain-network	(Optional) Number of an AppleTalk network.

Command Mode

EXEC

Usage Guidelines

If you omit all options keywords and arguments, the **show appletalk remap** command displays all remapping information about all domains.

Sample Displays

The following is sample output from the **show appletalk remap** command:

```
Router# show appletalk remap

       AppleTalk   Remapping  Table :
       ------------------------------

       Domain 1 : Domain 1   State : Active
       ----------------------------------------

       Direction : IN

       Domain Net(Cable)       Remapped to         Status
       3     - 3               100   - 100          Good

        Direction : OUT

       Domain Net(Cable)       Remapped to         Status
       1     - 1               200   - 200          Good

       Domain 2 : Domain 2   State : Active
       ----------------------------------------

       Direction : IN

       Domain Net(Cable)       Remapped to         Status

       Direction : OUT

       Domain Net(Cable)       Remapped to         Status
       2     - 2               400   - 400          Good
       100   - 100             401   - 401          Good
```

The following is sample output from the **show appletalk remap** command when you specify a domain number:

```
Router# show appletalk remap domain 1

       AppleTalk   Remapping  Table :
       ------------------------------

       Domain 1 : Domain 1   State : Active
       ----------------------------------------

       Direction : IN

       Domain Net(Cable)       Remapped to         Status
       3     - 3               100   - 100          Good

       Direction : OUT

       Domain Net(Cable)       Remapped to         Status
       1     - 1               201   - 201          Good
```

The following is sample output from the **show appletalk remap** command to display inbound remappings for AppleTalk network 100:

```
Router# show appletalk remap domain 1 in from 100

       AppleTalk   Remapping  Table :
             -----------------------------

For the Remap 100  the Domain  net is 3
```

Table 14-32 explains the fields shown in the display.

Table 14-32 Show AppleTalk Remap Field Descriptions

Field	Description
Domain	Number of the AppleTalk Internetwork Protocol domain.
State	State of the domain. It can be either Active or Nonactive.
Direction	Indicates whether the mapping is an inbound one (for packets entering the local domain segment) or an outbound one (for packets leaving the local domain segment).
Domain Net (Cable)	Network number or cable range that is being remapped.
Remapped to	Number or range of numbers to which a network number or cable range has been remapped.
Status	It can be one of the following values: • UnAssigned—The network number or cable range was just remapped. • UnZipped—The remapped network number or cable range is trying to acquire a zone list. This state is possible for inbound remappings only. • Suspect—The IOS suspects that it already has this entry in the routing table, and it is performing loop detection for this entry. This state is possible for inbound remappings only. • Good—The remapped entry has a complete zone list and, for inbound remappings only, it is in the main routing table. • Bad—The remapping entry is about to be deleted from the remapping table.

Related Command

appletalk domain remap-range

show appletalk route

To display all entries or specified entries in the AppleTalk routing table, use the **show appletalk route** EXEC command.

show appletalk route [*network* | *type number*]

Syntax Description

network	(Optional) Displays the routing table entry for the specified network.
type number	(Optional) Displays the routing table entries for networks that can be reached via the specified interface.

Command Mode

EXEC

Usage Guidelines

If you omit the arguments, this command displays all entries in the routing table.

Sample Displays

The following is sample output from the **show appletalk route** command for a nonextended AppleTalk network:

```
Router#  show appletalk route

Codes: R - RTMP derived, E - EIGRP derived, C - connected, A - AURP
P - proxy, S - static
5 routes in internet
C Net 258 directly connected, 1431 uses, Ethernet0, zone Twilight
R Net 6 [1/G] via 258.179, 8 sec, 0 uses, Ethernet0, zone The O
C Net 11 directly connected, 472 uses, Ethernet1, zone No Parking
R Net 2154 [1/G] via 258.179, 8 sec, 6892 uses, Ethernet0, zone LocalTalk
S Net 1111 via 258.144, 0 uses, Ethernet0, no zone set
[hops/state] state can be one of G:Good, S:Suspect, B:Bad
```

The following is sample output from the **show appletalk route** command for an extended AppleTalk network:

```
Router#  show appletalk route

Codes: R - RTMP derived, E - EIGRP derived, C - connected, A - AURP
P - proxy, S - static
5 routes in internet
E Net 10000 -10000 [1/G] via 300.199, 275 sec, Ethernet2, zone France
R Net 890 [2/G] via 4.129, 1 sec, Ethernet0, zone release lab
R Net 901 [2/G] via 4.129, 1 sec, Ethernet0, zone Dave's House
C Net 999-999 directly connected, Serial3, zone Magnolia Estates
R Net 2003 [4/G] via 80.129, 6 sec, Ethernet4, zone Bldg-13
```

Table 14-33 describes the fields shown in the two displays as well as some fields not shown but that also may be displayed. Depending on the configuration of the global configuration commands **appletalk lookup-type** and **appletalk name-lookup-interval**, a node name may appear in this display instead of a node address.

Table 14-33 Show AppleTalk Route Field Descriptions

Field	Description
Codes:	Codes defining how the route was learned.
R	Route learned from an RTMP update.
E	route learned from an EIGRP update.
C	Directly connected network.
A	Route learned from an AURP update.
S	Statically defined route.
P	Proxy route. (Proxy routes are included in outgoing RTMP updates as if they were directly connected routes, [although they are not really directly connected], since they are not associated with any interface. Whenever an NBQ BrRq for the zone in question is generated by anyone anywhere in the Internet, an NBP FwdReq is directed to any router connected to the proxy route. The Phase 2 router [which is the only router directly connected] converts the FwdReq to LkUps which are understood by Phase 1 routers, and sends them to every network in the zone.)
3 routes	Number of routes in the table.
Net 258	Network to which the route goes.
Net 999-999	Cable range to which the route goes.
directly connected	Indicates that the network is directly connected to the router.
1431 uses	Fair estimate of the number of times a route gets used. It actually indicates the number of times the route has been selected for use prior to operations such as access list filtering.
Ethernet0	Possible interface through which updates to the remote network will be sent.
zone Twilight	Name of zone of which the destination network is a member.
[1/G]	Number of hops to this network, followed by the state of the link to that network. The state can be one of the following letters: • G—Link is good. • S—Link is suspect. • B—Link is bad. The state is determined from the routing updates that occur at 10-second intervals. A separate and nonsynchronized event occurs at 20-second intervals, checking and flushing the ratings for particular routes that have not been updated. For each 20-second period that passes with no new routing information, a rating changes from G to S and then from S to B. After 1 minute with no updates, that route is flushed. Every time the router receives a useful update, the status of the route in question is reset to G. Useful updates are those advertising a route that is as good or better than the one currently in the table. When an AppleTalk route is poisoned by another router, its metric gets changed to poisoned (that is, 31 hops). The router then will age this route normally during a holddown period, during which the route will still be visible in the routing table.
via 258.179	Address of a router that is the next hop to the remote network.
via gatekeeper	Node name of a router that is the next hop to the remote network.

Field	Description
8 sec	Number of seconds that have elapsed since an RMTP update about this network was last received.

The following is sample output from the **show appletalk route** command when you specify a network number:

```
Router#  show appletalk route 69

Codes: R - RTMP derived, E - EIGRP derived, C - connected, A - AURP
P - proxy, S - static

The first zone listed for each entry is its default (primary) zone.

R Net 69-69 [2/G] via gatekeeper, 0 sec, Ethernet0, zone Empty Guf
Route installed 125:20:21, updated 0 secs ago
Next hop: gatekeeper, 2 hops away
Zone list provided by gatekeeper
Route has been updated since last RTMP was sent
Valid zones: “Empty Guf”
```

Table 14-34 describes the fields shown in the display.

Table 14-34 Show AppleTalk Route Field Descriptions for a Specified Network

Field	Description
Codes:	Codes defining how the route was learned.
R	Route learned from an RTMP update.
E	Route learned from an EIGRP update.
C	Directly connected network.
A	Route learned from an AURP update.
S	Statically defined route.
P	Proxy route.
67 routes in internet	Number of routes in the Apple Talk internet.
Net 69-69	Cable range to which the route goes. This is the number of the network you specified on the **show appletalk route** command line.

Field	Description
[2/G]	Number of hops to this network, followed by the state of the link to that network. The state can be one of the following letters: • G—Link is good. • S—Link is suspect. • B—Link is bad. The state is determined from the routing updates that occur at 10-second intervals. A separate and nonsynchronized event occurs at 20-second intervals, checking and flushing the ratings for particular routes that have not been updated. For each 20-second period that passes with no new routing information, a rating changes from G to S and then from S to B. After 1 minute with no updates, that route is flushed. Every time the router receives a useful update, the status of the route in question is reset to G. Useful updates are those advertising a route that is as good or better than the one currently in the table. When an AppleTalk route is poisoned by another router, its metric gets changed to poisoned (that is, 31 hops). The router then will age this route normally during a holddown period, during which the route will still be visible in the routing table.
via gatekeeper	Address or node name of a router that is the next hop to the remote network.
0 sec	Number of seconds that have elapsed since an RMTP update about this network was last received.
Ethernet0	Possible interface through which updates to the remote network will be sent.
zone Empty Guf	Name of zone of which the destination network is a member.
Route installed 125:20:21	Length of time, in hours, minutes, and seconds, since this route was first learned about.
updated 0 secs ago	Time, in seconds, since the router received an update for this route.
Next hop: gatekeeper	Address or node name of the router that is one hop away.
2 hops away	Number of hops to the network specified in the **show appletalk route** command line.
Zone list provided by gatekeeper	Address or node name of the router that provided the zone list included with the RTMP update.
Route has been updated since last RTMP was sent	Indicates whether the router has received a routing update from a neighboring router since the last time the router sent an RTMP update for this route.
Valid zones: “Empty Guf”	Zone names that are valid for this network.

Related Commands

appletalk lookup-type
appletalk name-lookup-interval
appletalk proxy-nbp
clear appletalk route

show appletalk sockets

To display all information or specified information about process-level operation in the sockets of an AppleTalk interface, use the **show appletalk sockets** privileged EXEC command.

show appletalk sockets [*socket-number*]

Syntax Description

socket-number	(Optional) Displays information about the specified socket number.

Command Mode

Privileged EXEC

Usage Guidelines

If no socket number is specified, this command displays information about all sockets.

Sample Display

The following is sample output from the **show appletalk sockets** command when you do not specify a socket number:

```
Router# show appletalk sockets

Socket   Name      Owner          Waiting/Processed
1        RTMP      AT RTMP          0    148766
2        NIS       AT NBP           0    15642
4        AEP       AT Maintenance   0    0
6        ZIP       AT ZIP           0    13619
8        SNMP      AT SNMP          0    0
253      PingServ  AT Maintenance   0    0
```

The following is sample output from the **show appletalk socket** command when you do specify a socket number:

```
Router# show appletalk sockets 6
6        ZIP       AT ZIP           0    13619
```

Table 14-35 describes the fields shown in these displays.

Table 14-35 Show AppleTalk Socket Field Descriptions

Field	Description
Socket	Socket number.
Name	Name of the socket.
Owner	Process that is managing communication with this socket.
Waiting/Processed	Number of packets waiting to be processed by the socket, and number of packets that have been processed by the socket since it was established.

show appletalk static

To display information the statically defined routes, use the **show appletalk static** EXEC command.

show appletalk static

Syntax Description

This command has no arguments or parameters.

Command Mode

EXEC

Sample Display

The following is sample output from the **show appletalk static** command:

```
Router# show appletalk static

List of Static Routes:
(3 Static routes in internet)
Net 100-110 [1/G] via 1000.2, 11456 sec, Serial0, zone Twilight
Net 412-412 [1/G] via 1000.2, 11623 sec, Serial0, zone Twilight
Net 514-515 [1/G] via 1000.2, 11061 sec, Serial0, zone Twilight
```

Table 14-36 describes the fields shown in the display.

Table 14-36 Show AppleTalk Static Field Descriptions

Field	Description
3 Static routes in internet	Number of static routes that have been defined.
[1/G]	Number of hops to this network, followed by the state of the link to that network. The state can be one of the following letters: • G—Link is good. • S—Link is suspect. • B—Link is bad. The state is determined from the routing updates that occur at 10-second intervals. A separate and nonsynchronized event occurs at 20-second intervals, checking and flushing the ratings for particular routes that have not been updated. For each 20-second period that passes with no new routing information, a rating changes from G to S and then from S to B. After 1 minute with no updates, that route is flushed. Every time the router receives a useful update, the status of the route in question is reset to G. Useful updates are those advertising a route that is as good or better than the one currently in the table. When an AppleTalk route is poisoned by another router, its metric gets changed to poisoned (that is, 31 hops). The router then will age this route normally during a holddown period, during which the route will still be visible in the routing table.
via ...	Address or node name of a router that is the next hop to the remote network.

Field	Description
11456 sec	Number of seconds that have elapsed since an RMTP update about this network was last received.
Serial0	Possible interface through which updates to the remote network will be sent.
zone Twilight	Name of zone of which the destination network is a member.

Related Commands

appletalk static cable-range
appletalk static network
show appletalk neighbors
show appletalk route

show appletalk traffic

To display statistics about AppleTalk traffic, including MacIP traffic, use the **show appletalk traffic** EXEC command.

show appletalk traffic

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

For MacIP traffic, an IP alias is established for each MacIP client and for the IP address of the MacIP server if it does not match an existing IP interface address. To display the client aliases, use the **show ip aliases** command.

Sample Display

The following is sample output from the **show appletalk traffic** command:

```
Router# show appletalk traffic

AppleTalk statistics:
  Rcvd:  357471 total, 0 checksum errors, 264 bad hop count
         321006 local destination, 0 access denied
         0 for MacIP, 0 bad MacIP, 0 no client
         13510 port disabled, 2437 no listener
         0 ignored, 0 martians
  Bcast: 191881 received, 270406 sent
  Sent:  550293 generated, 66495 forwarded, 1840 fast forwarded, 0 loopback
         0 forwarded from MacIP, 0 MacIP failures
         436 encapsulation failed, 0 no route, 0 no source
  DDP:   387265 long, 0 short, 0 macip, 0 bad size
  NBP:   302779 received, 0 invalid, 0 proxies
         57875 replies sent, 59947 forwards, 418674 lookups, 432 failures
  RTMP:  108454 received, 0 requests, 0 invalid, 40189 ignored
         90170 sent, 0 replies
  EIGRP: 0 received, 0 hellos, 0 updates, 0 replies, 0 queries
         0 sent,     0 hellos, 0 updates, 0 replies, 0 queries
         0 invalid, 0 ignored
  AURP: 0 Open Requests, 0 Router Downs
        0 Routing Information sent, 0 Routing Information received
        0 Zone Information sent, 0 Zone Information received
        0 Get Zone Nets sent, 0 Get Zone Nets received
        0 Get Domain Zone List sent, 0 Get Domain Zone List received
AppleTalk statistics:
        0 bad sequence
  ATP:   0 received
  ZIP:   13619 received, 33633 sent, 32 netinfo
  Echo:  0 received, 0 discarded, 0 illegal
         0 generated, 0 replies sent
  Responder:  0 received, 0 illegal, 0 unknown
         0 replies sent, 0 failures
```

```
    AARP:  85 requests, 149 replies, 100 probes
           84 martians, 0 bad encapsulation, 0 unknown
           278 sent, 0 failures, 29 delays, 315 drops
    Lost: 0 no buffers
    Unknown: 0 packets
    Discarded: 130475 wrong encapsulation, 0 bad SNAP discriminator
```

Table 14-37 describes the fields shown in the display.

Table 14-37 Show Apple Traffic Field Descriptions

Field	Description
Rcvd:	This section describes the packets that the router has received.
357741 total	Total number of packets the router received.
0 checksum errors	Number of packets that were discarded because their DDP checksum was incorrect. The DDP checksum is verified for packets that are directed to the router. It is not verified for forwarded packets.
264 bad hop count	Number of packets discarded because they had traveled too many hops.
321006 local destination	Number of packets addressed to the local router.
0 access denied	Number of packets discarded because they were denied by an access list.
0 for MacIP	Number of AppleTalk packets the router received that were encapsulated within an IP packet.
0 bad MacIP	Number of bad MacIP packets the router received and discarded. These packets may have been malformed or may not have included a destination address.
0 no client	Number of packets discarded because they were directed to a nonexistent MacIP client.
13510 port disabled	Number of packets discarded because routing was disabled for that port (extended AppleTalk only). This is the result of a configuration error or a packet's being received while the router is in verification/discovery mode.
2437 no listener	Number of packets discarded because they were directed to a socket that had no services associated with it.
0 ignored	Number of routing update packets ignored because they were from a misconfigured neighbor or because routing was disabled.
0 martians	Number of packets discarded because they contained bogus information in the DDP header. What distinguishes this error from the others is that the data in the header is never valid as opposed to not being valid at a given point in time.
Bcast:	Number of broadcast packets sent and received by the router.
Sent:	This section describes the packets that the router has transmitted.
550293 generated	Number of packets sent that were generated by the router.
66495 forwarded	Number of packets sent that were forwarded by the router.
1840 fast forwarded	Number of packets sent using routes from the fast-switching cache.
0 forwarded from MacIP	Number of IP packets the router forwarded that were encapsulated within an AppleTalk DDP packet.

Field	Description
0 MacIP failures	Number of MacIP packets sent that were corrupted during the MacIP encapsulation process.
436 encapsulation failed	Number of packets the router could not send because encapsulation failed. This can happen because encapsulation of the DDP packet failed or because AARP address resolution failed.
0 no route	Number of packets the router could not send because it knew of no route to the destination.
0 no source	Number of packets the router sent when it did not know its own address. This should happen only if something is seriously wrong with the router or network configuration.
DDP:	This section describes DDP packets seen by the router.
387265 long	Number of DDP long packets.
0 short	Number of DDP short packets.
0 macip	Number of IP packets encapsulated in an AppleTalk DDP packet that the router sent.
0 bad size	Number of packets whose physical packet length and claimed length differed.
NBP:	This section describes NBP packets.
302779 received	Total number of NBP packets received.
0 invalid	Number of invalid NBP packets received. Causes include invalid op code and invalid packet type.
0 proxies	Number of NBP proxy lookup requests received by the router when it was configured for NBP proxy transition usage.
57875 replies sent	Number of NBP replies the router has sent.
59947 forwards	Number of NBP forward requests the router has received or sent.
418674 lookups	Number of NBP lookups the router has received.
432 failures	Generic counter that increments any time the NBP process experiences a problem.
RTMP:	This section describes RTMP packets.
108454 received	Total number of RTMP packets the router has received.
0 requests	Number of RTMP requests the router has received.
0 invalid	Number of invalid RTMP packets received. Causes include invalid op code and invalid packet type.
40189 ignored	Number of RTMP packets the router ignored. One reason for this is that the interface is still in discovery mode and is not yet initialized.
90170 sent	Number of RTMP packets the router has sent.
0 replies	Number of RTMP replies the router has sent.
ATP:	This section describes ATP packets.
0 received	Number of ATP packets the router received.
ZIP:	This section describes ZIP packets.
13619 received	Number of ZIP packets the router has received.
33633 sent	Number of ZIP packets the router has sent.

Field	Description
32 netinfo	Number of packets that requested port configuration via ZIP GetNetInfo requests. These are commonly used during node startup and are occasionally used by some AppleTalk network management software packages.
Echo:	This section describes AEP packets.
0 received	Number of AEP packets the router received.
0 discarded	Number of AEP packets the router discarded.
0 illegal	Number of illegal AEP packets the router received.
0 generated	Number of AEP packets the router generated.
0 replies sent	Number of AEP replies the router sent.
Responder:	This section describes Responder Request packets.
0 received	Number of Responder Request packets the router received.
0 illegal	Number of illegal Responder Request packets the router received.
0 unknown	Number of Responder Request packets the router received that it did not recognize.
0 replies sent	Number of Responder Request replies the router sent.
0 failures	Number of Responder Request replies the router could not send.
AARP:	This section describes AARP packets.
85 requests	Number of AARP requests the router received.
149 replies	Number of AARP replies the router received.
100 probes	Number of AARP probe packets the router received.
84 martians	Number of AARP packets the router did not recognize. If you start seeing an inordinate number of martians on an interface, check whether a bridge has been inserted into the network. When a bridge is starting up, it floods the network with AARP packets.
0 bad encapsulation	Number of AARP packets received that had an unrecognizable encapsulation.
0 unknown	Number of AARP packets the router did not recognize.
278 sent	Number of AARP packets the router sent.
0 failures	Number of AARP packets the router could not send.
29 delays	Number of AppleTalk packets delayed while waiting for the results of an AARP request.
315 drops	Number of AppleTalk packets dropped because an AARP request failed.
Lost: 0 no buffers	Number of packets lost due to lack of buffer space.
Unknown: 0 packets	Number of packets whose protocol could not be determined.
Discarded:	This section describes the number of packets that were discarded.
130475 wrong	Number of packets discarded because they had the wrong encapsulation.That is, nonextended AppleTalk packets were on an extended AppleTalk network, or vice versa.
0 bad SNAP discrimination	Number of packets discarded because they had the wrong SNAP discriminator. This occurs when another AppleTalk device has implemented an obsolete or incorrect packet format.

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

clear appletalk traffic
show appletalk macip-traffic
show ip aliases †

show appletalk zone

To display all entries or specified entries in the zone information table, use the **show appletalk zone** EXEC command.

show appletalk zone [*zone-name*]

Syntax Description

zone-name	(Optional) Displays the entry for the specified zone.

Command Mode

EXEC

Usage Guidelines

If no zone name is specified, the command displays all entries in the zone information table.

You can use this command on extended and nonextended networks.

A zone name can be associated with multiple network addresses or cable ranges, or both. There is not a one-to-one correspondence between a zone name and a local-area network (LAN); a zone name may correspond to one or more networks (LANs or network interfaces). This means that a zone name will effectively replace multiple network addresses in zone filtering. This is reflected in the output of the **show appletalk zone** command. For example, the zone named Mt. View 1 in the sample display below is associated with two network numbers and four cable ranges.

Sample Display

The following is sample output from the **show appletalk zone** command:

```
Router# show appletalk zone

Name                    Network(s)
Gates of Hell           666-666
Engineering             3 29-29 4042-4042
customer eng            19-19
CISCO IP                4140-4140
Dave's House            3876 3924 5007
Narrow Beam             4013-4013 4023-4023 4037-4037 4038-4038
Low End SW Lab          6160 4172-4172 9555-9555 4160-4160
Tir'n na'Og             199-199
Mt. View 1              7010-7010 7122 7142 7020-7020 7040-7040 7060-7060
Mt. View 2              7152 7050-7050
UDP                     1112-12
Empty Guf               69-69
Light                   80
europe                  2010 3010 3034 5004
Bldg-13                 4032 5026 61669 3012 3025 3032 5025 5027
Bldg-17                 3004 3024 5002 5006
```

Table 14-38 describes the fields shown in the display.

Table 14-38 Show AppleTalk Zone Field Descriptions

Field	Description
Name	Name of the zone.
Network	Cable ranges or network numbers assigned to this zone.

The following is sample output from the **show appletalk zone** command when you specify a zone name:

```
Router# show appletalk zone CISCO IP

AppleTalk Zone Information for CISCO IP:
  Valid for nets: 4140-4140
  Not associated with any interface.
  Not associated with any access list.
```

Table 14-39 describes the fields shown in the display.

Table 14-39 Show AppleTalk Zone Field Descriptions for a Specific Zone Name

Field	Description
AppleTalk Zone Information for CISCO IP:	Name of the zone.
Valid for nets: 4140-4140	Cable range(s) or network numbers assigned to this zone.
Not associated with any interface.	Interfaces that have been assigned to this zone.
Not associated with any access list.	Access lists that have been defined for this zone.

Related Command

appletalk zone

CHAPTER 15

Banyan VINES Commands

The Banyan VINES protocol is a networking system for personal computers. "VINES" is an acronym for Virtual Network System. This proprietary protocol was developed by Banyan and is derived from Xerox's XNS protocol. Cisco's implementation of VINES has been designed in conjunction with Banyan.

Cisco's implementation of Banyan VINES provides routing of VINES packets on all media types. Although the software automatically determines a metric value that it uses to route updates based on the delay set for the interface, Cisco's software implementation allows you to customize the metric. Cisco's implementation also offers address resolution to respond to address requests. MAC-level echo support is also available for Ethernet, IEEE 802.2, Token Ring, and FDDI media. Name-to-address mapping for VINES host names also is supported, as are access lists to filter outgoing packets.

Use the commands in this chapter to configure and monitor VINES networks. For VINES configuration information and examples, refer to the "Configuring Banyan VINES" chapter in the *Router Products Configuration Guide*.

clear vines cache

To delete entries from the VINES fast-switching cache , use the **clear vines cache** EXEC command.

clear vines cache [**interface** *interface* | **neighbor** *address* | **server** *network*]

Syntax Description

interface *interface*	(Optional) Deletes from the fast-switching cache table any entry that has one or more paths that go through the specified interface.
neighbor *address*	(Optional) Deletes from the fast-switching cache table any entry that has one or more paths via the specified neighbor router.
server *network*	(Optional) Deletes from the fast-switching cache table any entry whose network number part of the destination address matches the specified network address.The argument *network* can be either a 4-byte hexadecimal number or a 4-byte decimal number (if you have issued a **vines decimal** command).

Command Mode

EXEC

Usage Guidelines

If you do not specify any keywords or arguments, all entries in the fast-switch cache are deleted.

The fast-switching cache is a table of routes used when fast switching is enabled.

Examples

The following example deletes from the fast-switching cache table all entries from the VINES fast-switching cache table:

```
clear vines cache
```

The following example deletes all entries whose destination server has the address 30002E6D:

```
clear vines cache server 30002E6D
```

Related Commands

show vines cache
vines decimal
vines route-cache

clear vines ipc

To delete VINES IPC connection blocks from the router, use the **clear vines ipc** EXEC command.

clear vines ipc *number*

Syntax Description

number	Hexadecimal number of the IPC connection to delete.

Command Mode

EXEC

Usage Guidelines

An IPC connection entry is built each time the router initiates or receives an IPC DATA message from a router that is not already in this table.

Examples

The following example deletes IPC connection 0x1D from the table of VINES IPC connections:

```
clear vines ipc 1D
```

Related Command

show vines ipc

clear vines neighbor

To delete entries from the neighbor table, use the **clear vines neighbor** EXEC command.

clear vines neighbor {*network* | *****}

Syntax Description

network	Network number of the neighbor whose entry should be deleted from the neighbor table. The argument *network* can be either a 4-byte hexadecimal number or a 4-byte decimal number (if you have issued a **vines decimal** command).
*****	Deletes all entries from the neighbor path table except the entry for the local router.

Command Mode

EXEC

Usage Guidelines

The neighbor table contains an entry for each of the router's neighbor nodes.

Deleting an entry from the neighbor table also deletes any routes in the routing table that have that neighbor as the first hop and all fast-switching cache entries that have that neighbor as the first hop in any of their paths.

Example

The following example deletes all entries from the neighbor table:

```
clear vines neighbor *
```

Related Commands

clear vines route
show vines neighbor
show vines route
vines decimal
vines neighbor
vines route

clear vines route

To delete network addresses from the routing table, use the **clear vines route** EXEC command.

clear vines route {*network* | ***}**

Syntax Description

network	Network number of the entry to delete from the routing table. The argument *network* can be either a 4-byte hexadecimal number, a 4-byte decimal number (if you have issued a **vines decimal** command), or a host name (if you have issued a **vines host** command).
*****	Deletes all entries from the routing table.

Command Mode

EXEC

Usage Guidelines

Deleting an entry from the routing table with the **clear vines route** command also deletes any entries in the fast-switching table that are a part of that logical network.

Example

The following example deletes all entries from the VINES routing table:

```
clear vines route *
```

Related Commands

clear vines neighbor
show vines neighbor
show vines route
vines decimal
vines host
vines route

clear vines traffic

To clear all VINES-related statistics that are displayed by the **show vines traffic** command, use the **clear vines traffic** EXEC command.

clear vines traffic

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

The **clear vines traffic** command clears only the statistics displayed by the **show vines traffic** command. It has no effect on the value of the VINES counters retrieved by SNMP.

Example

The following example zeros all VINES-related traffic statistics:

```
clear vines traffic
```

Related Command

show vines traffic

ping

To determine basic network connectivity, use the **ping** EXEC command.

ping [**vines**] [*address*]

Syntax Description

vines	(Optional) Specifies the VINES protocol. If you omit this keyword, the router prompts for it.
address	(Optional) Address of system to ping. If you omit the address, the router prompts for it.

Command Mode

EXEC

Usage Guidelines

The **ping** command determines network connectivity by sending datagrams to another host on the network.

Sample Display

The following is sample output from the **ping** command:

```
Router# ping vines 27AF92:1
Type escape sequence to abort.
Sending 5, 100-byte VINES Echos to 27AF92:1,
timeout is 2 seconds:
!!!!!
Success rate is 100 percent, round-trip min/avg/max = 4/7/8 ms

Router# ping
Protocol [ip]: vines
Target VINES address: 27AF92:1
Repeat count [5]: 10
Datagram size [100]: 500
Timeout in seconds [2]:
Verbose [n]:
Type escape sequence to abort.
Sending 10, 500-byte VINES Echos to 27AF92:1,
timeout is 2 seconds:
!!!!!!!!!!
Success rate is 100 percent, round-trip min/avg/max = 4/7/8 ms
```

show vines access

To display the VINES access lists currently defined, use the **show vines access** EXEC command.

show vines access [*access-list-number*]

Syntax Description

access-list-number	(Optional) Number of the access list to display.

Command Mode

EXEC

Usage Guidelines

If no access list number is specified, all access lists are displayed.

Sample Display

The following is sample output from the **show vines access** command:

```
Router# show vines access
Vines access list 1
  deny   SPP 30015800:0001 00000000:00000000 202 00123456:8005 00000000:0000 249
  permit IP 00000000:0000 FFFFFFFF:FFFF 00000000:0000 FFFFFFFF:FFFF
Vines access list 101
  deny   SPP 00112233:0001 00000000:0000 0006 0000
             00123456:8005 00000000:00000000 0000 FFFF
  permit IP 00000000:0000 FFFFFFFF:FFFF 00000000:0000 FFFFFFFF:FFFF
```

Table 15-1 describes the fields shown in the display.

Table 15-1 Show VINES Access Field Descriptions

Field	Description
Vines access list ...	Number of the VINES access list.
deny	Networks to which access is denied.
permit	Networks to which access is permitted.

Related Commands

vines access-list (standard)
vines access-list (extended)
vines access-list (simple)

show vines cache

To display the contents of the VINES fast-switching cache, use the **show vines cache** EXEC command.

show vines cache [*address* | **interface** *type number* | **neighbor** *address* | **server** *network*]

Syntax Description

address	(Optional) Displays the entry in the fast-switching cache for the specified station.
interface *type number*	(Optional) Displays all neighbors in the fast-switching cache that are accessible via the specified interface type and number.
neighbor *address*	(Optional) Displays all routes in the VINES fast-switching cache that have the specified neighbor as their first hop. The argument *address* is a 6-byte hexadecimal number in the format *network:host*, where *network* is 4 bytes and *host* is 2 bytes, a 4-byte decimal number in the same format (if you have issued a **vines decimal** command), or a host name (if you have issued a **vines host** command).
server *network*	(Optional) Displays all entries in the VINES fast-switching cache that are in the specified logical network. The argument *network* can be either a 4-byte hexadecimal number or a 4-byte decimal number (if you have issued a **vines decimal** command).

Command Mode

EXEC

Usage Guidelines

If no keywords or arguments are specified, all entries in the fast-switching cache are displayed.

Sample Display

The following is sample output from **show vines cache** command. This sample shows all entries in the VINES fast-switching cache.

```
Router# show vines cache
Vines fast switching cache version is 36

Hash  Destination     Int    Age  Length  MAC Header
13/00 Router1        *T0      46   16/18  10005A746A3600003080FB06BCBC03BA
27/00 Router2         E1      11   14/14  00000C01D87C00000C0158010BAD
                     *T0      11   16/18  0000300043550000 3080FB06BCBC03BA
3E/00 Router3        *T0      42   16/18  10005A6FBC15000003080FB06BCBC03BA
72/00 30002E6D:0001   E1      32   14/14  00000C01D87C00000C0158010BAD
                     *T0      32   16/18  0000300043550000 3080FB07BCBC03BA
                      T0      32   16/18  10005A6FBC1500003080FB06BCBC03BA
                      T0      32   16/18  10005A6FBC1500003080FB06BCBC03BA
FE/00 Router4        *E2     264   14/14  00000C0124EA00000C0151AF0BAD
```

Table 15-2 describes fields shown in the display.

Note that neighbor information is not explicitly displayed by the **show vines cache** command. However, you can determine it by looking at the neighbor and routing tables (using the **show vines neighbor** and **show vines route** commands, respectively).

Table 15-2 Show VINES Cache Field Descriptions

Field	Description
Vines fast switching cache version	Version number of the VINES fast-switching cache table. The number is incremented each time an entry is added to or deleted from this table.
Hash	Position of this entry in the neighbor table.
Destination	Name or address of the destination station.
Int	Interface out which the packet will be sent. An asterisk preceding the interface name indicates that this is the next entry that will be used for the destination.
Age	Age of the entry, in seconds.
Length	Stored length of the packet's MAC header, followed by a slash and the actual length of the MAC header. Both lengths do not include the length of the Type field. These two lengths may differ because the initial bytes of Token Ring and FDDI frames are not stored.
MAC Header	MAC header that will be used to reach the destination.

Related Commands

clear vines cache
show vines neighbor
show vines route
vines decimal
vines route-cache

show vines host

To display the entries in the VINES host name table, use the **show vines host** EXEC command.

show vines host [*name*]

Syntax Description

name	(Optional) Displays the entry in the VINES name table that has the specified name.

Command Mode

EXEC

Usage Guidelines

If no name is specified, all entries in the host name table are displayed.

Sample Display

The following is sample output from the **show vines host** command:

```
Router# show vines host
Name          Address
Router1       0027AF9A:0001
Router2       0027D0E4:0001
Router3       002ABFAA:0001
Router4       30015800:0001
```

Table 15-3 describes the fields shown in the display.

Table 15-3 Show VINES Host Field Descriptions

Field	Description
Name	Name of the VINES host.
Address	Address of the VINES host.

Related Command

vines host

show vines interface

To display status of the VINES interfaces configured in the router and the parameters configured on each interface, use the **show vines interface** EXEC command.

show vines interface [*type number*]

Syntax Description

type	(Optional) Interface type.
number	(Optional) Interface number.

Command Mode

EXEC

Usage Guidelines

If you omit all keywords, this command displays values for all interfaces, and displays all VINES global parameters.

Sample Display

The following is sample output from the **show vines interface** command:

```
Router# show vines interface
VINES address is 3000902D:0001
  Next client will be 3000902D:8001
  Addresses are displayed in hexadecimal format.
  Slowest update interval is 90 seconds
  Roll Call timer queue:
    Neighbor Router3-Et2-0000.0c01.24ea in 180 seconds
  Sequence: 01029DD7, Packet ID: 00000003
  Reassembly timer queue:  (empty)
  Retry timer queue:  (empty)
   Participating in vines time of day synchronization
Hssi0 is down, line protocol is down
  VINES protocol processing disabled
Fddi0 is up, line protocol is up
  VINES broadcast encapsulation is ARPA
  Interface metric is 0008 [0 5000] (0.1000 seconds)
  Split horizon is enabled
  ARP processing is dynamic, state is learning (for another 18 seconds)
  Special serverless net processing enabled
  Outgoing access list is not set
  Fast switching is enabled
  Routing updates every 90 seconds. Next in 50 seconds.
  Next synchronization update in 11:58:17.
  Nodes present: 0 5.5x servers, 0 5.5x routers, 0 5.5x clients
                 0 4.11 servers, 0 4.11 routers, 0 4.11 clients
  Neighbors:  none.
```

Table 15-4 describes the fields that may be shown in the display.

Table 15-4 Show VINES Interface Field Descriptions

Field	Description
VINES address	Address of the router.
Next client will be	Address the router will assign to the next client that requests an address. This line is interesting only if the router has been configured via the **vines arp-enable**command to respond to address assignment requests.
Addresses	Indicates whether addresses will be displayed as decimal or hexadecimal numbers.
Slowest update interval	Indicates the longest time interval (in seconds) between routing updates on any of the router's interfaces.
Roll Call timer Neighbor	Displays a list of all neighbor paths for which an RTP request will be sent on a regular basis, and the interval until that timer expires.
Sequence	Current SRTP sequence number for this router.
Packet ID	Identifier number that will be used on the last SRTP update message sent by this router
Reassembly timer	Displays a list of all neighbor paths for which an SRTP update is currently being reassembled, and the interval until that timer expires.
Retry timer	Displays a list of all neighbor paths for which an SRTP request is currently being retried, and the interval until that timer expires.
Participating in vines time of day synchronization	Indicates whether the router is participating in VINES time-of-day synchronization. This is controlled by the **vines time participate** global configuration command.
Hssi0/Ethernet 0/Ethernet 1 is up/down	Type and number of interface, and whether it is currently active and inserted into network (up) or inactive and not inserted (down).
Line protocol is	Indicates whether the software processes that handle the line protocol believe the interface is usable (that is, whether keepalives are successful). This field can report the values "up," "down," and "administratively down."
VINES protocol processing disabled	Indicates that VINES processing is not enabled on the interface (that is, you have not issued a **vines metric** command on the interface).
VINES broadcast encapsulation	Type of encapsulation used for VINES broadcast packets, as defined with the **vines encapsulation** command. This field can report the values "arpa," "vines-tr," and "snap."
Interface metric	Metric that has been configured for the interface with the **vines metric** command. The metric is shown in internal form, configuration form, and in seconds.
Split horizon	Indicates whether split horizon has been enabled or disabled (via the **vines split-horizon** command).
ARP processing	Indicates whether this interface will process ARP packets, as specified by the **vines arp-enable** command.
Special serverless net processing	Indicates whether this interface is defined via the **vines serverless** command as being connected to a serverless network.
Outgoing access list	Indicates whether an access list is set.
Fast switching	Indicates whether fast switching has been enabled via the **vines route-cache** command). The value reported in this field can be "enabled," "disabled," or "not supported."

Field	Description
Routing updates every Next in	Frequency of routing updates, in seconds. This also indicates when the next routing update will be transmitted on the interface. You set the update interval with the **vines update interval** command.
Routing updates	Indicates whether routing updates contain all entries in the routing table or just changes to the table since the last update was sent. You set the method used with the **vines update deltas** command.
Next synchronization	Indicates when the next SRTP synchronization update will be sent.
Nodes Present	Indicates the number and type of all VINES-speaking devices present on the given physical network segment.
Neighbors 0 Router2	List of all VINES neighbor on that interface and what version of the RTP protocol they are running. 0 means RTP, and 1 means SRTP.

Related Commands

vines arp-enable
vines encapsulation
vines metric
vines route-cache
vines serverless
vines split-horizon
vines update deltas
vines update interval

show vines ipc

To display information about any currently active IPC connections, use the **show vines ipc** EXEC command.

show vines ipc

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

Information about the IPC protocol formats, data sequences, and state machines can be found in Banyan documentation.

Sample Display

The following is sample output from the **show vines ipc** command:

```
Router# show vines ipc
Vines IPC Status:

Next Port: 513
Next Connection: 3
Next check in: 27 sec

Connection 2, state: connected
  Local address: Router1, id 0002, last port: 0200
  Remote address: Router2, id 0002, last port: 0001
  Last send seq: 0005, Last rcvd seq: 0005
  Next send ack: 0005, Last sent ack: 0005
  Server metric 4, last hop 0, bias 0, total 800 (ms)
  Send ACK in 0 ms, Retransmit in 0 ms
  Idle check in 0 sec
  Retransmit queue contains 0 packets
  No packet in reassembly
```

Table 15-5 describes the fields shown in the display.

Table 15-5 Show VINES IPC Field Descriptions

Field	Description
Next Port:	IPC port number that the router will use when a new, unique IPC port number is needed.
Next Connection:	IPC connection number that the router will use when a new, unique IPC connection number is needed.
Next check in:	When the router will make the next pass of the IPC connection table to examine each of the connection-specific timers.
Connection 2, state:	State of a particular connection. Possible states are connecting, connected, idle, and dead.

Field	Description
Local address:	VINES IP address of the local side of the connection.
last port:	Last port number used on this particular connection by the local host.
Remote address:	VINES IP address of the remote side of the connection.
last port:	Last port number used on this particular connection by the remote host.
Last send seq:	Last sequence number sent on this particular connection used by the local host.
Last rcvd seq:	Last sequence number received on this particular connection used by the local host.
Next send ack:	Next acknowledgment number that will be sent on this particular connection by the local host.
Last sent ack:	Last acknowledgment number that has been sent on this particular connection by the local host.
Server metric	Metric value from this host to the remote host's server or router.
last hop	Metric value from the remote host's server or router to the remote host itself. If the remote host is a server or router, this value should be zero.
bias	Bias added to the metric to account for variance in the round-trip delay of a message going to the remote host.
total	Total metric value used to reach the remote host. It is the sum of the three previous numbers.
Send ACK	Time, in seconds, until the next acknowledgment message is sent by the local host.
Retransmit	Time, in seconds, until a message is retransmitted by the local host.
Idle check in	Time, in seconds, until this connection will be checked to see if it has been idle for 30 seconds.
Retransmit queue contains ... packets	Number of messages that have been sent but not acknowledged.
No packet in reassembly	Number of packets that have been received and are being reassembled into a larger message.

show vines neighbor

To display the entries in the VINES neighbor table, use the **show vines neighbor** EXEC command.

show vines neighbor [*address* | **interface** *type number* | **server** *number*]

Syntax Description

address	(Optional) Displays the entry for the specified neighbor.
interface *type number*	(Optional) Displays all neighbor paths in the neighbor table that use the specified interface.
server *number*	(Optional) Displays all entries in the neighbor table that have the specified network number.

Command Mode

EXEC

Usage Guidelines

If no keywords or arguments are specified, all entries in the neighbor table are displayed.

Sample Displays

The following is sample output from the **show vines neighbor** command. This sample shows all entries in the VINES neighbor table.

```
Router# show vines neighbor
6 neighbors, 7 paths, version 14, next update 34 seconds

Address        Hardware Address   Type  Int    Flag Age  Metric  Uses

Router1        -                  HDLC  Se0    R0*  n/a   0230      7
Router2        -                  -     -      C1   -        -      -
Router3        0000.0c01.24ea     ARPA  Et2    R0*  42    0020      9
Router4        -                  PPP   Se1    R1   n/a   0230      0
 Router4       0000.0c01.0506     ARPA  Et0    R1.  n/a   0020      0
 Router4       0000.0c01.9ac9     VINES To0    R1*  n/a   0020      0


Router# show vines neighbor router3
3 neighbors, 4 paths, version 7, next update 24 seconds

Address          Hardware Address     Type  Int       Flag Age  Metric    Uses

Router3          0000.0c01.24ea       ARPA  Et2       R0*   42   0020        9

  RTP Counters:

    Interface Ethernet2, address Router3-Et2-0000.0c01.24ea
      Timers:
        Roll Call:  00:03:00
      Received counters:
        Requests:   00000000
        Responses:  00000000
        Updates:    00000000
        Redirects:  00000000
```

```
        Unknown:    00000000

Router# show vines neighbor router4
3 neighbors, 4 paths, version 7, next update 5 seconds

Address          Hardware Address     Type  Int       Flag Age  Metric    Uses

Router4          -                    -     -         R1     -      -        -

  SRTP Counters:

    Interface Ethernet0, address Router4-Et0-0000.0c01.0506, state up
      Origin 0001BE9A, Local 00006262, Flags 0001, ID 007F
      Timers:
        Reassembly: not active
        Retry request: not active
      Received counters:
        Requests:   specific   changes      full      null   unknown
                    00000000  00000000  00000000  00000000  00000000
        Updates:      failed      less     equal  one more   greater
          null:     00000000  00000000  00000000  00000000  00000000
          change:   00000000  00000000  00000000  00000000  00000000
          full:     00000000  00000000  00000000  00000000  00000000
          sync:     00000000  00000000  00000000  00000000  00000000
        Redirects:  00000000
        Reinits:    00000000
      Transmitted counters:
        Requests:    unknown  specific   changes      full      null
                    00000000  00000000  00000000  00000000  00000000
        Updates:    00000000
        Responses:  00000000
        Redirects:  00000000
        Reinits:    00000000
```

Table 15-6 describes the fields shown in the display.

Table 15-6 Show VINES Neighbor Field Descriptions

Field	Description
neighbors	Number of neighbors in the neighbor table.
paths	Number of paths to the neighbor.
version	Version number of the VINES neighbor table. The number is incremented each time a route or path is added to or deleted from this table.
next update	Time, in seconds, until the next routing update is sent.
Address	Address of the neighbor station. The neighbor's name is displayed if you have issued a **vines host** command.
Hardware Address	MAC address of the router interface through which the VINES neighbor in this entry can be reached.
Type	Type of MAC-level encapsulation used to communicate with this neighbor.
Int	Type and number of interface through which the VINES neighbor can be reached

Field	Description
Flag	This field is a three-column field. The first column indicates how the path was learned. It can be one of the following values: • C—Connected (that is, this is the entry for this router). • D—Learned via an RTP redirect message. • P—Placeholder. This neighbor is currently used as the next hop for a static route. • R—Learned via an RTP update message. • S—Static path entry (entered with the **vines neighbor** command). The second column indicates what version of the RTP protocol this neighbor is running. It can be one of the following values: • 0—Version 0 of the RTP protocol. This is the version used by VINES servers prior to VINES version 5.50. • 1—Version 1 of the RTP protocol, commonly called SRTP. This is the version used by VINES servers in VINES version 5.50 and later. The third column indicates how this path will be used. It can be one of the following values: • *—An asterisk means that this is the next path that will be used next when forwarding a frame to that neighbor. • .—A dot means that this is the alternate path that will be used in round-robin fashion. • Blank—No value means this is backup path that will not be used. In the sample output, there are two paths to Router4 with the same metric. These two paths will be used in a round-robin fashion, and the Token Ring path will be the next one of the two used. There is a third path to Router4 via the serial line, but this will not be used unless both of the other paths are lost.
Age	Age of this VINES neighbor table entry, in seconds. This entry will show an age of “n/a” for RTP Version 0 neighbors on WAN interfaces, when the interface has been configured for delta-only updates. In all other cases, this entry will contain a number.
Metric	Distance to this neighbor. This normally is the same as the interface metric, but may be different because of network topology or router configuration.
Uses	For all entries except placeholders, indicates the number of times that path was used to forward a packet. For placeholder entries, indicates the number of static routes that use the neighbor as the first hop.
RTP Counter:	This section shows counters that are specific to a neighbor port that is running the RTP protocol only. If the neighbor has multiple interfaces, then multiple sections will show up in this part of the display.
Interface ...	Identifies the network interface and full identifier for a neighbor port.
Timers: Roll Call	Identifies whether or not the roll call timer is active for this neighbor, and if so, when it will expire.
Received Counters	Indicates the number and type of RTP packets received from this neighbor port.

Field	Description
SRTP Counter:	This section shows counters that are specific to a neighbor port that is running the SRTP protocol. If the neighbor has multiple interfaces, then multiple sections will show up in this part of the display.
Interface	Identifies the network interface and full identifier for a neighbor port.
Timers: Reassembly	Identifies whether or not the reassembly timer is active for this neighbor, and if so, when it will expire.
Timers: Retry	Identifies whether or not the retry timer is active for this neighbor, and if so, when it will expire.
Received Counters	Indicates the number, type, and sequence number of matching SRTP packets received from this neighbor port.
Transmitted Counters	Indicated the number and type of SRTP packets transmitted explicitly to this neighbor port.

Related Commands

clear vines neighbor
clear vines route
show vines cache
vines host
vines neighbor
vines update deltas
vines update interval

show vines route

To display the contents of the VINES routing table, use the **show vines route** EXEC command.

show vines route [*number* | **neighbor** *address*]

Syntax Description

number	(Optional) Displays the routing table entry for the specified network.
neighbor *address*	(Optional) Displays all routes in the VINES routing table that have the specified neighbor as their first hop.

Command Mode

EXEC

Usage Guidelines

If no keywords or arguments are specified, all entries in the routing table are displayed.

Sample Display

The following is sample output from the **show vines route** command. This sample shows all entries in the VINES routing table.

```
Router# show vines route

Worf              Worf              R0*       2       2      0
Succubus          Succubus          R1*       2       2      0
Aloe              -                 C1        -       -      -
Vera              Vera              R0*       2       2      0
Falcon            Falcon            R0*       2       2      0
Zangbutt          Worf              R0*       2       4      0
Zangbutt          Vera              R0        2       4      0

Router# show vines route Router1
8 servers, 10 routes, version 58, next update 32 seconds

Network       Neighbor     Flags     Age      Metric   Uses    Origin    Local    Flags

Router1       Router2      R0*       n/a       0250      0     001AFE7B  00010FCA  0009
```

Table 15-7 describes the fields shown in the display.

Table 15-7 Show VINES Route Field Descriptions

Field	Description
servers	Number of servers in the routing table.
routes	Number of routes in the routing table.
version	Version number of the VINES routing table. This number is incremented each time a server or route is added to or deleted from this table.

Field	Description
next update	Time, in seconds, until the next routing update is sent.
Hash	Position of this entry in the routing table.
Network	Name or number of the remote network. Networks take the name of the server that defines the network.
Neighbor	Next hop to the destination network.
Flags	This field is a series of single-column fields. The first column indicates how the route was learned. It can be one of the following values: • C—Connected (that is, this is the entry for this router). • D—Learned via an RTP redirect message. • R—Learned via an RTP update message. • S—Static entry (entered with the **vines route** command). The second column indicates what version of the RTP protocol this router is running. It can be one of the following values: • 0—Version 0 of the RTP protocol. This is the version used by VINES servers prior to VINES version 5.50. This version number will also be shown if the route was learned via a pre-5.50 server, and thus the version information was lost. • 1—Version 1 of the RTP protocol, commonly called SRTP. This is the version used by VINES servers in VINES version 5.50 and later. An asterisk in the third column indicates that this route will be used next when forwarding a frame to that server. The fourth column indicates whether that route will be used to forward a broadcast from a serverless network. It can be one of the following values: • N—This server is considered to be the nearest server and is on a directly connected network. • n—This server is considered to be the nearest server but is not on a directly connected network. The fifth column contains the letter “S” if the route is in a suppression state. The sixth column contains the letter “h” if this path has a metric that is higher than the best metric for this neighbor. This indicates that the path is not eligible for use in load sharing.
Age	Age of this VINES routing table entry, in seconds. An age of n/a indicates the destination is accessible via a neighbor that is sending delta-only updates. Note that even though the neighbor entry for Pica has an age, there is no age available for its routing table entry or other routing entries reachable via Pica. This is because the periodic hello messages from Pica contain no routing information, only neighbor reachability information.
Metric	Distance to this server. This normally is the distance to the neighbor router plus the distance advertised by that neighbor. This does not hold for static routes.
Uses	Number of times this route has been used to forward a packet.

Field	Description
Origin	Last known timestamp that originated from this server. If this field is not valid, as indicated by the following set of flags, then it will be zero.
Local	Local timestamp then this route entry was learned or last changed.
Flags	This field is a series of bit flags presented as a hexadecimal number. The following are the defined values: • 0001—The neighbor of this server reaches it through a LAN interface. • 0002—The neighbor of this server reaches it through a WAN interface. • 0004—The neighbor of this server reaches it through a non-VINES interface. • 0008—The origin timestamp for this entry is not valid. The entry is either for a pre-5.50 server, or the entry was learned via a pre-5.50 server.

Related Commands

clear vines neighbor
clear vines route
show vines cache
vines route
vines update deltas
vines update interval

show vines service

To display information about the router's application layer support, use the **show vines service** EXEC command.

show vines service [**fs** | **nsm** | **ss** | **vs**]

Syntax Description

fs	(Optional) Displays file service information.
nsm	(Optional) Displays network and system management service information.
ss	(Optional) Displays server service information.
vs	(Optional) Displays security service information.

Command Mode

EXEC

Sample Display

The following is sample output from the **show vines service** command:

```
Router# show vines service
Vines Files Service:
    Name:     FS@Doc-ags+1@Servers  (FS)
    Ports:    Well Known 6, Transient 0
    Timer:    not running

Network & System Management Service:
    Name:     NSM@Doc-ags+1@Servers  (NSM)
    Ports:    Well Known 25, Transient 0
    Timer:    not running

Server Service:
    Name:     SS@Doc-ags+1@Servers  (SS)
    Ports:    Well Known 7, Transient 0
    Emulates: 5.50(0),  Supports: 3.22(49) - 6.99(49)
    Timer:    not running

VINES Security Service:
    Name:     VS@Doc-ags+1@Servers  (VS)
    Ports:    Well Known 19, Transient 0
    Timer:    not running
```

Table 15-8 describes the fields shown in the display.

Table 15-8 Show VINES Services Field Descriptions

Field	Description
Name:	Name of the service.
Ports:	Ports on which the service is running.
Timer:	Time at which this service will wake up and perform some periodic functions.

The following is sample output from the **show vines service** command using the **fs**, **nsm**, **ss**, and **vs** keywords:

```
Router# show vines service fs
Vines Files Service:
  Periodic timer not running.

Router# show vines service nsm
Network & System Management Service:
  Next wakeup in 00:00:29.

Router# show vines service ss
Server Service:
  Next wakeup in 00:01:51.
  Time is 17:12:55 PDT Jun 23 1994
  Time last set by Doc-ags, 0:28:09 ago.
  Time epoch is SS@Doc-ags@Servers-9, started 00:28:09 ago.
    Participating in vines time of day synchronization.
    Sending time messages to the broadcast address.
    Synchronizing vines time with system time.

Router# show vines service vs
VINES Security Service:
  Periodic timer not running.
```

Table 15-9 describes the fields shown in the displays.

Table 15-9 Show VINES Services Field Descriptions

Field	Description
Periodic timer not running	Indicates that this service has no periodic functions to perform.
Next wakeup in ...	Time, in seconds, until the service performs its periodic actions. For the Server service, this is to send a time synchronization message. For the NSM service, this is to send any requested trace packets. The periodic interval for the NSM service is 30 seconds when no trace messages are pending.
Time is ...	Current time (in the format *hours*:*minutes*:*seconds*) and date.
Time last set ...	Server that last adjusted the time, how much it adjusted the time, and how long ago it was adjusted. For times within the last 24 hours, the time format is *hours*:*minutes*:*seconds*. For times longer ago than 24 hours, the time format is *weeks*w*days*d.
Time epoch is ...	Name of the current time epoch (in the format *name-number*), and when it was established.

Related Commands

vines time access-group
vines time participate
vines time set-system
vines time use-system

show vines traffic

To display the statistics maintained about VINES protocol traffic, use the **show vines traffic** EXEC command.

show vines traffic [*type number*]

Syntax Description

type number	(Optional) Displays values for a specific interface.

Command Mode

EXEC

Usage Guidelines

If no interface is specified, values for all interfaces are displayed.

Sample Display

The following is sample output from the **show vines traffic** command:

```
Router# show vines traffic
SYSTEM TRAFFIC:
  Rcvd: 204 total, 12708 bytes, 0 format errors, 0 not enabled,
        15 local dst, 189 bcast, 0 forwarded
        0 no route, 0 zero hops
        0 checksum errors, 3 IP unknown, 0 IPC unknown
        3 bcast forwarded, 1 bcast helpered, 0 dup bcast
  Sent: 21 packets, 1278 bytes
        0 unicast, 21 bcast, 0 forwarded
        0 encap failed, 0 access failed, 0 down
        0 bcast fwd, 3 not fwd (toward source)
        0 notlan, 0 not gt4800, 0 no pp charge
   ARPv0: Rcvd 0/0/0/0/0, Sent 0/0/0/0
   ARPv1: Rcvd 0/0/0/0/0, Sent 0/0/0/0
     ICP: Rcvd 0/0/0, Send 0/0
     IPC: Rcvd 17, Sent 8
   RTPv0: Rcvd 2/10/0/0/170/0/0/5, Sent 0/6/00/0/91/10/0
   RTPv1: Rcvd 0/0/0/0/0/0, Sent 0/3/60/0
     SPP: Rcvd 0, Sent 0
    Echo: Rcvd 5, Sent 5
   Proxy: Rcvd 0, Sent 0
IPC TRAFFIC BY PORT NUMBER:
Broadcast: Other:00000000, 01:00000000, 02:00000000, 03:00000000, 04:00000000
              05:00000000, 06:00000000, 07:00000000, 08:00000000, 09:00000000
              0A:00000000, 0B:00000000, 0C:00000000, 0D:00000000, 0E:00000000
              0F:00000000, 10:00000000, 11:00000000, 12:00000000, 13:00000000
              14:00000000, 15:00000000, 16:00000000, 17:00000000, 18:00000000
              19:00000000
 Helpered: Other:00000000, 01:00000000, 02:00000000, 03:00000000, 04:00000000
              05:00000000, 06:00000000, 07:00000000, 08:00000000, 09:00000000
              0A:00000000, 0B:00000000, 0C:00000000, 0D:00000000, 0E:00000000
              0F:00000000, 10:00000000, 11:00000000, 12:00000000, 13:00000000
              14:00000000, 15:00000000, 16:00000000, 17:00000000, 18:00000000
              19:00000000
  Unicast: Other:00000000, 01:00000000, 02:00000000, 03:00000000, 04:00000000
              05:00000000, 06:00000000, 07:00000000, 08:00000000, 09:00000000
```

```
              0A:00000000, 0B:00000000, 0C:00000000, 0D:00000000, 0E:00000000
              0F:00000000, 10:00000000, 11:00000000, 12:00000000, 13:00000000
              14:00000000, 15:00000000, 16:00000000, 17:00000000, 18:00000000
              19:00000000
 Proxied: Other:00000000, 01:00000000, 02:00000000, 03:00000000, 04:00000000
              05:00000000, 06:00000000, 07:00000000, 08:00000000, 09:00000000
              0A:00000000, 0B:00000000, 0C:00000000, 0D:00000000, 0E:00000000
              0F:00000000, 10:00000000, 11:00000000, 12:00000000, 13:00000000
              14:00000000, 15:00000000, 16:00000000, 17:00000000, 18:00000000
              19:00000000
P_Replies: Other:00000000, 01:00000000, 02:00000000, 03:00000000, 04:00000000
              05:00000000, 06:00000000, 07:00000000, 08:00000000, 09:00000000
              0A:00000000, 0B:00000000, 0C:00000000, 0D:00000000, 0E:00000000
              0F:00000000, 10:00000000, 11:00000000, 12:00000000, 13:00000000
              14:00000000, 15:00000000, 16:00000000, 17:00000000, 18:00000000
              19:00000000

Interface Hssi0:
  Rcvd: 0 packets, 0 bytes, 0 format errors, 0 not enabled,
        0 local dst, 0 bcast, 0 forwarded,
        0 no route, 0 zero hops
        0 checksum errors, 0 IP unknown, 0 IPX unknown
        0 bcast forwarded, 0 bcast helpered, 0 dup bcast
  Sent: 0 packets, 0 bytes
        0 unicast, 0 bcast, 0 forwarded
        0 encap failed, 0 access failed, 0 down
        0 bcast fwd, 0 not fwd (toward source)
        0 notlan, 0 not gt4800, 0 no pp charge
   ARPv0: Rcvd 0/0/0/0/0, Sent 0/0/0/0
   ARPv1: Rcvd 0/0/0/0/0, Sent 0/0/0/0
     ICP: Rcvd 0/0/0, Send 0/0
     IPC: Rcvd 0, Sent 8
   RTPv0: Rcvd 0/10/0/0/0/0/0/0, Sent 0/0/00/0/0/0/0
   RTPv1: Rcvd 0/0/0/0/0/0, Sent 0/3/60/0
     SPP: Rcvd 0, Sent 0
    Echo: Rcvd 0, Sent 0
   Proxy: Rcvd 0, Sent 0
```

Table 15-10 describes the fields shown in the display.

Table 15-10 Show VINES Traffic Field Descriptions

Field	Description
SYSTEM TRAFFIC:	This section displays statistics about all VINES packets handled by the router.
Rcvd:	This section displays statistics about VINES packets received by the router.
total packets	Total number of VINES packets received.
bytes	Total bytes in all the VINES packets received.
format errors	Number of VINES packets that had errors in the format of the VINES IP header. Currently, the only thing checked is the length field in the header. The number of packets with format errors is included in the count of total packets received (in the Rcvd: field).
not enabled	Number of VINES packets received on an interface on which VINES was not enabled. These packets are not included when counting the total packets received (in the Rcvd: field).

Field	Description
local dst	Number of packets accepted for further processing because they were addressed to the router's unicast address.
bcast	Number of packets accepted for further processing because they were addressed to the router's broadcast address.
forwarded	Number of packets not accepted for further processing but that were simply forwarded out another interface.
no route	Number of packets discarded because the router did not know how to reach the destination.
zero hops	Number of packets discarded because the hop count field in the VINES IP header was zero.
checksum errors	Number of packets accepted by the router for further processing (the sum of the "local dest" and "bcast" fields) that were discarded because the checksum was bad.
IP unknown	Number of packets accepted by the router (the sum of the "local dest" and "bcast" fields) that were discarded because the IP protocol type was unknown.
IPC unknown	Number of packets accepted by the router for further processing (the sum of the "local dest" and "bcast" fields) that were discarded because the IPC port number was unknown.
bcast forwarded	Number of broadcast packets accepted by the router for further processing (as shown in the "bcast" field) that were forwarded because they had a nonzero hop count. (Note that the sum of the "bcast forwarded," "bcast helpered," and "dup bcast" fields will not equal the total number of broadcast packets received.)
bcast helpered	Number of broadcast packets accepted by the router (as shown in the "bcast" field) that were "helpered" to a Banyan server. (Note that the sum of the "bcast forwarded," "bcast helpered," and "dup bcast" fields will not equal the total number of broadcast packets received.)
dup bcast	Number of broadcast packets accepted by the router (as shown in the "bcast" field) that were classified as duplicates and discarded. (Note that the sum of the "bcast forwarded," "bcast helpered," and "dup bcast" fields will not equal the total number of broadcast packets received.)
Sent:	This section displays statistics about VINES packets sent by the router.
packets	Total number of VINES packets sent.
bytes	Total bytes in all the VINES packets sent.
unicast	Number of unicast packets originating at the router.
bcast	Number of broadcast packets originating at the router.
forwarded	Number of unicast packets that were forwarded from another interface.
encap failed	Number of packets not sent because of an encapsulation failure. This usually happens when entries in a map for a public data network, such as X.25 or Frame Relay, are missing.
access failed	Number of packets not sent because the destination was denied by an access list.
down	Number of packets not sent because the interface was down.

Field	Description
bcast fwd	Number of broadcast packets that were forwarded from another interface.
not fwd (toward source)	Number of broadcast packets that were not forwarded because this interface is the interface on which the broadcast was received.
not lan	Number of broadcast packets that were not forwarded because they were marked for LANs only and this interface is not a LAN (for example, it might be a serial interface.)
not gt	Number of broadcast packets that were not forwarded because they were marked for high-speed interfaces only and this interface is a low-speed interface (line speed of 4800 baud or less).
no pp charge	Number of broadcast packets that were not forwarded because they were marked to send only to networks that do not have per-packet charging and this interface is to a network that has per-packet charging.
ARPv0:	This section displays statistics about VINES ARP packets sent and received by the router.
Rcvd *x*/*x*/*x*/*x*/*x*	Number of ARP packets received of type 0, 1, 2, 3, and other.
Sent *x*/*x*/*x*/*x*	Number of ARP packets sent of type 0, 1, 2, and 3.
ARPv1:	This section displays statistics about VINES SARP packets sent and received by the router.
Rcvd *x*/*x*/*x*/*x*/*x*	Number of SARP packets received of type 0, 1, 2, 3, and other.
Sent *x*/*x*/*x*/*x*	Number of SARP packets sent of type 0, 1, 2, and 3.
ICP:	This section displays statistics about VINES ICP packets sent and received by the router.
Rcvd *x*/*x*/*x*	Number of ICP packets received of type 0, 1, and other.
Sent *x*/*x*	Number of ICP packets sent of type 0 and 1.
IPC:	This section displays statistics about VINES IPC packets sent and received by the router.
Rcvd	Number of IPC packets received.
Sent	Number of IPC packets sent.
RTPv0:	This section displays statistics about VINES routing protocol (RTP) packets sent and received by the router.
Rcvd *x*/*x*/*x*/*x*/*x*/*x*/*x*	Number of RTP packets received of type 0, 1, 2, 3, 4, 5, 6, and other. The counts of type 0, type 2, type 3, and other RTP packets should always be zero.
Sent *x*/*x*/*x*/*x*/*x*/*x*/*x*	Number of RTP packets sent of type 0, 1, 2, 3, 4, 5, and 6.
RTPv0:	This section displays statistics about VINES routing protocol (RTP) packets sent and received by the router.
Rcvd *x*/*x*/*x*/*x*/*x*	Number of SRTP packets received of type 0, 1, 2, 3, and other. The count of other SRTP packets should always be zero.
Sent *x*/*x*/*x*/*x*/*x*	Number of SRTP packets sent of type 0, 1, 2, 3.
SPP:	This section displays statistics about VINES SPP packets sent and received by the router.
Rcvd	Number of SPP packets received.
Sent	Number of SPP packets sent.

Field	Description
Echo:	This section displays statistics about VINES echo packets sent and received by the router.
Rcvd	Number of MAC-level echo packets received.
Sent	Number of MAC-level echo packets sent.
Proxy:	This section displays statistics about VINES proxies sent and received by the router. A proxy is when a client sends a query directly to the router for which the router does not have the intelligence to respond. The router then sends these queries to a Banyan server, and when it receives the response from the server, the router relays it back to the client.
Rcvd	Number of proxy queries received by the router.
Sent	Number of proxy queries sent by the router.
IPC TRAFFIC BY PORT NUMBER:	This section displays statistics about VINES Interprocess Communications Protocol (IPC) packets. The information displayed in this section is particularly useful when a serverless network is connected to the router.
Broadcast:	Number of VINES IPC messages, by destination port number, received by the router because they were addressed to the VINES IP broadcast address.
Helpered:	Number of broadcast messages that were sent toward a Banyan server because they were received on an interface for a serverless network.
Unicast:	Number of VINES IPC messages, by destination port number, received by the router because they were specifically addressed to the VINES IP address of the router.
Proxied:	Number of unicast messages received that were sent to a Banyan server because they were received on a serverless interface and because the router did not know how to respond to the message.
P_Replies:	Number of responses to a proxy query that were received from a Banyan server.
Interface	This section displays statistics about the individual interfaces in the router. The fields in this section have the same meanings as the fields of the same name in the “SYSTEM TRAFFIC” section, except that the statistics are for the particular interface, not for the entire router.

Related Commands

clear vines traffic
vines serverless

trace

To determine the path that a packet takes when traversing a VINES network, use the **trace** EXEC command.

trace [**vines** | **oldvines**] [*address*]

Syntax Description

vines	(Optional) Specifies the VINES protocol. This trace is compatible with the Banyan VINES traceroute function.
oldvines	(Optional) Specifies the VINES protocol. This trace is compatible with our **trace** function prior to IOS Release 10.2.
address	(Optional) Address of a node. This is a 6-byte hexadecimal number in the format *network:host*, where *network* is 4 bytes and *host* is 2 bytes.

Command Mode

EXEC

Usage Guidelines

The **trace** EXEC command supports the Banyan traceroute function. This enables trace requests on a VINES network to reach all servers on the network.

This command does not produce the names of any VINES servers that are traversed.

Table 15-11 explains the **trace** test characters when you specify the **oldvines** keyword.

Table 15-11 Trace Test Characters

Character	Meaning
nn msec	For each node, the round-trip time for each probe in milliseconds.
*	The probe timed out.
?	Unknown packet type.

Sample Displays

The following is sample output from the VINES **trace** command when you specify the **vines** keyword:

```
Router# trace
Protocol [ip]: vines
Target Vines address: wayfinder
Source Vines address: coinspinner
From: 0002801578 Coinspinner        To: 0002609380 Wayfinder

Server                    Gate                        metric media address
0002801578 Coinspinner    0805371606 Router               4     40 000030C0FEB6
0805371606 Router         0002609380 Wayfinder            2   2560 10005A746A36
```

The following is sample output from the VINES **trace** command when you specify the **oldvines** keyword:

```
Router# trace
Protocol [ip]: oldvines
Target vines address: 27AF92:1
Numeric display [n]:
Timeout in seconds [3]:
Probe count [3]:
Minimum Time to Live [0]:
Maximum Time to Live [15]:
Type escape sequence to abort.
Tracing the route to COINSPINNER (27AF92:1)
 0 Farslayer (30002A2D:1) 0 msec 4 msec 4 msec
 1 Coinspinner (27AF92:1) 4 msec 4 msec 8 msec
```

vines access-group

To apply an access list to an interface, use the **vines access-group** interface configuration command. To remove the access list, use the **no** form of this command.

vines access-group *access-list-number*
no vines access-group *access-list-number*

Syntax Description

access-list-number	Number of the access list. All outgoing packets defined with either standard or extended access lists and forwarded through the interface are filtered by the entries in this access list. For standard access lists, *access-list-number* is a decimal number from 1 to 100. For extended access lists, *access-list-number* is a decimal number from 101 to 200.

Default

No access list is applied.

Command Mode

Interface configuration

Usage Guidelines

The **vines access-group** command applies an access list created with the **vines access-list** command to an interface.

You can apply only one access list to an interface.

Example

In the following example, access list 1 is applied to Ethernet interface 0:

```
interface ethernet 0
vines access-group 1
```

Related Commands

vines access-list (standard)
vines access-list (extended)

vines access-list (standard)

To specify a standard VINES access list, use this version of the **vines access-list** global configuration command. To remove the access list, use the **no** form of this command.

vines access-list *access-list-number* **{deny | permit}** *protocol source-address source-mask [source-port] destination-address destination-mask [destination-port]*
no vines access-list *access-list-number*

Syntax Description

access-list-number	Number of the access list. This is a decimal number from 1 to 100.
deny	Denies access if the conditions are matched.
permit	Allows access if the conditions are matched.
protocol	VINES protocol ID number or name. It can be a value from 1 to 255 or one of the following protocol keywords: • **arp**—Address Resolution Protocol • **icp**—Internet Control Protocol • **ip**—VINES Internet Protocol • **ipc**—Interprocess Communications • **rtp**—Routing Update Protocol • **spp**—Sequence Packets Protocol
source-address	Address of the network from which the packet is being sent. This is a 6-byte hexadecimal number in the format *network:host*, where *network* is 4 bytes and *host* is 2 bytes.
source-mask	Mask to be applied to *source-address*. This is a 6-byte hexadecimal value. Place ones in the bit positions you want to mask. These bits correspond to the bit in the address that should be ignored.
source-port	(Optional) Number of the local port from which the packet is being sent. This argument is required when the protocol specified is IPC or SPP, and is not accepted when any other protocol is specified. It can be a number from 0x0000 through 0xFFFF. Well-known local port numbers have values from 0x0001 through 0x01FF. Transient local port numbers have values from 0x0200 through 0xFFFE. Table 15-12 in the "Usage Guidelines" section lists some IPC port numbers.
destination-address	Address of the network to which the packet is being sent. This is a 6-byte hexadecimal number in the format *network:host*, where *network* is 4 bytes and *host* is 2 bytes.

destination-mask	Mask to be applied to *destination-address*. This is a 6-byte hexadecimal value. Place ones in the bit positions you want to mask. These bits correspond to the bits in the address that should be ignored.
destination-port	(Optional) Number of the local port to which the packet is being sent. This argument is required when the protocol specified is IPC or SPP, and is not accepted when any other protocol is specified. It can be a number from 0x0000 through 0xFFFF. Well-known local port numbers have values from 0x0001 through 0x01FF. Transient local port numbers have values from 0x0200 through 0xFFFE. Table 15-12 in the "Usage Guidelines" section following lists some IPC port numbers.

Default

No standard VINES access list is specified.

Command Mode

Global configuration

Usage Guidelines

A standard VINES access list filters packets based on their protocol, source and destination addresses, and source and destination address masks, and optionally on their source and destination ports.

Use the **vines access-group** command to apply an access list to an interface.

Keep the following in mind when configuring VINES network access control:

- You can apply only one access list to an interface.
- The conditions in the access list are applied to all outgoing packets that are forwarded by the router. Packets generated by the router are not subject to the access list.
- Access list entries are scanned in the order you enter them. The first matching entry is used.
- An implicit *deny everything* entry is defined at the end of an access list unless you include an explicit *permit everything* entry at the end of the list.
- All new entries to an existing list are placed at the end of the list. You cannot add an entry to the middle of a list. This means that if you have previously included an explicit *permit everything* entry, new entries will never be scanned. The solution is to delete the access list and retype it with the new entries.

If you specify a protocol type of IPC, the port (either *source-port* or *destination-port*) can be one of the values shown in Table 15-12.

Table 15-12 Some VINES IPC Port Numbers

IPC Port Number (Hexadecimal)	Service
0x0003	Back End (only on PCs; it is the 25th line notification)
0x0004	Mail Service
0x0006	"VINES Files" File Service
0x0007	Server Service
0x000F	StreetTalk Service
0x0012	Network Management
0x0013	VINES Security
0x0016	StreetTalk Directory Assistance
0x0017	StreetTalk Directory Assistance Service Listening Port
0x0019	Systems and Network Management

Examples

In the following example, the first line prohibits any communication on StreetTalk port (port number 0xF); the second line permits all other communication:

```
vines access-list 1 deny   IPC 0:0 ffffffff:ffff 0xf 0:0 ffffffff:ffff 0xf
vines access-list 1 permit IP 0:0 ffffffff:ffff     0:0 ffffffff:ffff
```

The following example filters all mail service on Ethernet interface 0 and permits all other traffic:

```
interface Ethernet 0
vines access-group 101
!
vines access-list 101 deny ipc 0:0 FFFFFFFF:FFFF 4 0 0:0 FFFFFFFF:FFFF 0 0xF FFF
vines access-list 101 permit ip 0:0 FFFFFFFF:FFFF 0:0 FFFFFFFF:FFFF
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

priority-list protocol †
show vines access
vines access-group
vines access-list (extended)
vines access-list (simple)

vines access-list (extended)

To create an extended VINES access list, use this version of the **vines access-list** global configuration command. To remove an extended access list, use the **no** form of this command.

vines access-list *access-list-number* {**deny** | **permit**} *protocol source-address source-mask* [*source-port source-port-mask*] *destination-address destination-mask* [*destination-port destination-port-mask*]
no vines access-list *access-list-number*

Syntax Description

access-list-number	Number of the access list. This is a decimal number from 101 to 200.
deny	Denies access if the conditions are matched.
permit	Allows access if the conditions are matched.
protocol	VINES protocol ID number or name. The number can be a value from 1 to 255 or one of the following protocol keywords: • **arp**—Address Resolution Protocol • **icp**—Internet Control Protocol • **ip**—VINES Internet Protocol • **ipc**—Interprocess Communications • **rtp**—Routing Update Protocol • **spp**—Sequence Packets Protocol **arp**—Address Resolution Protocol • **icp**—Internet Control Protocol • **ip**—VINES Internet Protocol • **ipc**—Interprocess Communications • **rtp**—Routing Update Protocol • **spp**—Sequence Packets Protocol
source-address	Address of the network from which the packet is being sent. This is a 6-byte hexadecimal number in the format *network:host*, where *network* is 4 bytes and *host* is 2 bytes.
source-mask	Mask to be applied to *source-address*. This is a 6-byte hexadecimal value. Place ones in the bit positions you want to mask. These bits correspond to the bits in the address that should be ignored.

source-port	Number of the local port from which the packet is being sent. This argument is required when the protocol specified is IPC or SPP, and is not accepted when any other protocol is specified. It can be a number from 0x0000 through 0xFFFF. Well-known local port numbers have values from 0x0001 through 0x01FF. Transient local port numbers have values from 0x0200 through 0xFFFE. Table 15-13 in the "Usage Guidelines" section lists some IPC port numbers.
source-port-mask	(Optional) Mask to be applied to *source-port*. This argument is required when the protocol specified is IPC or SPP, and is not accepted when any other protocol is specified. It can be a number from 0x0000 through 0xFFFF. These bits correspond to the bits in the port that should be ignored.
destination-address	VINES address of the network to which the packet is being sent. This is a 6-byte hexadecimal number in the format *network:host*, where *network* is 4 bytes and *host* is 2 bytes.
destination-mask	Mask to be applied to *destination-address*. This is a 6-byte hexadecimal value. Place ones in the bit positions you want to mask. These bits correspond to the bits in the address that should be ignored.
destination-port	Number of the local port to which the packet is being sent. This argument is required when the protocol specified is IPC or SPP, and is not accepted when any other protocol is specified. It can be a number from 0x0000 through 0xFFFF. Well-known local port numbers have values from 0x0001 through 0x01FF. Transient local port numbers have values from 0x0200 through 0xFFFE. Table 15-13 in the "Usage Guidelines" section lists some IPC port numbers.
destination-port-mask	(Optional) Mask to be applied to *destination-port*. This argument is required when the protocol specified is IPC or SPP, and is not accepted when any other protocol is specified. It can be a number from 0x0000 through 0xFFFF. These bits correspond to the bits in the port that should be ignored.

Default

No extended VINES access list is specified.

Command Mode

Global configuration

Usage Guidelines

An extended VINES access list filters packets based on their protocol, source and destination addresses, and source and destination address masks, and optionally on their source and destination ports, and source and destination port masks. This differs from the standard access list filters in that you can specify port masks.

Use the **vines access-group** command to assign an access list to an interface.

Keep the following in mind when configuring VINES network access control:

- You can apply only one access list to an interface.
- The conditions in the access list are applied to all outgoing packets that are forwarded by the router. Packets generated by the router are not subject to the access list.
- Access list entries are scanned in the order you enter them. The first matching entry is used.
- An implicit *deny everything* entry is defined at the end of an access list unless you include an explicit *permit everything* entry at the end of the list.
- All new entries to an existing list are placed at the end of the list. You cannot add an entry to the middle of a list. This means that if you have previously included an explicit *permit everything* entry, new entries will never be scanned. The solution is to delete the access list and retype it with the new entries.

If you specify a protocol type of IPC, the port (either *source-port* or *destination-port*) can be one of the values shown in Table 15-13.

Table 15-13 Some VINES IPC Port Numbers

IPC Port Number (Hexadecimal)	Service
0x0003	Back End (only on PCs; it is the 25th line notification)
0x0004	Mail Service
0x0006	“VINES Files” File Service
0x0007	Server Service
0x000F	StreetTalk Service
0x0012	Network Management
0x0013	VINES Security
0x0016	StreetTalk Directory Assistance
0x0013	StreetTalk Directory Assistance Service Listening Port
0x0019	Systems and Network Management

Example

In the following example, the first line prohibits communication from any client process to the service on IPC port 0x14; the second line permits all other communication:

```
vines access-list 101 deny   IPC 0:0 ffffffff:ffff 0x14 0 0:0 ffffffff:ffff 0 0xFFFF
vines access-list 101 permit IP 0:0 ffffffff:ffff     0:0 ffffffff:ffff
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

priority-list protocol †
show vines access
vines access-group
vines access-list (extended)
vines access-list (simple)

vines access-list (simple)

To create a simple VINES access list, use this version of the **vines access-list** global configuration command. To remove a simple access list, use the **no** form of this command.

vines access-list *access-list-number* {**deny** | **permit**} *source-address source-mask*
no vines access-list *access-list-number*

Syntax Description

access-list-number	Access list number. It is a number from 201 to 300.
deny	Denies access if the conditions are matched.
permit	Allows access if the conditions are matched.
source-address	Address of the network from which the packet is being sent. This is a 6-byte hexadecimal number in the format *network:host*, where *network* is 4 bytes and *host* is 2 bytes.
source-mask	Mask to be applied to *source-address*. This is a 6-byte hexadecimal value. Place ones in the bit positions you want to mask. These bits correspond to the bits in the address that should be ignored.

Default

No simple VINES access list is specified.

Command Mode

Global configuration

Usage Guidelines

A simple VINES access list filters packets based on their source address and source address mask. These access lists are used to decide which stations to accept time updates from.

Use the **vines access-group** command to assign an access list to an interface.

Keep the following in mind when configuring VINES network access control:

- You can assign only one access list to an interface.
- The conditions in the access list are applied to all outgoing packets that are forwarded by the router. Packets generated by the router are not subject to the access list.
- Access list entries are scanned in the order you enter them. The first matching entry is used.
- An implicit *deny everything* entry is defined at the end of an access list unless you include an explicit *permit everything* entry at the end of the list.
- All new entries to an existing list are placed at the end of the list. You cannot add an entry to the middle of a list. This means that if you have previously included an explicit *permit everything* entry, new entries will never be scanned. The solution is to delete the access list and retype it with the new entries.

Example

The following example defines an access list that accept time updates only from the stations on networks 30015800 and 30004355; it defines time updates from all other sources:

```
vines access-list 201 permit 30015800:0001 00000000:0000
vines access-list 201 permit 30004355:0001 00000000:0000
vines access-list 201 deny 00000000:0000 FFFFFFFF:FFFF
interface ethernet 0
vines access-group 201
```

Related Commands

show vines access
vines access-group
vines access-list (standard)
vines access-list (simple)
vines time access-group
vines time participate
vines time set-system
vines time use-system

vines arp-enable

To enable the processing of ARP packets, use the **vines arp-enable** interface configuration command. To disable the processing of ARP packets, use the **no** form of this command.

vines arp-enable [**dynamic**]
no vines arp-enable [**dynamic**]

Syntax Description

dynamic	(Optional) Responds to ARP and SARP requests on this interface only if there are no other VINES servers present.

Default

The interface always responds to ARP and SARP requests.

Command Mode

Interface configuration

Usage Guidelines

Client systems on VINES networks are assigned network addresses dynamically. When a VINES client boots, it has no knowledge of their addresses and preferred servers. Immediately after it initializes its hardware interface, the client sends broadcast requests asking a server to provide it with a network-layer address. In a network that has a server, our routers do not normally respond to these broadcast requests. However, on a network that has only clients and no servers (called a serverless network), the router does need to respond to the broadcast requests so that all the clients on that serverless network can acquire network addresses. By default, the router will respond to ARP requests and assign addresses to network clients only if there is no VINES server present on that network segment. When it does, the router then acts as a network communication service provider for the client. You may configure the router to respond to these requests even if a VINES servers is present, or never to respond to these requests. If the router assigns an address, it will generate a unique network number based on its own VINES address.

A VINES file server must still be present somewhere on the network in order for the client to continue the booting process.

Example

The following example configures a router when Ethernet interface 1 is a network that does not contain any VINES servers:

```
interface ethernet 0
vines metric 2
!
interface ethernet 1
vines metric 2
```

The following example configures a router to always provide ARP service on Ethernet interface 1, even when VINES servers are present on that network:

```
interface ethernet 0
vines metric 2
!
interface ethernet 1
vines metric 2
vines arp-enable
```

Related Command

vines propagate
vines serverless

vines decimal

To display VINES addresses in decimal notation, use the **vines decimal** global configuration command. To return to displaying the addresses in hexadecimal, use the **no** form of this command.

vines decimal
no vines decimal

Syntax Description

This command has no arguments or keywords.

Default

Addresses are displayed in hexadecimal.

Command Mode

Global configuration

Usage Guidelines

When displaying addresses, the router always uses a name if one has been configured via the **vines host** command. The **vines decimal** command affects the radix in which the address is presented when a name is not available.

Example

The following example displays VINES addresses in decimal:

vines decimal

Related Commands

clear vines cache
clear vines neighbor
clear vines route
show vines cache
vines host

vines encapsulation

To set the MAC-level encapsulation used for VINES broadcast packets, use the **vines encapsulation** interface configuration command. To disable encapsulation, use the **no** form of this command.

vines encapsulation [**arpa** | **snap** | **vines-tr**]
no vines encapsulation

Syntax Description

arpa	(Optional) ARPA encapsulation. This is the default encapsulation for Ethernet interfaces.
snap	(Optional) SNAP encapsulation. This encapsulation uses an IEEE 802.2 SNAP header. It is the default encapsulation for all media except Ethernet and Token Ring.
vines-tr	(Optional) Our VINES Token Ring encapsulation. This is the default encapsulation for Token Ring interfaces.

Default

ARPA encapsulation for Ethernet
VINES Token Ring encapsulation for Token Ring
SNAP encapsulation for all other media

Command Mode

Interface configuration

Usage Guidelines

You can choose a MAC-level encapsulation type for each Ethernet, Token Ring, or IEEE 802.2 interface.

Setting the MAC-level encapsulation type with the **vines encapsulation** command affects broadcast packets sent by the router. The router keeps track of which encapsulation is used by each of its neighbors and uses the same style of encapsulation when talking directly to a neighbor.

You should not use this command with the current versions of VINES software that are available. This command is present for future interoperability when Banyan begins using encapsulations other than the current default ones.

Example

The following example configures IEEE 802.2 SNAP encapsulation on Ethernet interface 0:

```
vines routing
!
interface ethernet 0
vines metric 2
vines encapsulation snap
```

vines host

To associate a host name with a VINES address, use the **vines host** global configuration command. To delete the association, use the **no** form of this command.

vines host *name address*
no vines host *name*

Syntax Description

name	VINES host name. It can be any length and sequence of characters separated by white space.
address	Number of a VINES network. You enter it in the current VINES radix, in the format *network*:*host*, where *network* is 4 bytes and *host* is 2 bytes.

Default

Hosts are displayed by address.

Command Mode

Global configuration

Usage Guidelines

The router maintains a table of the mappings between host names and addresses.

When displaying addresses, the router uses the name instead of the numerical address if you have configured one with the **vines host** command.

Our software provides only static name-to-address bindings for the VINES protocol. This is completely separate from Banyan's distributed naming system, StreetTalk. The router does not learn names from StreetTalk, nor does the router provide names to StreetTalk.

Example

The following example assigns names to four VINES servers:

```
! cisco names
vines host FARSLAYER 30002A2D:0001
vines host DOOMGIVER 30000A83:0001
! VINES PS/2 server
vines host COINSPINNER 0027AF92:0001
! PC clone client
vines host STUFF 0027AF92:8001
```

Related Commands

clear vines neighbor
clear vines route
show vines host
vines decimal

vines input-network-filter

To filter the information contained in routing messages received from other stations, use the **vines input-network-filter** interface configuration command. To disable this filtering, use the **no** form of this command.

vines input-network-filter *access-list-number*
no vines input-network-filter

Syntax Description

access-list-number	Number of the access list. It is a decimal number from 201 to 300.

Default

No filtering.

Command Mode

Interface configuration

Usage Guidelines

VINES routing messages contain topological entries that allow service and client nodes to select the best paths to destinations. This command provides filtering ability to an administrator so that they may selectively determine which routing entries should be accepted from other routers and which routing entries should be dropped. This command may be useful in enforcing administrative policies of local server usage.

Example

The following example prevents a route to one specific server from ever being learned via interface Ethernet 0:

```
vines routing
!
vines access-list 201 deny 27AF9A:1 0:0
vines access-list 201 permit 0:0 FFFFFFFF:FFFF
!
interface ethernet 0
vines metric 2
vines input-network-filter 201
```

vines input-router-filter

To filter received routing messages based upon the address of the sending station, use the **vines input-router-filter** interface configuration command. To disable this filtering, use the **no** form of this command.

vines input-router-filter *access-list-number*
no vines input-router-filter

Syntax Description

access-list-number	Number of the access list. It is a decimal number from 201 to 300.

Default

No filtering.

Command Mode

Interface configuration

Usage Guidelines

VINES routing messages contain topological entries that allow service and client nodes to select the best paths to destinations. This command provides filtering ability to an administrator so that they may selectively determine the routers from which routing entries will be accepted.

Example

The following example prevents the router from ever learning routing information from one specific server on interface Ethernet 0:

```
vines routing
!
vines access-list 201 deny 27AF9A:1 0:0
vines access-list 201 permit 0:0 FFFFFFFF:FFFF
!
interface ethernet 0
vines metric 2
vines input-router-filter 201
```

vines metric

To enable VINES routing on an interface, use the **vines metric** interface configuration command. To disable VINES routing, use the **no** form of this command.

vines metric [*whole* [*fractional*]]
no vines metric

Syntax Description

whole	(Optional) Integer cost value associated with the interface. It is optional for all interface types. If you omit *whole*, the router automatically chooses a reasonable value. These values are listed in Table 15-14 in the "Usage Guidelines" section. For additional information, refer to the discussion and table in the "Usage Guidelines" section. If whole is zero, then a fractional portion must be supplied.
fractional	(Optional) Fractional cost value associated with the interface expressed in 10,000ths. It is optional for all interface types, but may only be present if a whole number portion is specified. This number will be rounded to the nearest 1/16. If you omit both whole and fractional numbers, the router automatically chooses a reasonable value. These values are listed in Table 15-14 in the "Usage Guidelines" section. For additional information, refer to the discussion and table in the "Usage Guidelines" section.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

The metric is the cost value associated with the interface media type. It is generally inversely proportional to the speed of the interface. The lower the delay metric, the more like it is that the router will use that interface.

Our router automatically chooses a reasonable metric. These numbers match as closely as possible the numbers a Banyan server would choose for an interface of the same type and speed.

When enabling VINES for a serial interface, you should keep in mind that the VINES metric is based upon the configured bandwidth for the interface. To insure that the router selects the correct VINES metric, you need to make sure that the correct bandwidth has been configured. To do this, first issue the **show interface** command to determine the speed of the interface. Then issue the **bandwidth** command to set the bandwidth rate that is appropriate for that interface type and speed.After that, issue the **vines metric** command and the router will choose a metric appropriate to that speed. If you do not issue the **bandwidth** command first, you will need to either reissue the **vines metric** command or issue it with a metric number to get an appropriate metric.

Banyan servers use these metrics to compute timeouts when communicating with other hosts. If you do specify a metric, be careful that you do not set this number too high or too low. Doing so could disrupt the normal function of the Banyan servers.

Table 15-14 lists some example delay metric values.

Table 15-14 Example Delay Metric Values

Interface Type	Old Format	New Internal Format	New Configuration File Format	Seconds
FDDI	1	0010	1 0000	0.2000
Ethernet	2	0020	2 0000	0.4000
16-Mb Token Ring	2	0020	2 0000	0.4000
4-Mb Token Ring	4	0040	4 0000	0.8000
T1 HDLC	35	0230	35 0000	7.0000
56-kb HDLC	45	02D0	45 0000	9.0000
9600 baud HDLC	90	05A0	90 0000	18.0000
4800 baud HDLC	150	0960	150 0000	30.0000
2400 baud HDLC	250	0F00	250 0000	50.0000
1200 baud HDLC	450	1C20	450 0000	90.0000
T1 X.25	45	02D0	45 0000	9.0000
56-kb X.25	55	0370	55 0000	11.0000
9600 baud X.25	100	0640	100 0000	20.0000
4800 baud X.25	160	0A00	160 0000	32.0000
2400 baud X.25	260	1040	260 0000	52.0000
1200 baud X.25	460	1CC0	460 0000	92.0000

Examples

The following example enables VINES routing on Ethernet interface 0 and sets the metric to 2:

```
vines routing
!
interface ethernet 0
vines metric 2
```

The following example enables VINES routing on FDDI interface 0 and sets the metric to 0.25:

```
vines routing
!
interface fddi 0
vines metric 0 2500
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

bandwidth †
vines routing
vines update deltas
vines update interval

vines neighbor

To specify a static path to a neighbor station, use the **vines neighbor** interface configuration command. To remove a static path from the neighbor table, use the **no** form of this command.

vines neighbor *address mac-address encapsulation* [*whole* [f*ractional*]]
no vines neighbor *address mac-address*

Syntax Description

address	VINES IP address of the station to which to add or remove a static path.
mac-address	MAC-level address used to reach the neighbor station.
encapsulation	Encapsulation type to use on the media. It can be one of the following values: • **arpa**—Use ARPA encapsulation. This is recommended for Ethernet interfaces. • **snap**—Use an IEEE 802.2 SNAP header. This is recommended for FDDI interfaces. • **vines-tr**—Use our VINES Token Ring encapsulation. This is recommended for Token Ring interfaces.
whole	(Optional) Delay metric to use on the neighbor. If you omit this argument, the metric used is that specified with the **vines metric** command for the selected interface.
fractional	(Optional) Fractional metric value associated with this neighbor. This number will be rounded to the nearest 1/16. If you omit both whole and fractional numbers, then the interface metric will be used.

Default

No static paths are specified.

Command Mode

Interface configuration

Usage Guidelines

You can configure static neighbor entries only on Ethernet, FDDI, and Token Ring interfaces.

The decision to use a static path or a dynamic path is always determined by the relative metric numbers.

Be careful when assigning static paths. If a static path is assigned with a better metric than the dynamic paths and the link associated with the static path is lost, traffic may stop being forwarded, even though an alternative path might be available.

The metric is the cost value associated with the interface media type. It is generally inversely proportional to the speed of the interface. The lower the delay metric, the more like it is that the router will use that interface.

This command is useful for testing VINES networks with test equipment that does not generate hello packets.

Table 15-15 lists some example delay metric values.

Table 15-15 Example Delay Metric Values

Interface Type	Old Format	New Internal Format	New Configuration File Format	Seconds
FDDI`	1	0010	1 0000	0.2000
Ethernet	2	0020	2 0000	0.4000
16-Mb Token Ring	2	0020	2 0000	0.4000
4-Mb Token Ring	4	0040	4 0000	0.8000
T1 HDLC	35	0230	35 0000	7.0000
56-kb HDLC	45	02D0	45 0000	9.0000
9600 baud HDLC	90	05A0	90 0000	18.0000
4800 baud HDLC	150	0960	150 0000	30.0000
2400 baud HDLC	250	0F00	250 0000	50.0000
1200 baud HDLC	450	1C20	450 0000	90.0000
T1 X.25	45	02D0	45 0000	9.0000
56-kb X.25	55	0370	55 0000	11.0000
9600 baud X.25	100	0640	100 0000	20.0000
4800 baud X.25	160	0A00	160 0000	32.0000
2400 baud X.25	260	1040	260 0000	52.0000
1200 baud X.25	460	1CC0	460 0000	92.0000

Example

The following example defines a static path to the neighbor station at address 12345678:0001 using ARPA encapsulation:

```
interface ethernet 0
vines neighbor 12345678:0001 0001.0002.0003 arpa 20
```

Related Commands

clear vines neighbor
show vines neighbor
show vines route
vines route

vines output-network-filter

To filter the information contained in routing updates transmitted to other stations, use the **vines output-network-filter** interface configuration command. To disable this filtering, use the **no** form of this command.

vines output-network-filter *access-list-number*
no vines output-network-filter

Syntax Description

access-list-number	Number of the access list. It is a decimal number from 201 to 300.

Default

No filtering.

Command Mode

Interface configuration

Usage Guidelines

VINES routing messages contain topological entries that allow service and client nodes to select the best paths to destinations. This command provides filtering ability to an administrator so that they may selectively determine which routing entries should be passed on to other routers. This command may be useful in enforcing administrative policies of local server usage.

Example

The following example prevents all routes from being advertised to interface Ethernet 0 except the route to one single server:

```
vines routing
!
vines access-list 201 permit 27AF9A:1 0:0
vines access-list 201 deny 0:0 FFFFFFFF:FFFF
!
interface ethernet 0
vines metric 2
vines output-network-filter 201
```

vines propagate

To modify how routers forward a broadcast packet, use the **vines propagate** interface configuration command. To return to the default forwarding scheme, use the **dynamic** form of this command.

vines propagate [**dynamic**]
no vines propagate [**dynamic**]

Syntax Description

dynamic	(Optional) Propagate broadcasts on this interface only if there are no servers on any local network.

Default

Dynamic forwarding

Command Mode

Interface configuration

Usage Guidelines

If you specify the **vines propagate** command with no keywords, broadcast messages are always propagated on the interface.

The **vines propagate** command affects how the router decides whether to forward a broadcast packet out an interface. The normal decision is based on the settings of both the “hop count” and “class” fields of the VINES IP header, and also whether or not there are any servers present on any of the local network segments. In the default configuration, the router first looks to see if there are any local servers, and if so, follows the normal rules of VINES IP and forwards the broadcast out this interface based upon the “hop count” and the “class” field. If there are no local servers, then the router looks only at the “hop count” field before forwarding the broadcast out this interface. Enabling this command with no argument tells the router to always ignore the “class” field and make the forwarding decision based solely upon the “hop count” field. The **no** form of this command tells the router to always examine both the “hop count” and “class” fields.

Example

The following example always ignores the “class” field of the VINES IP header when deciding whether to forward a broadcast packet on interface Serial0:

```
interface serial 0
vines propagate
```

Related Commands

vines arp-enable
vines serverless

vines redirect

To determine how frequently a router sends an RTP redirect message on an interface, use the **vines redirect** interface configuration command. To restore the default, use the **no** form of this command.

vines redirect [*seconds*]
no vines redirect

Syntax Description

seconds	(Optional) Interval, in seconds, that the router waits after sending a redirect message on an interface before it sends another redirect message on that same interface. If you specify a value of 0, the router never sends redirect messages on that interface.

Default

1 second

Command Mode

Interface configuration

Usage Guidelines

VINES routing redirect packets contain topological entries that allow service and client nodes to select the best paths to destinations. When a service node determines that it should not be forwarding packets between two nodes, it sends a redirect packet to the sending node informing it of the better path.

Example

The following example prevents redirect messages from ever being sent on Ethernet interface 0:

```
vines routing
!
interface ethernet 0
vines metric 2
vines redirect 0
```

vines route

To specify a static route to a server, use the **vines route** global configuration command. To remove a static route from the routing table, use the **no** form of this command.

vines route *number address* [*whole* [*fractional*]]
no vines route *number address* [*whole* [*fractional*]]

Syntax Description

number	Number of the server to which to add or remove the static route.
address	VINES IP address of the neighbor station to use to reach the server.
whole	(Optional) Metric value assigned to this route.
fractional	(Optional) Fractional cost value associated with this route.

Default

No static routes are specified.

Command Mode

Global configuration

Usage Guidelines

The decision to use a static route or a dynamic route is always determined by the relative metric numbers.

Be careful when assigning static routes. If a static route is assigned with a better metric than the dynamic routes and the links associated with the static routes are lost, traffic may stop being forwarded, even though an alternative route might be available.

Example

The following example establishes a static route to the server at ABCD1234:

```
vines route ABCD1234 12345678:1 35
```

Related Commands

clear vines neighbor
clear vines route
show vines neighbor
show vines route
vines neighbor

vines route-cache

To enable fast switching, use the **vines route-cache** interface configuration command. To disable fast switching, use the **no** form of this command.

vines route-cache
no vines route-cache

Syntax Description

The command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

Usage Guidelines

The **vines route-cache** command enables the fast switching of VINES packets being transmitted out of the interface. However, forwarding of broadcast packets and responding to packets destined for the local router still occurs at the process level. When fast switching is disabled, all packets are forwarded at the process level.

Fast switching allows higher throughput by switching a packet using a cache created by previous packets. Fast switching provides load sharing on a per-packet basis just as slow switching does. Fast switching is enabled by default on all interfaces where it is supported. It is not supported on very old Ethernet, serial, and Token Ring interfaces, nor is it supported on serial interfaces using an encapsulation other than HDLC.

Packet transfer performance is generally better when fast switching is enabled. However, you may want to disable fast switching in order to save memory space on interface cards and help avoid congestion when high-bandwidth interfaces are writing large amounts of information to low-bandwidth interfaces.

When fast switching is enabled, the router maintains a fast-switching cache table. When transmitting a packet that is eligible to be fast switched, the router first checks the fast-switching cache table. If it finds an entry for the destination, the router uses that path. Otherwise, it searches the standard routing table and places the route it finds into the fast-switching cache table. The next time the router receives a packet for that destination, it uses the route in the fast-switching cache table.

Example

The following example disables fast switching on serial interface 0:

```
interface serial 0
bandwidth 19200
vines metric
no vines route-cache
```

Related Commands

clear vines cache
show vines cache
show vines route

vines routing

To enable VINES routing, use the **vines routing** global configuration command. To disable VINES routing, use the **no** form of this command.

vines routing [*address* | **recompute**]
no vines routing

Syntax Description

address	(Optional) Network address of the router. You should specify an address on a router that does not have any Ethernet or FDDI interfaces. You also can specify an address in the unlikely event that two routers map themselves to the same address.
recompute	(Optional) Dynamically redetermine the router's network address.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

Enabling VINES routing with the **vines routing** command starts both the VINES RTP and SRTP protocols. The router software dynamically determines which version of the VINES routing protocol stations on the network are using and then uses one or the other, or both protocols, as appropriate.

If a router contains Ethernet or FDDI interfaces, you do not need to specify an address because the router automatically maps itself into the VINES address space that is reserved for our routers. If you do specify an address, the router will use the specified address.

If a router contains only Token Ring interfaces (or Token Ring and serial interfaces), either the Token Ring interface must be fully initialized before you issue the **vines routing** command or you must specify an address in the **vines routing** command. This is because Token Ring interfaces have MAC addresses of 0000.0000.0000 until they are fully initialized.

Banyan has assigned us a portion of the overall VINES network number space. This portion is the set of all numbers that begin with the first 11 bits (of the 32) of 0011 0000 000. This number set appears in all our displays as a hexadecimal number beginning with 0x300 or 0x301. Routers attempt to automatically map themselves into our number space based upon the first nonzero Ethernet, Token Ring, or FDDI address found.

In theory, address conflicts are impossible, because VINES servers use their Banyan-assigned, unique key serial numbers as their network numbers and use a subnetwork number of one. Because the keys are unique, the server addresses are unique. VINES clients do not have addresses per se. The clients use a modified version of the address of the first file server found on the physical network: they assume the server's network number and are assigned a subnetwork number by that server. This address-assignment scheme means that it is likely that two clients on the same physical LAN will have different addresses. It requires that the router keep a cache of local neighbors as well as a cache of routing entries.

If you do not specify a network address and the router cannot compute one from a MAC address, the router selects a random address. There is no guarantee that this will be a unique address.

If you find that two routers have the same VINES network address, you should issue the **vines routing recompute** command on both routers. When recomputing its address, the router uses the same method used when originally determining its network address. If you issue this command on a router on which you have enabled the processing of ARP packets (with the **vines arp-enable** command) and if the router's address changes when it is recomputed, any clients that received their VINES network addresses from the router will lose all network connectivity, and you will have to reboot them.

Older implementations of out software mapped themselves to numbers beginning with 0xF80. This was done before Banyan made the address assignment.

Example

The following example enables VINES routing on interface Ethernet 0:

```
vines routing
!
interface ethernet 0
vines metric 2
```

Related Commands

vines arp-enable
vines metric

vines serverless

To configure a Banyan VINES network that does not have a server, use the **vines serverless** interface configuration command. To disable this feature, use the **no** form of this command.

vines serverless [**dynamic** | **broadcast**]
no vines serverless [**dynamic** | **broadcast**]

Syntax Description

dynamic	(Optional) Forward broadcasts toward one server only if there are no servers present on this interface.
broadcast	(Optional) Always flood broadcasts out all other router interfaces in order to reach all servers.

Default

Dynamic forwarding

Command Mode

Interface configuration

Usage Guidelines

If all keywords are omitted, broadcasts are alwasy forwarded toward one server.

The **vines serverless** command provides special processing for certain broadcast packets and certain packets directed at the router.

When you have a Banyan VINES network that has no server, by default the router will provide special processing for certain broadcast packets and certain packets directed at the router. This is necessary for proper functioning of the clients on a network without a server. This special processing allows a client to find the services that are provided by a server on another network. The dynamic nature of this processing allows the router to switch over from not providing serverless support to providing serverless support if the last server on a network fails. If you want the router to always provide serverless support, even when there are local servers present, you may override the default processing by issuing the **vines serverless** command with no argument. If you do not want the router to ever provide serverless support, you may also override the default in this way by issuing the **no vines serverless** command.

When the router receives a zero-hop broadcast on a serverless network, it does not follow the normal processing rules for VINES packets and discard the frames. Instead, it looks in its routing table for the nearest Banyan server. If this server is on a directly connected network, the router resends the broadcast message on that network as a MAC-level broadcast so that server and any others present can respond to it. If the nearest Banyan server is not on a directly connected network, the router resends the broadcast message on that network as a MAC-level unicast message directed at the first hop to that server. The next router will perform these same steps, assuming it is also configured for serverless support. The router can also be configured to always flood these broadcasts on all interfaces by using the command **vines serverless broadcast**. The decision on whether or not to flood is a trade-off between network bandwidth and finding more servers.

If you have configured this interface to forward towards a single destination, you may see which server has been selected as the forwarding target by looking at the output of the **show vines route** command. All servers on the same physical network as the target server will receive the broadcast.

Examples

The following example configures Ethernet interface 1, which is a network with no VINES servers:

```
interface ethernet 0
vines metric 2
!
interface ethernet 1
vines metric 2
```

Note that the **vines serverless** command is not necessary because the default setting is what is desired.

The following example configures Ethernet interface 1, which is a network with no VINES servers to always flood broadcasts to all other interfaces in the router:

```
interface ethernet 0
vines metric 2
!
interface ethernet 1
vines metric 2
vines serverless broadcast
```

The **vines serverless** command is necessary here because a nondefault setting is desired.

Related Command

show vines route
vines arp-enable
vines propagate

vines split-horizon

To use split horizon when sending routing updates, use the **vines split-horizon** interface configuration command. To disable split horizon, use the **no** form of this command.

vines split-horizon
no vines split-horizon

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

Usage Guidelines

The **vines split-horizon** command also affects whether broadcasts packets received on an interface are resent on the same interface.

The **vines split-horizon** command determines how much information is included in routing updates sent out an interface. It also determines whether received broadcasts will be retransmitted on the same interface. When you enable split horizon, routing updates sent out on a given interface will not include any information that was originally learned from that interface, and broadcasts will not be retransmitted on the receiving interface. This is because split horizon is designed for networks that are either broadcast networks, or are fully connected mesh networks. In these types of networks, resending this information is a waste of network bandwidth because all other stations on that network have already heard the information. Disabling split horizon will cause the router to include all information in routing updates, and to resend broadcast packets on the network from which they were received.

You can use this command on any interface, but generally it makes sense to use it only for X.25 and Frame Relay interfaces. You should disable split horizon on X.25 and Frame Relay networks that are not fully connected mesh topologies.

Example

The following example disables split horizon on an X.25 network:

```
interface serial 0
vines metric 2
no vines split-horizon
```

vines srtp-enabled

To enable Sequenced Routing Update Protocol (SRTP), use the **vines srtp-enabled** global configuration command. To disable SRTP, use the **no** form of this command.

vines srtp-enabled
no vines srtp-enabled

Syntax Description

This command has no arguments or keywords.

Default

The router runs Banyan's Routing Update Protocol (RTP) routing protocol only.

Command Mode

Global configuration

Usage Guidelines

When SRTP is enabled, the router dynamically determines whether it needs to send RTP messages, SRTP messages, or both.

Example

The following example enables SRTP on the router:

```
interface serial 0
vines routing
vines srtp-enabled
```

Related Command

vines routing

vines time access-group

To control the servers from which the router will accept VINES network time, use the **vines time access-group** global configuration command. To accept VINES network time messages from any server, use the **no** form of this command.

vines time access-group *access-list-number*
no vines time access-group

Syntax Description

access-list-number	Number of the access list. It is a decimal number from 201 to 300.

Default

Disabled

Command Mode

Global configuration

Example

The following example applies an access list to incoming time messages:

```
vines access-list 201 permit 27AF9A:1 0:0
vines access-list 201 deny 0:0 FFFFFFFF:FFFF
!
vines time participate
vines time access-group 201
```

Related Commands

show vines service
vines access-list (simple)
vines time destination
vines time participate
vines time set-system
vines time use-system

vines time destination

To control the servers to which the router sends VINES network time, use the **vines time destination** global configuration command. To send VINES network time messages to all servers, use the **no** form of this command.

vines time destination *address*
no vines time destination

Syntax Description

address	Destination VINES address for the network time messages.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

By default, the router sends VINES network time messages to the broadcast address.

You can enter the **vines time destination** command up to 20 times for 20 destination addresses.

Example

The following example specifies the servers to receive VINES time messages:

```
vines time participate
vines time destination 0027AF9F:0001
vines time destination 300001239:001
```

Related Commands

show vines service
vines time access-group
vines time participate
vines time set-system
vines time use-system

vines time participate

To enable the router's participation in the synchronization of time across a VINES network, use the **vines time participate** global configuration command. To disable the router's participation in time synchronization, use the **no** form of this command.

vines time participate
no vines time participate

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Global configuration

Usage Guidelines

The router always listens to the time synchronization messages on the network, and it tracks the network time. This command controls only the sending of time synchronization messages by the router. This arrangement means that you can use the **show vines services** EXEC command to see the network time even if the router is not actively participating in time synchronization.

Example

The following example disables the router's participation in the sending of VINES time messages:

```
no vines time participate
```

Related Commands

show vines service
vines access-list (simple)
vines time access-group
vines time destination
vines time set-system
vines time use-system

vines time set-system

To set the router's internal time based upon the received VINES network time, use the **vines time set-system** global configuration command. To uncouple the router's time from VINES network time, use the **no** form of this command.

vines time set-system
no vines time set-system

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

You should not use the **vines time set-system** command when running NTP on a router, because this command has no effect on these systems. NTP is considered to be a higher-priority clock than VINES, because it is a much more accurate timekeeping system.

Example

The following example sets the router's time from received VINES time messages:

```
vines time participate
vines time set-system
```

Related Commands

show vines service
vines access-list (simple)
vines time access-group
vines time destination
vines time participate
vines time use-system

vines time use-system

To set VINES network time based upon the router's internal time, use the **vines time use-system** global configuration command. To uncouple VINES network time from the router's time, use the **no** form of this command.

vines time use-system
no vines time use-system

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

The **vines time use-system** command causes the router to import the locally available time source (such as NTP, the Cisco 7000 clock, or DNSIX time) into the VINES time world as an authoritative clock. This is most useful when running NTP on the router. The router appears to the VINES network as a server dialing the NIST clock.

When you specify the **vines use-system** command, VINES will extract the system time and propagate it into the VINES world only if the system time is valid. If you are running NTP, the system time becomes valid when NTP synchronizes with a master. If you are not running NTP, but you do have an internal clock (such as exists on the Cisco 7000 router), you can force that time to be valid by specifying the **clock calendar-valid** command. This will allow VINES to propagate time based upon the Cisco 7000's clock chip.

Example

The following example sets VINES network time from the router's internal time:

```
ntp peer 131.108.13.111 version 2
!
vines time participate
vines time use-system
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

clock calendar-valid †
show vines service
vines access-list (simple)
vines time access-group
vines time destination
vines time participate
vines time set-system

vines update deltas

To modify the manner in which routing updates are sent, use the **vines update deltas** interface configuration command. To return to the default method, use the **no** form of this command.

vines update deltas
no vines update deltas

Syntax Description

This command has no arguments or keywords.

Default

No deltas

Command Mode

Interface configuration

Usage Guidelines

The **vines update deltas** command significantly modifies the way that routing information is propagated across the network.

On LAN media, using this command causes the router to stop transmitting and to stop expecting periodic routing updates. Instead, the router transmits and expects a periodic hello message. The difference between these two messages is whether routing information is included. The router will continue to send flash updates to inform its neighbors of any changes to current routing table information. This is the same frequency and type of routing updates used on LANs by VINES version 5.50, but our packet format differs from the VINES format.

On WAN media, using this command causes the router to transmit three normally spaced routing updates and then cease transmission. The router does *not* send periodic hello messages. The router will, however, continue to send flash updates to inform its neighbors of any changes to current routing table information. This is the same frequency and type of routing updates used on LANs by all versions of VINES, but our packet format differs from the VINES format.

Example

The following example modifies the propagation of routing update information on the WAN interface connected to serial interface 0:

```
interface serial 0
vines metric
vines update deltas
```

Related Commands

show vines interface
show vines neighbor
show vines route
vines metric

vines update interval

To modify the frequency at which routing updates are sent, use the **vines update interval** interface configuration command. To return to the default frequency, use the **no** form of this command.

vines update interval [*seconds*]
no vines update interval [*seconds*]

Syntax Description

seconds	Interval, in seconds, between the sending of periodic VINES routing updates. This can be a number in the range 0 to 2^{32} and will be rounded up to the nearest 5 seconds. The default value is 90 seconds. If you omit *seconds* or specify a value of 0, the default value of 90 seconds is used.

Default

90 seconds

Command Mode

Interface configuration

Usage Guidelines

The **vines update interval** command controls the interval at which the router sends routing updates. The routing update interval should be the same on all VINES-speaking entities on the same physical network.

For networks on which other vendors' entities are present, it is safe to use any setting in the range 30 to 100 seconds on networks. This is the range of update intervals supported by Banyan servers. You should use values outside of this range (with the exception of zero) only on networks that contain only our routers. You can use a value of zero on networks with only our routers or on WAN links connecting our routers and Banyan servers. In this configuration, you must also address application-level security requirements.

For Banyan VINES sites that support "change-only" updates on LAN networks, you can use the **vines update interval** command in LAN networks with both our routers and Banyan servers.

Example

The following example sets the update interval on serial interface 0 to a value of 270 seconds:

```
interface serial 0
vines metric
vines update interval 270
```

Related Commands

show vines interface
show vines neighbor
show vines route
vines metric

CHAPTER 16

DECnet Commands

Digital Equipment Corporation (Digital) developed the DECnet protocol to provide a way for its computers to communicate with one another. DECnet is currently in its fifth major product release called Phase V. DECnet Phase V is a superset of the OSI protocol suite, supports all OSI protocols, and is compatible with the previous release, Phase IV. DECnet Phase IV Prime supports inherent MAC addresses, which allow DECnet nodes to coexist with systems running other protocols that have MAC address restrictions. DECnet support on our routers includes local-area and wide-area DECnet Phase IV routing over Ethernet, Token Ring, FDDI, and serial lines (X.25, Frame Relay, SMDS).

Use the commands in this chapter to configure and monitor DECnet networks. For DECnet protocol configuration information and examples, refer to the "Configuring DECnet" chapter of the *Router Products Configuration Guide*.

access-list (standard)

To create a standard access list, use the **access-list** global configuration command. Use the **no** form of this command to delete the entire access list.

access-list *access-list-number* {**permit** | **deny**} *source source-mask*
no access-list

Syntax Description

access-list-number	Integer you choose between 300 and 399 that uniquely identifies the access list.
permit	Permits access when there is an address match.
deny	Denies access when there is an address match.
source	Source address. DECnet addresses are written in the form *area.node*. For example, 50.4 is node 4 in area 50. All addresses are in decimal.
source-mask	Mask to be applied to the address of the source node. Bits are set wherever the corresponding bits in the address should be ignored. All masks are in decimal.

Default

No access list is defined.

Command Mode

Global configuration

Usage Guidelines

In contrast with IP masks, a DECnet mask specification of “all ones” is entered as the decimal value 1023. In IP, the equivalent is 255.

Example

The following example sets up access list 300 to deny packets coming from node 4.51 and permit packets coming from 2.31:

```
access-list 300 deny 4.51 0.0
access-list 300 permit 2.31 0.0
```

Related Commands

access-list (extended)
access-list (filter connect initiate packets)
decnet access-group
decnet in-routing-filter
decnet out-routing-filter
show decnet interface

access-list (extended)

To create an extended access list, use the **access-list** global configuration command. Use the **no** form of this command to delete the entire access list.

access-list *access-list-number* {**permit** | **deny**} *source source-mask* [*destination destination-mask*]
no access-list

Syntax Description

access-list-number	Integer you choose between 300 and 399 that uniquely identifies the access list.
permit	Permits access when there is an address match.
deny	Denies access when there is an address match.
source	Source address. DECnet addresses are written in the form *area.node*. For example, 50.4 is node 4 in area 50. All addresses are in decimal.
source-mask	Mask to be applied to the address of the source node. All masks are in decimal.
destination	(Optional) Destination node's DECnet address in decimal format. DECnet addresses are written in the form *area.node*. For example, 50.4 is node 4 in area 50.
destination-mask	(Optional) Destination mask. DECnet addresses are written in the form *area.node*. For example, 50.4 is node 4 in area 50. All masks are in decimal.

Default

No access list is defined.

Command Mode

Global configuration

Example

In the following example, access list 301 is configured to allow traffic from any host in networks 1 and 3. It implies no other traffic will be permitted. (The end of a list contains an implicit "deny all else" statement.)

```
access-list 301 permit 1.0 0.1023 0.0 63.1023
access-list 301 permit 3.0 0.1023 0.0 63.1023
```

Related Commands

access-list (standard)
access-list (filter connect initiate packets)
decnet access-group
decnet in-routing-filter
decnet out-routing-filter
show decnet interface

access-list (filter connect initiate packets)

To create an access list that filters *connect initiate* packets, use the **access-list** global configuration command. Use the **no** form of this command to disable the access list.

access-list *access-list-number* {**permit** | **deny**} *source source-mask*
[*destination destination-mask* {**eq** | **neq**} [[*source-object*] [*destination-object*] [*identification*]] **any**]

no access-list

The optional argument *source-object* consists of the following string:

src [{**eq** | **neq** | **gt** | **lt**} *object-number*] [**exp** *regular-expression*] [**uic** [*group, user*]]

The optional argument *destination-object* consists of the following string:

dst [{**eq** | **neq** | **gt** | **lt**} *object-number*] [**exp** *regular-expression*] [**uic** [*group, user*]]

The optional argument *identification* consists of the following string:

[**id** *regular-expression*] [**password** *regular-expression*] [**account** *regular-expression*]

Syntax Description

access-list-number	Integer you choose between 300 and 399 that uniquely identifies the access list.
permit	Permits access when there is an address match.
deny	Denies access when there is an address match.
source	Source address. DECnet addresses are written in the form *area.node*. For example, 50.4 is node 4 in area 50. All addresses are in decimal.
source-mask	Mask to be applied to the address of the source node. All masks are in decimal.
destination	(Optional) Destination node's DECnet address in decimal format. DECnet addresses are written in the form *area.node*. For example, 50.4 is node 4 in area 50. All addresses are in decimal.
destination-mask	(Optional) Destination mask. DECnet addresses are written in the form *area.node*. For example, 50.4 is node 4 in area 50. All masks are in decimal.
eq | **neq**	Use either of these keywords: **eq**—item matches the packet if *all* the specified parts of *source-object*, *destination-object*, and *identification* match data in the packet. **neq**—item matches the packet if *any* of the specified parts do *not* match the corresponding entry in the packet.

source-object

(Optional) Contains the mandatory keyword **src** and one of the following optional keywords:

eq | **neq** | **lt** | **gt**—equal to, not equal to, less than, or greater than. These keywords must be followed by the argument *object-number*, a numeric DECnet object number.

exp—stands for expression; followed by a regular expression that matches a string.

uic—stands for user identification code; followed by a numeric user ID (UID*)* expression.The argument [*group, user*] is a numeric UID expression. In this case, the bracket symbols are literal; they must be entered. The group and user parts can either be specified in decimal, in octal by prefixing the number with a 0, or in hex by prefixing the number with 0x. The uic expression is displayed in show displays as an octal number.

destination-object

(Optional) Contains the mandatory keyword **dst** and one of the following optional keywords:

eq | **neq** | **lt** | **gt**—equal to, not equal to, less than, or greater than. These keywords must be followed by the argument *object-number*, a numeric DECnet object number.

exp—stands for expression; followed by a regular expression that matches a string.

uic—stands for user identification code; followed by a numeric user ID (UID*)* expression. In this case, the bracket symbols are literal; they must be entered. The group and user parts can either be specified in decimal, in octal by prefixing the number with a 0, or in hex by prefixing the number with 0x. The uic expression is displayed in show displays as an octal number.

identification

(Optional) Uses any of the following three keywords:

id—regular expression; refers to user ID.

password—regular expression; the password to the account.

account—regular expression; the account string.

any	Item matches if *any* of the specified parts *do* match the corresponding entries for *source-object*, *destination-object*, or *identification*.

See the "Regular Expressions" appendix for a description of regular expressions.

Default

No access list is defined.

Command Mode

Global configuration

Usage Guidelines

Depending upon the arguments you use, you can define access lists in three ways:

- Restrict access based on source addresses

  Use the *source* and *source-mask* arguments only.

- Restrict access based on destination addresses

  Use the *source*, *source-mask, destination*, and *destination-mask* arguments.

- Add filters to further narrow access

  Use the *source*, *source-mask, destination*, and *destination-mask* arguments, the **eq** | **neq** or **any** keywords and any or all of the following arguments: *source-object*, *destination-object*, and *identification*.

Table 16-1 lists the DECnet object numbers.

Table 16-1 Common DECnet Object Numbers

Name	Number	Description
FAL	17	File Access Listener
HLD	18	Host Loader
NML	19	Network Monitor Link/NICE
MIRROR	25	Loopback mirror
EVL	26	Event logger
MAIL	27	Mail
PHONE	29	Phone
NOTES	33	VAX Notes
CTERM	42	Terminal sessions
DTR	63	DECnet Test Sender/Receiver

Examples

The following example illustrates an access list for matching all connect packets for object number 27:

```
access-list 300 permit 0.0 63.1023 eq dst eq 27
```

The following example illustrates an access list for matching all connect packets *except* for the object number 17:

```
access-list 300 permit 0.0 63.1023 neq dst eq 17
```

The following example illustrates an access list for matching all connect packets where the access identification was *SYSTEM*:

```
access-list 300 permit 0.0 63.1023 eq id ^SYSTEM$
```

The following example illustrates an access list for matching all connect packets from area 1 to object number 27 (27 = VAX/VMS Personal Utility or MAIL) where *SYSTEM* is the originating user:

```
access-list 300 permit 1.0 0.1023 eq src exp ^SYSTEM$ dst eq 27
```

The following example illustrates an access list for matching any connect packet and can be used at the end of a list to permit any packets not already matched:

```
access-list 300 permit 0.0 63.1023 eq any
```

Related Commands

access-list (standard)
access-list (extended)
decnet access-group
decnet in-routing-filter
decnet out-routing-filter
show decnet interface

clear decnet counters

To clear DECnet counters that are shown in the output of the **show decnet traffic** EXEC command, use the **clear decnet counters** EXEC command.

clear decnet counters

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Example

The following example provides sample output of the **clear decnet counters** EXEC command:

```
Router# clear decnet counters
Router# show decnet traffic
Total: 3 received, 0 format errors, 0 unimplemented
       0 not a gateway, 0 no memory, 0 no routing vector
       0 congestion encountered
Hellos: 3 received, 0 bad, 0 other area, 4 sent
Level 1 routing: 0 received, 0 bad, 0 other area, 4 sent
Level 2 routing: 0 received, 0 not primary router, 2 sent
Data: 0 received, 0 not long format, 0 too many visits
      0 forwarded, 0 returned, 0 converted, 0 local destination
      0 access control failed, 0 no route, 0 encapsulation failed
      0 inactive network, 0 incomplete map
Router#
```

Related Command

show decnet traffic

decnet access-group

To create a DECnet access group, use the **decnet access-group** interface configuration command.

decnet access-group *access-list-number*

Syntax Description

access-list-number	Either a standard or extended DECnet access list. A standard DECnet access list applies to source addresses. The value (or values in the case of extended lists) can be in the range 300 through 399.

Default

No access group is defined.

Command Mode

Interface configuration

Example

The following example applies access list 389 to interface Ethernet 1:

```
interface ethernet 1
decnet access-group 389
```

Related Commands

access-list (standard)
show decnet interface

decnet advertise

To configure border routers to propagate Phase IV areas through an OSI backbone, use the **decnet advertise** global configuration command. To disable this feature, use the **no** form of this command.

decnet advertise *decnet-area hops cost*
no decnet advertise [*decnet-area*]

Syntax Description

decnet-area	Phase IV area that you want propagated.
hops	Hop count to be associated with the route being advertised. Default is 0.
cost	Cost to be associated with the route being advertised. Default is 0.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

The output from the **show decnet route** EXEC command shows the cost and hop count for routes.

The **decnet advertise** command is used by border routers for propagating Phase IV areas through an OSI backbone.

The **decnet advertise** command and the **clns route** *nsap-prefix* **discard** command work together. When a router has DECnet Phase IV/V conversion enabled, any packet with the specified CLNS NSAP prefix causes CLNS to behave as if no route was found. That router then looks up the route to the border router that is advertising the Phase IV route. In turn, the router that is advertising the DECnet Phase IV route converts the packet to Phase V and sends it through the OSI cloud to the border router that is advertising the CLNS discard static route. Once it gets there, the packet is converted back to Phase IV.

The CLNS discard routes are created dynamically when the advertised adjacencies are propagated through the CLNS cloud. When a DECnet interface is disabled, the adjacencies are lost and the CLNS discard route is deleted. The DECnet area routing states are displayed in the output from the **show decnet route** EXEC command.

Example

The following example shows a partial use of the **decnet advertise** command:

```
decnet conversion 49
decnet advertise 4
clns route 49.0001 discard
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

clns route discard †
show decnet route

decnet area-max-cost

To set the maximum cost specification value for *interarea* routing, use the **decnet area-max-cost** global configuration command.

decnet [*network-number*] **area-max-cost** *value*

Syntax Description

network-number	(Optional) Network number in the range 0 through 3. Specified when using Address Translation Gateway (ATG). If not specified, the default is network 0.
value	Maximum cost for a route to a distant area that the router may consider usable; the router treats as unreachable any route with a cost greater than the value you specify. A valid range for cost is from 1 through 1022. This parameter is only valid for area routers. The default is 1022.

Defaults

*network-numbe*r: 0
value: 1022

Command Mode

Global configuration

Usage Guidelines

Make sure you have used the **decnet node-type area** global configuration command before using this command.

Example

In the following example, the node type is specified as area and the maximum cost is set to 500. Any route with a cost exceeding 500 will be considered unreachable by this router.

```
decnet node-type area
decnet area-max-cost 500
```

Related Commands

decnet area-max-hops
decnet node-type
show decnet interface

decnet area-max-hops

To set the maximum hop count value for *interarea* routing, use the **decnet area-max-hops** global configuration command.

decnet [*network-number*] **area-max-hops** *value*

Syntax Description

network-number	(Optional) Network number in the range 0 through 3. Specified when using Address Translation Gateway (ATG). If not specified, the default is network 0.
value	Maximum number of hops for a usable route to a distant area. The router treats as unreachable any route with a count greater than the value you specify. A valid range for the hop count is from 1 through 30. The default is 30 hops.

Default

30 hops

Command Mode

Global configuration

Usage Guidelines

This command is only valid for area routers. Make sure you have issued the **decnet node-type area** global configuration command before using this command.

Example

The following example sets the router to be a Level 2 router, then sets a maximum hop count of 21:

```
decnet node-type area
decnet area-max-hops 21
```

Related Commands

decnet area-max-cost
decnet node-type
show decnet interface

decnet congestion-threshold

Use the **decnet congestion-threshold** interface configuration command to set the congestion-experienced bit if the output queue has more than the specified number of packets in it. A *number* value of zero or the **no** form of the command prevents this bit from being set. Use the **no decnet congestion-threshold** command to remove the parameter setting and set it to 0.

decnet congestion-threshold *number*
no decnet congestion-threshold

Syntax Description

number	Number of packets that are allowed in the output queue before the system will set the congestion experience bit. This value is an integer between 0 and 0x7fff. The value zero prevents this bit from being set. Only relatively small integers are reasonable. The default is 1 packet.

Default

1 packet

Command Mode

Interface configuration

Usage Guidelines

If a router configured for DECnet experiences congestion, it sets the congestion-experienced bit.

Example

The following example sets the congestion threshold to 10:

```
interface Ethernet 0
decnet congestion-threshold 10
```

decnet conversion

To allow Phase IV routers (running Software Release 9.1 or higher) to run in a Phase V network and vice versa, enable conversion with the **decnet conversion** global configuration command. To disable conversion, use the **no** form of this command.

decnet conversion *nsap-prefix*
no decnet conversion *nsap-prefix*

Syntax Description

nsap-prefix	Value used for the IDP field when constructing NSAPs from a Phase IV address.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

To enable DECnet conversion, you must configure both DECnet and ISO CLNS on your router.

DECnet Phase V is OSI-compatible and conforms to the ISO 8473 (CLNP/CLNS) and ISO 9542 (ES-IS) standards. Digital has defined algorithms for mapping a subset of the Phase V address space onto the Phase IV address space and for converting Phase IV and Phase V packets back and forth. This allows a network administrator to support both Phase IV hosts in Phase V networks and Phase V hosts in Phase IV networks.

Our implementation differs from Digital's in how reachability information is advertised. Our implementation allows you to add Phase V support without modifying your existing Phase IV support. It also delays converting packets from Phase IV to Phase V, while Digital's implementation converts as soon as possible.

It is essential that the area you specify in the **decnet routing** global configuration command is the same as the local area you specified with the **net** router configuration command for the CLNS network.

Make sure that the area you specify in the **decnet conversion** command is the same as the area you specified for the CLNS network. Also note that the DECnet area is specified in decimal, and the CLNS area is specified in hexadecimal.

The **decnet routing** command is specified with a decimal address, while the **net** command address is specified in hexadecimal. In addition, the *nsap-prefix* specified on the **decnet conversion** command must match one of the NETs for this router.

The following guidelines apply:

- Host connectivity across multiple areas is only possible if a Level 2 path exists for which every Level 2 router in the path supports a common protocol: Phase IV or Phase V. If not all routers support both protocols, those routers that do *must* have conversion enabled.

- Host connectivity across a single area is only possible if a Level 1 path exists for which every Level 1 router in the path supports a common protocol: Phase IV or Phase V. If not all routers support both protocols, those routers that do *must* have conversion enabled.
- The Level 2 backbone *must* have conversion enabled in all Level 2 routers that support an area that needs conversion.

Example

The following example enables DECnet conversion on a router with the area tag xy and Phase IV address 20.401 using an ISO IGRP router:

```
clns routing
decnet routing 20.401
decnet max-address 600
!
router iso-igrp xy
net 47.0004.004d.0014.aa00.0400.9151.00
!
decnet conversion 47.0004.004d
!
interface ethernet 0
decnet cost 4
clns router iso-igrp xy
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

net †
show decnet interface
show decnet route

decnet cost

To set a cost value for an interface, use the **decnet cost** interface configuration command. Use the **no** form of this command to disable DECnet routing for an interface.

decnet cost *cost-value*
no decnet cost

Syntax Description

cost-value	Integer from 1 through 63. There is no default cost for an interface, although a suggested cost for FDDI is 1, for Ethernet is 4, and for serial links is greater than 10.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

After DECnet routing has been enabled, you must assign a cost to each interface over which you want DECnet to run. Assigning a cost in effect enables DECnet routing for an interface. Most DECnet installations have an individualized routing strategy for using costs. Therefore, check the routing strategy used at your installation to ensure that costs you specify are consistent with those set for other hosts on the network.

Example

The following example establishes a DECnet routing process for the router and sets the router's DECnet address to 21.456, then sets a cost of 4 for the Ethernet 0 interface:

```
decnet routing 21.456
interface ethernet 0
decnet cost 4
```

Related Commands

decnet encapsulation
decnet node-type
decnet routing
show decnet interface
show decnet route

decnet encapsulation

To provide DECnet encapsulation over Token Ring, use the **decnet encapsulation** interface configuration command.

decnet encapsulation {**pre-dec** | **dec**}

Syntax Description

pre-dec	Configures routers for operation on the same Token Ring with routers running software versions prior to 9.1. In this mode, Cisco routers cannot communicate with non-Cisco equipment. Referred to as Cisco-style encapsulation.
dec	Provides encapsulation that is compatible with other Digital equipment. All Cisco routers must be running Software Release 9.1 or later.

Default

Encapsulation is compatible with other Digital equipment.

Command Mode

Interface configuration

Usage Guidelines

If you have both Software Release 9.0 and 9.1 routers in the same network, you must use the **pre-dec** encapsulation type on the 9.1 routers.

Note You must first enable DECnet routing on the selected Token Ring interface before you can configure the DECnet encapsulation mode.

Example

The following example sets Cisco-style encapsulation for DECnet routing, which means that Cisco and Digital equipment will not interoperate over Token Ring:

```
interface tokenring 0
decnet encapsulation pre-dec
decnet cost 4
```

Related Commands

decnet cost
show decnet interface

decnet hello-timer

To change the interval for sending broadcast hello messages, use the **decnet hello-timer** interface configuration command. To restore the default value, use the **no** form of this command.

decnet hello-timer *seconds*
no decnet hello-timer

Syntax Description

seconds	Interval at which the router sends hello messages. It can be a decimal number in the range 1 through 8191 seconds; the default is 15 seconds.

Default

15 seconds

Command Mode

Interface configuration

Usage Guidelines

The router broadcasts hello messages on all interfaces with DECnet enabled. Other hosts on the network use the hello messages to identify the hosts with which they can communicate directly. On extremely slow serial lines, you may want to increase the default value to reduce overhead on the line.

Example

The following example increases the hello interval to 2 minutes (120 seconds) on interface serial 1:

```
interface serial 1
decnet hello-timer 120
```

Related Command

show decnet interface

decnet host

Use the **decnet host** global configuration command to associate a name-to-DECnet address mapping, which will show up in the output of various commands. To disable name mapping, use the **no** form of this command.

decnet host *name decnet-address*
no decnet host *name*

Syntax Description

name	A name you choose that uniquely identifies this DECnet address.
decnet-address	Source address. DECnet addresses are written in the form *area.node*. For example, 50.4 is node 4 in area 50. All addresses are in decimal.

Default

No name is defined.

Command Mode

Global configuration

Usage Guidelines

The assigned name is displayed, where applicable, in **show decnet route** and **show hosts** EXEC command output.

The name can also be used with the **ping decnet** command.

Example

The following example defines name-to-DECnet address mapping:

```
decnet host cisco1 3.33
router# show decnet route
  Area       Cost  Hops   Next Hop to Node       Expires  Prio
*2             0     0     (Local) -> 2.33
*3             4     1   Ethernet1 -> CISCO1        33     64    A+
router# show hosts
Default domain is cisco.com
Name/address lookup uses domain service
Name servers are 255.255.255.255

Host                    Flags      Age Type   Address(es)
CISCO1                  (perm, OK)  0 DECnet   3.33
```

decnet in-routing-filter

To provide access control to hello messages or routing information received on an interface, use the **decnet in-routing-filter** interface configuration command. Use the **no** form of this command to remove access control.

decnet in-routing-filter *access-list-number*
no decnet in-routing-filter

Syntax Description

access-list-number	Standard DECnet access list. This list applies to source addresses. The value can be in the range 300 through 399.

Default

No access control is defined.

Command Mode

Interface configuration

Example

In the following example, interface Ethernet 0 is set up with a DECnet in-routing filter of 321, which means that any hello messages sent from addresses that are denied in list 321 will be ignored. Additionally, all node addresses listed in received routing messages on this interface will be checked against the access list, and only routes passing the filter will be considered usable.

```
interface ethernet 0
decnet in-routing-filter 321
```

Related Commands

access-list (standard)
decnet out-routing-filter
show decnet interface

decnet map

To establish an address translation for selected nodes, use the **decnet map** global configuration command.

decnet *first-network* **map** *virtual-address second-network real-address*

Syntax Description

first-network	DECnet network numbers in the range 0 through 3.
virtual-address	Numeric DECnet address (10.5, for example).
second-network	DECnet network number you map to; DECnet numbers range from 0 through 3.
real-address	Numeric DECnet address (10.5, for example).

Default

No address translation is defined.

Command Mode

Global configuration

Usage Guidelines

Keep the following limitations in mind when configuring the Address Translation Gateway (ATG):

- Both nodes that want to communicate across the ATG must exist in the translation map. Other nodes outside of the map will see route advertisements for the mapped address but will be unable to communicate with them. An unmapped node trying to communicate with a mapped node will always get the message "Node unreachable." This can be confusing if another nearby node can communicate with mapped nodes because it is also a mapped node.
- Third-party DECnet applications could fail if they pass node number information in a data stream (most likely a sign of a poorly designed application).
- Routing information for mapped addresses is static and does not reflect the reachability of the actual node in the destination network.

As an additional feature and security caution, DECnet "Poor Man's Routing" can be used between nodes outside of the translation map as long as those nodes have access to nodes that are in the map, so that a user on node B could issue the following VMS command:

```
$ dir A::D::E::
```

When a Poor Man's Routing connection is made between two networks, only the two adjacent nodes between the networks will have any direct knowledge about the other network. Application-level network access may then be specified to route through the connection.

Note We do not support Poor Man's Routing directly; the intermediate nodes must be VMS systems with Poor Man's Routing enabled in file access language (FAL).

Example

In the following example, packets in Network 0 sent to address 19.5 will be routed to Network 1, and the destination address will be translated to 50.1. Packets sent to address 47.1 in Network 1 will be routed to Network 0 as 19.1.

```
decnet 0 map 19.5 1 50.1
decnet 1 map 47.1 0 19.1
```

Related Command

show decnet map

decnet max-address

To configure the router with a maximum number of node addresses, use the **decnet max-address** global configuration command.

decnet [*network-number*] **max-address** *value*

Syntax Description

network-number	(Optional) Network number in the range 0 through 3. Specified when using Address Translation Gateway (ATG). If not specified, the default is network 0.
value	A number less than or equal to 1023 that represents the maximum address possible on the network. In general, all routers on the network should use the same value for this argument. The default is 1023.

Default

1023 node addresses

Command Mode

Global configuration

Usage Guidelines

DECnet routers do not have the concept of aging out a route. Therefore, all possible areas or nodes must be advertised as unreachable if they cannot be reached. Since it is best to keep routing updates small, you need to indicate the default maximum possible node and area numbers that can exist in the network.

Example

The following example configures a small network to a maximum address value of 300:

```
decnet max-address 300
```

Related Command

decnet max-area

decnet max-area

To set the largest number of areas that the router can handle in its routing table, use the **decnet max-area** global configuration command.

decnet [*network-number*] **max-area** *area-number*

Syntax Description

network-number	(Optional) Network number in the range 0 through 3. Specified when using Address Translation Gateway (ATG). If not specified, the default is network 0.
area-number	Area number from 1 through 63. Like the **decnet max-address** global configuration command value, this argument controls the sizes of internal routing tables and of messages sent to other nodes. All routers on the network should use the same maximum address value. The default is 63.

Default

63 areas

Command Mode

Global configuration

Example

In the following example, the largest area to be stored in the routing table is 45:

```
decnet max-area 45
```

Related Commands

decnet max-address
show decnet interface

decnet max-cost

To set the maximum cost specification for *intra-area* routing, use the **decnet max-cost** global configuration command.

decnet [*network-number*] **max-cost** *cost*

Syntax Description

network-number	(Optional) Network number in the range 0 through 3. Specified when using Address Translation Gateway (ATG). If not specified, the default is network 0.
cost	Cost from 1 through 1022. The default is 1022.

Default

1022

Command Mode

Global configuration

Usage Guidelines

The router ignores routes within its local area that have a cost greater than the value you specify.

Example

In the following example, the node type is specified as a Level 1 router and the maximum cost is set to 335. Any route whose cost exceeds 335 will be considered unreachable by this router.

```
decnet node-type routing-iv
decnet max-cost 335
```

Related Commands

decnet max-hops
decnet max-paths
decnet node-type routing-iv
decnet path-split-mode
show decnet interface

decnet max-hops

To set the maximum hop count specification value for *intra-area* routing, use the **decnet max-hops** global configuration command.

decnet [*network-number*] **max-hops** *hop-count*

Syntax Description

network-number	(Optional) Network number in the range 0 through 3. Specified when using Address Translation Gateway (ATG). If not specified, the default is network 0.
hop-count	Hop count from 1 through 30. The router ignores routes that have a hop count greater than the corresponding value of this parameter. The default is 30 hops.

Default

30 hops

Command Mode

Global configuration

Example

The following example sets the router to be a Level 1 router, then sets a maximum hop count of 2:

```
decnet node-type routing-iv
decnet max-hops 2
```

Related Commands

decnet max-cost
decnet max-paths
decnet multicast-map
decnet node-type routing-iv

decnet max-paths

To define the maximum number of equal-cost paths to a destination that the router will keep in its routing table, use the **decnet max-paths** global configuration command.

decnet [*network-number*] **max-paths** *value*

Syntax Description

network-number	(Optional) Network number in the range 0 through 3. Specified when using Address Translation Gateway (ATG). If not specified, the default is network 0.
value	Decimal number equal to the maximum number of equal-cost paths the router will save. The valid range is from 1 through 31. The default is 1.

Default

1 equal-cost path

Command Mode

Global configuration

Usage Guidelines

Limiting the number of equal-cost paths can save memory on routers with limited memory or very large configurations. Additionally, in networks with a large number of multiple paths and end-systems with limited ability to cache out-of-sequence packets, performance may suffer when traffic is split between many paths.

Limiting the size of the routing table will not affect your router's ability to recover from network failures transparently, provided that you do not make the maximum number of paths too small. If more than the specified number of equal-cost paths exist, and one of those paths suddenly becomes unusable, the router will discover an additional path from the paths it has been ignoring.

Example

In the following example, the router will save no more than three equal-cost paths:

```
decnet max-paths 3
```

Related Commands

decnet max-cost
decnet max-hops
decnet path-split-mode
show decnet interface
show decnet route

decnet max-visits

To set the limit on the number of times a packet can pass through a router, use the **decnet max-visits** global configuration command.

decnet [*network-number*] **max-visits** *value*

Syntax Description

network-number	(Optional) Network number in the range 0 through 3. Specified when using Address Translation Gateway (ATG). If not specified, the default is network 0.
value	Number of times a packet can pass through a router. It can be a decimal number in the range 1 through 63. If a packet exceeds *value*, the router discards the packet. Digital recommends that the value of the **max-visits** parameter be at least twice that of the **max-hops** parameter, to allow packets to still reach their destinations when routes are changing. The default is 63 times.

Default

63 times

Command Mode

Global configuration

Example

The following example of intra-area routing configuration specifies Level 1 routing, a maximum hop count of 28, and maximum number of visits of 62 (which is more than twice 28).

```
decnet node-type routing-iv
decnet max-hops 28
decnet max-visits 62
```

Related Commands

decnet max-hops
show decnet interface
show decnet traffic

decnet multicast-map

Use the **decnet multicast-map** interface configuration command to specify a mapping between DECnet multicast addresses and Token Ring functional addresses, other than the default mapping. The **no** form of this command deletes the specified information.

decnet multicast-map *multicast-address-type functional-address*
no decnet multicast-map *multicast-address-type functional-address*

Syntax Description

multicast-address-type	Type of multicast address that is used. The following are valid values for the argument: **iv-all-routers** (All Phase-IV routers) **iv-all-endnodes** (All Phase-IV endnodes) **iv-prime-all-routers** (All Phase IV Prime routers)
functional-address	Functional MAC address that this multicast ID will map to. In the form of "c000.xxxx.yyyy."

Default

Enabled, with the default mapping listed in Table 16-2.

Command Mode

Interface configuration

Usage Guidelines

This command is valid for Token Ring interfaces only. The command will reject a functional address that does not start with "C000" or "c000."

Routing multicasts and end node multicasts must be on different functional addresses.

Table 16-2 Default Mapping of DECnet Multicast Address Types and Token Ring Functional Addresses

DECnet Multicast Address Type	Token Ring Functional Address
L1 router L2 router	C000.1000.0000
End node	C000.0800.0000
DECnet Phase IV-Prime router	C000.1000.0000

Example

In the following example, interface Token Ring 1 is configured for multicasts of all Phase IV end nodes and the multicast ID is configured to map to MAC address c000.2222.3333.

```
interface tokenring 1
```

```
decnet multicast-map iv-all-endnodes c000.2222.3333
```

decnet node-type

To specify the node type, use the **decnet node-type** global configuration command.

decnet [*network-number*] **node-type** {**area** | **routing-iv**}

Syntax Description

network-number	(Optional) Network number in the range 0 through 3. Specified when using Address Translation Gateway (ATG). If not specified, the default is network 0.
area	Router participates in the DECnet routing protocol with other area routers, as described in the Digital documentation, and routes packets from and to routers in other areas. This is sometimes referred to as Level 2, or interarea, routing. An area router does not just handle interarea routing; it also acts as an intra-area or Level 1 router in its own area.
routing-iv	Router acts as an intra-area (standard DECnet Phase IV, Level 1 router) and ignores Level 2 routing packets. In this mode, it routes packets destined for other areas to a designated interarea router, exchanging packets with other end-nodes and routers in the same area.

Default

No node type is specified.

Command Mode

Global configuration

Example

In the following example, the router node type is specified as *area*, or Level 2:

```
decnet node-type area
```

Related Commands

decnet cost
decnet routing
show decnet interface

decnet out-routing-filter

To provide access control to routing information being sent out on an interface, use the **decnet out-routing-filter** interface configuration command. Use the **no** form of this command to remove access control.

decnet out-routing-filter *access-list-number*
no decnet out-routing-filter

Syntax Description

access-list-number	Standard DECnet access list applying to source addresses. The value can be in the range 300 through 399**.**

Default

No access control to routing information is defined.

Command Mode

Interface configuration

Usage Guidelines

Addresses that fail this test are shown in the update message as unreachable.

Example

In the following example, interface Ethernet 1 is set up with a DECnet out-routing filter of 351. This filter is applied to addresses in the transmitted routing updates. Transmitted hello messages are not filtered.

```
interface ethernet 1
decnet out-routing-filter 351
```

Related Commands

access-list (standard)
decnet in-routing-filter
show decnet interface

decnet path-split-mode

To specify how the router will split the routable packets between equal-cost paths, use the **decnet path-split-mode** global configuration command with the appropriate keyword.

decnet path-split-mode {**normal** | **interim**}

Syntax Description

normal	Normal mode, where equal-cost paths are selected on a round-robin basis. This is the default.
interim	Traffic for any particular (higher-layer) session is always routed over the same path. This mode supports older implementations of DECnet (VMS Versions 4.5 and earlier) that do not support out-of-order packet caching. Other sessions may take another path, thus using equal-cost paths that a router may have for a particular destination.

Default

Normal mode

Command Mode

Global configuration

Example

In the following example, the router will split routable packets between equal-cost paths using the round-robin (or first-come, first-served) basis:

```
decnet path-split-mode normal
```

Related Commands

decnet max-cost
decnet max-paths

decnet propagate static

Use this form of the **decnet propagate static** global configuration command to enable static route propagation. The **no** form of this command disables propagation.

decnet route propagate static
no decnet route propagate static

Syntax Description

This command has no arguments or keywords.

Default

No default routes are propagated.

Command Mode

Global configuration

Usage Guidelines

By default, DECnet static routes will not be propagated to other routers. Use the **decnet propagate static** command to enable static route propagation. A default route will be used only after DECnet conversion is checked.

Examples

The following example shows how to enable static route propagation for the specified static and default routes:

```
decnet propagate static
!
decnet route 3.0 ethernet 0 aa00.0400.0404
decnet route 5.0 serial 0
decnet route 5.100 serial 2
decnet route default 2.100
decnet route 6.0 2.3 4 5
```

Related Commands

decnet route (interface static route)
decnet route (to enter a static route)
show decnet
show decnet static

decnet route-cache

To enable fast-switching, use the **decnet route-cache** interface configuration command. To disable fast switching, use the **no** form of this command.

decnet route-cache
no decnet route-cache

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

Usage Guidelines

By default, our DECnet routing software implements fast switching of DECnet datagrams. There are times when it makes sense to disable fast switching. This is especially important when using rates slower than T1.

Fast switching uses memory space on interface cards. In situations where a high-bandwidth interface is writing large amounts of information to a low-bandwidth interface, additional memory could help avoid congestion on the slow interface.

Example

In the following example, fast switching is disabled on interface ethernet 0:

```
interface ethernet 0
no decnet route-cache
```

decnet router-priority

To elect a designated router to which packets will be sent when no destination is specified, use the **decnet router-priority** interface configuration command.

decnet router-priority *value*

Syntax Description

value	Priority of the router. This can be a number in the range 0 through 127. The larger the number the higher the priority. The default priority is 64.

Default

64

Command Mode

Interface configuration

Usage Guidelines

The *designated* router is the router to which all end nodes on an Ethernet communicate if they do not know where else to send a packet. The designated router is chosen through an election process in which the router with the highest priority gets the job. When two or more routers on a single Ethernet in a single area share the same highest priority, the unit with the highest node number is elected. You can reset a router's priority to help ensure that it is elected designated router in its area.

On a LAN with both DECnet IV and DECnet IV Prime hosts, make sure that a bilingual router always becomes the designated router.

DECnet end systems use the designated router only when they have no other information about how to reach a particular system. The end systems maintain a cache of how to reach other systems on the network. The cache contains the following information:

```
<remote system DECnet address>  <next hop DECnet address>
```

When an end system receives a packet, it examines three pieces of information: the intra-LAN bit, the source address, and the previous hop. If the intra-LAN bit is set, indicating that the packet has never left this wire (and thus the remote system is reachable without a router), a cache entry is created as follows:

```
<remote system DECnet address> = <source address>
<next hop DECnet address> = <source address>
```

If the intra-LAN bit is not set, indicating that the packet has come from another network, the cache entry is created as follows:

```
<remote system DECnet address> = <source address>
<next hop DECnet address> = <previous hop>
```

If there is no cache entry, then the designated router is used. This means that when starting a session, the designated router is used, but the reverse traffic will populate a cache entry so that the router can later communicate directly.

A DECnet IV Prime end node sends a packet to the Unknown Destination multicast if it has no cache entry for the destination and has no designated router.

Example

In the following example, DECnet priority for this router is set to 110 on Ethernet 1:

```
interface ethernet 1
decnet router-priority 110
```

decnet route (interface static route)

Use this form of the **decnet route** global configuration command to create an interface static route. The **no** form of this command removes this route.

decnet route *decnet-address next-hop-type number* [*snpa-address*] [*hops* [*cost*]]
no decnet route *decnet-address next-hop-type number*

Syntax Description

decnet-address	DECnet address. This value is entered into a static routing table and used to match a destination DECnet address. Use a node address value of 0 to specify an area static route.
next-hop-type	Interface type.
number	Interface unit number.
snpa-address	(Optional) Optional for serial links; required for multiaccess networks.
hops	(Optional) Hop count to be associated with the route being advertised. Default is 0.
cost	(Optional) Cost to be associated with the route being advertised. Default is 0.

Default

No interface static routes are created.

Command Mode

Global configuration

Usage Guidelines

If you do not specify an SNPA address when you have a multiaccess network, you will receive an error message indicating a bad SNPA. By default, DECnet static routes will not be propagated to other routers. Use the **decnet propagate static** command to enable propagation.

Examples

The following example shows how to create a static route for a serial interface. No SNPA need be specified for point-to-point interfaces.

```
decnet route 3.1 serial 1
```

The following example shows how to create a static route for an Ethernet interface. The SNPA must be specified for an interface that is not point-to-point.

```
decnet route 3.2 ethernet 1 aa00.0400.0104
```

Related Commands

decnet propagate static
decnet route (to enter a static route)
decnet route default (interface default route)
decnet route default (to enter a default route)
show decnet static

decnet route (to enter a static route)

Use this form of the **decnet route** global configuration command to enter a specific static route. DECnet addresses that match are forwarded to the *next-hop-address*. The **no** form of this command removes this route.

decnet route *decnet-address next-hop-address* [*hops* [*cost*]]
no decnet route *decnet-address next-hop-address*

Syntax Description

decnet-address	DECnet address. This value is entered into a static routing table and used to match a destination DECnet address. Use a node address value of 0 to specify an area static route.
next-hop-address	This value is used to establish the next hop of the route for forwarding packets.
hops	(Optional) Hop count to be associated with the route being advertised. Default is 0.
cost	(Optional) Cost to be associated with the route being advertised. Default is 0.

Default

No interface static routes are created.

Command Mode

Global configuration

Usage Guidelines

Area static routes can be configured by specifying a DECnet node address of 0. By default, DECnet static routes will not be propagated to other routers. Use the **decnet propagate static** command to enable propagation.

Examples

The following example shows how to create a static route for 1.1 that points to 1.9 and uses default values of 0 for the *hops* and *cost*:

```
decnet route 1.1 1.9
```

The following example shows how to create a static route for 3.100 that points to 3.4 and specifies values for the *hops* and *cost*:

```
decnet route 3.100 3.4 9 8
```

The following example shows how to create a static route for area 1 that points to 2.999:

```
decnet route 1.0 2.999
```

Related Commands

decnet propagate static
decnet route (interface static route)
decnet route default (interface default route)
decnet route default (to enter a default route)
show decnet static

decnet route default (interface default route)

Use this form of the **decnet route default** global configuration command to create an interface default route. The **no** form of this command removes this route.

decnet route default *next-hop-type number* [*snpa-address*] [[*hops* [*cost*]]
no decnet route default *next-hop-type number*

Syntax Description

next-hop-type	Interface type.
number	Interface unit number.
snpa-address	(Optional) Optional for serial links; required for multiaccess networks.
hops	(Optional) Hop count to be associated with the route being advertised. Default is 0.
cost	(Optional) Cost to be associated with the route being advertised. Default is 0.

Default

No interface default routes are created.

Command Mode

Global configuration

Usage Guidelines

If you do not specify an SNPA address when you have a multiaccess network, you will receive an error message indicating a bad SNPA.

A default route will be used only after DECnet conversion is checked. DECnet default routes will not be propagated to other routers.

Examples

The following example shows how to create a default route for a serial interface. No SNPA need be specified for point-to-point interfaces.

```
decnet route default serial 1
```

The following example shows how to create a default route for an Ethernet interface. The SNPA must be specified for an interface that is not point-to-point.

```
decnet route default ethernet 1 aa00.0400.0104
```

Related Commands

decnet propagate static
decnet route (interface static route)
decnet route default (to enter a default route)
decnet route default (interface default route)
show decnet static

decnet route default (to enter a default route)

Use this form of the **decnet route default** global configuration command to enter a specific default route. The **no** form of this command removes this route.

decnet route default *next-hop-address* [*hops* [*cost*]]
no decnet route default *next-hop-address*

Syntax Description

next-hop-address	This value is used to establish the next hop of the route for forwarding packets.
hops	(Optional) Hop count to be associated with the route being advertised. Default is 0.
cost	(Optional) Cost to be associated with the route being advertised. Default is 0.

Default

No interface default routes are created.

Command Mode

Global configuration

Usage Guidelines

A default route will be used only after DECnet conversion is checked. By default, DECnet static routes will not be propagated to other routers. Use the **decnet propagate static** command to enable propagation.

DECnet packets not for the current area are forwarded to the *next-hop-address*.

Examples

The following example shows how to create a default route for 1.3 which uses default values of 0 for hops and cost:

```
decnet route default 1.3
```

Related Commands

decnet propagate static
decnet route (interface static route)
decnet route (to enter a static route)
decnet route default (interface default route)
show decnet static

decnet routing

To enable DECnet routing, use the **decnet routing** global configuration command. To disable DECnet routing, use the **no** form of this command.

decnet [*network-number*] **routing** [**iv-prime**] *decnet-address*
no decnet routing

Syntax Description

network-number	(Optional) Network number in the range 0 through 3. Specified when using Address Translation Gateway (ATG). If not specified, the default is network 0.
iv-prime	(Optional) Enables DECnet Phase IV Prime routing.
decnet-address	Address in DECnet format X.Y, where X is the area number and Y is the node number.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

Enabling DECnet changes the MAC addresses of the router's interfaces. This is not a problem on routers equipped with nonvolatile memory. On systems that attempt to get their IP network addresses from network servers rather than from nonvolatile memory, there may be a problem as with the hardware addresses changing and confusing other IP-speaking hosts. This potential problem can be avoided by configuring and enabling DECnet before enabling other protocols.

Note You can configure up to four DECnet networks (numbered 0 through 3). To set up multiple DECnet networks, use the **decnet** global configuration commands with the appropriate network number and keywords. If the network number is omitted from the commands, network 0 will be configured for DECnet routing.

DECnet Phase IV Prime eliminates the DEC addressing restrictions so that DECnet nodes can coexist with systems running other protocols that have other MAC address restrictions. If **iv-prime** is not specified, only Phase IV will be enabled; configuring the MAC address will then make DECnet inoperable. The standard "AA-00-04-00" form will be set as the address of the interface on which DECnet is enabled. If Phase IV Prime was already running and this command is reissued without the **iv-prime** keyword (that is, going from Phase IV Prime to Phase IV), the command will return an error if any of the interfaces that have DECnet enabled have MAC addresses that are not compliant with DECnet Phase IV, requiring the user to evaluate conflicting interface commands.

The **no** form of this command will disable Phase IV and Phase IV Prime routing.

Example

In the following example, DECnet routing is enabled for the router in area 21 with node number 456:

```
decnet routing 21.456
```

Related Commands

decnet cost
decnet node-type

decnet routing-timer

To specify how often the router sends routing updates that list the hosts that the router can reach, use the **decnet routing-timer** interface configuration command. Use the **no** form of this command to disable the routing update timer.

decnet routing-timer *seconds*
no decnet routing-timer

Syntax Description

seconds	Time, in seconds, from 1 through 65535. The default is 40 seconds.

Default

40 seconds

Command Mode

Interface configuration

Usage Guidelines

Other routers use this information to construct local routing tables. In a network where changes occur infrequently or do not need to be responded to immediately (it is small and uncomplicated, applications are not particularly sensitive to delays or occasional packet loss, slow serial links, and so on), increasing the time between routing updates reduces the amount of unnecessary network traffic. Digital calls this argument the *broadcast routing timer* because they use a different timer for serial lines; our DECnet implementation does not make this distinction.

Example

In the following example, a serial interface is set to broadcast routing updates every 2 minutes (120 seconds):

```
interface serial 0
decnet routing-timer 120
```

lat host-delay

To set the delayed acknowledgment for incoming LAT slave connections, use the **lat host-delay** global configuration command. To restore the default, use the **no** form of this command.

lat host-delay *number*
no host-delay

Syntax Description

number	The delay in milliseconds.

Default

Disabled

Command Mode

Global configuration

Example

The following example sets the acknowledgment for incoming LAT slave connections to 100 milliseconds:

```
lat host-delay 100
```

lat service autocommand

To associate a command with a service, use the **lat service autocommand** global configuration command. To remove the specified autocommand, use the **no** form of this command.

lat service *service-name* **autocommand** *command*
no lat service *service-name* **autocommand** *command*

Syntax Description

service-name	Name of the service.
command	Command to be associated with the service.

Default

No commands are automatically associated with a service.

Command Mode

Global configuration

Usage Guidelines

When an inbound connection is received for the specified service, the command associated with the service is automatically executed instead of the user receiving a virtual terminal session.

Authentication is bypassed for these services; only the LAT password is checked.

Note Do not use this option with the **rotary** keyword.

Example

The following example associates the command **telnet readings** to the service *readings*:

```
lat service readings autocommand telnet readings
```

ping (privileged)

Use the DECnet **ping** privileged EXEC command to send DECnet echo packets to test the reachability of a remote host over a DECnet network.

ping

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Usage Guidelines

To abort a ping session, type the escape sequence (by default, Ctrl-^ X, which is done by simultaneously pressing the Ctrl, Shift, and 6 keys, letting go, then pressing the X key).

Table 16-3 describes the test characters that the ping facility sends.

Table 16-3 Ping Test Characters

Character	Description
!	Each exclamation point indicates receipt of a reply.
.	Each period indicates the network server timed out while waiting for a reply.
U	A destination unreachable error PDU was received.
C	A congestion-experienced packet was received.
I	User interrupted test.
?	Unknown packet type.
&	Packet lifetime exceeded.

Sample DECnet Display Using a DECnet Address

The following display shows a sample DECnet **ping** session that uses a DECnet address to specify the source:

```
Router# ping
Protocol [ip]: decnet
Target DECnet address: 2.16
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Type escape sequence to abort.
Sending 5, 100-byte DECnet Echos to 2.16,
timeout is 2 seconds:
!!!!!
Success rate is 100 percent, round-trip min/avg/max = 1/4/8 ms
```

Table 16-4 describes the fields shown in the display.

Table 16-4 Ping Field Descriptions

Field	Description
Protocol [ip]:	Default is IP.
Target DECnet address:	Prompts for the DECnet address of the destination node you plan to ping.
Repeat count [5]:	Number of ping packets that will be sent to the destination address. Default: 5.
Datagram size [100]:	Size of the ping packet (in bytes). Default: 100 bytes.
Timeout in seconds [2]:	Timeout interval (in seconds). Default: 2 seconds.

Related Command

ping (user)

ping (user)

Use the DECnet **ping** user EXEC command to send DECnet echo packets to test the reachability of a remote host over a DECnet network.

ping decnet {*host* | *address*}

Syntax Description

decnet	DECnet protocol keyword.
host	DECnet host of system to ping.
address	DECnet address of system to ping.

Command Mode

EXEC

Usage Guidelines

The **ping** Exec command provides a basic user ping facility for DECnet users who do not have system privileges. This feature allows the router to perform the simple default ping functionality for the DECnet protocol. Only the nonverbose form of the **ping** command is supported for user-level pings.

To abort a ping session, type the escape sequence (by default, Ctrl-^ X, which is done by simultaneously pressing the Ctrl, Shift, and 6 keys, letting go, then pressing the X key).

Table 16-5 describes the test characters that the ping facility sends.

Table 16-5 Ping Test Characters

Character	Description
!	Each exclamation point indicates receipt of a reply.
.	Each period indicates the network server timed out while waiting for a reply.
U	A destination unreachable error PDU was received.
C	A congestion-experienced packet was received.
I	User interrupted test.
?	Unknown packet type.
&	Packet lifetime exceeded.

Sample DECnet Display Using a DECnet Address

The following display shows sample ping output when you ping the DECnet address of 2.16:

```
router> ping decnet 2.16
Sending 5, 100-byte DECnet Echos to 2.16,
timeout is 2 seconds:
!!!!!
Success rate is 100 percent, round-trip min/avg/max = 1/4/8 ms
```

Related Command

ping (privileged)

show decnet

Use the **show decnet** privileged EXEC command to display the global DECnet parameters.

show decnet

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Sample Display

The following is sample output from the **show decnet** command:

```
Router# show decnet

Global DECnet parameters for network 0:
   Local address is 19.15, node type is area (Phase-IV Prime)
   Level-2 'Attached' flag is FALSE,  nearest level-2 router is 19.5
   Maximum node is 350, maximum area is 63, maximum visits is 63
   Maximum paths is 1, path split mode is normal
   Local maximum cost is 1022, maximum hops is 30
   Area maximum cost is 1022, maximum hops is 30
   Static routes *NOT* being sent in routing updates
   Default route configured; next hop address of 2.100
```

Table 16-6 describes significant fields shown in the display.

Table 16-6 Show DECnet Field Descriptions

Field	Description
Global DECnet parameters for network 0:	Indicates the DECnet network number of the network being described.
Local address is 19.15	DECnet address of the router.
node type is area	Indicates the DECnet node type with which the interface has been configured. Possible values include area (area router) or routing-iv (intra-area router).
Level-2 'Attached' flag is FALSE	(DECnet Level-2 routers only) Indicates that this Level-2 router is not "attached" (does not have reachability to other DECnet Phase IV areas). If the 'Attached' flag is TRUE, the router has reachability to other areas. If the 'Attached' flag is FALSE, other displays on this line are the following: • Nearest Level-2 router is NONE—(DECnet Level-1 routers only) Indicates that this Level-1 router has not heard from any eligible Level-2 router (to send out-of-area packets to) • Nearest Level-2 router is 1.200—(DECnet Level-1 routers only) Indicates that this router's nearest Level-2 router is 1.200. Any packets received by this router destined for other areas will be sent to 1.200.

Field	Description
(Phase-IV Prime)	Indicates that the router is running DECnet Phase IV Prime routing.
Maximum node is 350	Highest node number that the router will recognize.
maximum area is 63	Indicates the maximum DECnet area number, which is used to control the size of internal routing tables and messages sent to other routers. Range: 1 through 63. Default: 63.
maximum visits is 63	Indicates the maximum number of times (visits) a packet can pass through a router. Range: 1 through 63. Default: 63.
Maximum paths is 1	Indicates the maximum number of equal-cost paths the router will save. Range: 1 through 31. Default: 1.
path split mode is normal	Indicates how the router will split the routable packets among equal-cost paths. Possible values: normal (default) or interim.
Local maximum cost is 1022	For intra-area routes. Router ignores routes in its area that have a cost greater than this value.
maximum hops is 30	Indicates the maximum number of hops for a usable route within the local area. The router ignores routes within the local area that use more than this number of hops.
Area maximum cost is 1022	Indicates the maximum cost specification for interarea routing. The router ignores routes to other areas that have a cost greater than this value. Range: 1 through 1022: Default: 1022.
maximum hops is 30	Indicates the maximum number of hops for a usable route to other areas. The router ignores routes to other areas that use more than this number of hops.
Static routes *NOT* being sent in routing updates	Indicates static routes are not included in routing updates.
Default route configured; next hop address of 2.100	Indicates a default route is configured on this router and shows the next hop address.

show decnet interface

Use the **show decnet interface** EXEC command to display the global DECnet status and configuration for all interfaces, or the status and configuration for a specified interface.

show decnet interface [*type number*]

Syntax Description

type	(Optional) Interface type.
number	(Optional) Interface unit number.

Command Mode

EXEC

Sample Display

The following is sample output from the **show decnet interface** command when you do not specify an interface:

```
Router# show decnet interface e1
Global DECnet parameters for network 0:
  Local address is 19.15, node type is area
  Maximum node is 350, maximum area is 63, maximum visits is 63
  Maximum paths is 1, path split mode is normal
  Local maximum cost is 1022, maximum hops is 30
  Area maximum cost is 1022, maximum hops is 30
Ethernet 1 is up, line protocol is up, encapsulation is ARPA
  Interface cost is 4, priority is 64, DECnet network: 0
  The designated router is 1.9
  Sending HELLOs every 15 seconds, routing updates 40 seconds
  Smallest router blocksize seen is 1498 bytes
  Routing input list is not set, output list is not set
  Access list is not set
  DECnet fast switching is enabled
  Number of L1 router adjacencies is : 3
  Number of non-PhaseIV+ router adjacencies is : 3
  Number of PhaseIV+ router adjacencies is : 0
  Router is bilingual
```

Table 16-7 describes significant fields shown in the display.

Table 16-7 Show DECnet Interface Field Descriptions when an Interface Is Not Specified

Field	Description
Global DECnet parameters for network 0:	Indicates the DECnet network number of the network being described.
Local address is 19.15	DECnet address of the router.
node type is area	Indicates the DECnet node type with which the interface has been configured. Possible values include area (area router) or routing-iv (intra-area router).
Maximum node is 350	Highest node number that the router will recognize.

Field	Description
maximum area is 63	Indicates the maximum DECnet area number, which is used to control the size of internal routing tables and messages sent to other routers. Range: 1 through 63. Default: 63.
maximum visits is 63	Indicates the maximum number of times (visits) a packet can pass through a router. Range: 1 through 63. Default: 63.
Maximum paths is 1	Indicates the maximum number of equal-cost paths the router will save. Range: 1 through 31. Default: 1.
path split mode is normal	Indicates how the router will split the routable packets among equal-cost paths. Possible values: normal (default) or interim.
Local maximum cost is 1022	For intra-area routes. Router ignores routes in its area that have a cost greater than this value.
maximum hops is 30	Indicates the maximum number of hops for a usable route within the local area. The router ignores routes within the local area that use more than this number of hops.
Area maximum cost is 1022	Indicates the maximum cost specification for interarea routing. The router ignores routes to other areas that have a cost greater than this value. Range: 1 through 1022: Default: 1022.
maximum hops is 30	Indicates the maximum number of hops for a usable route to other areas. The router ignores routes to other areas that use more than this number of hops.
Ethernet 0 is up	Indicates whether the interface hardware is currently active and if it has been taken down by an administrator.
line protocol is up	Indicates whether the software processes that handle the line protocol believe the interface is usable (that is, whether keepalives are successful).
encapsulation is ARPA	Indicates the encapsulation type.
Interface cost is 4	Indicates the cost that has been assigned to this interface using the **decnet cost** interface configuration command. If there are multiple paths to a destination, the one with the lowest cost is selected.
priority is 64	Indicates the priority that has been assigned to this router on this interface. End systems will select the router with the highest priority as their designated router.
DECnet network: 0	Indicates that this interface is on DECnet network 0. This fact is significant only if Address Translation Gateway (ATG) is turned on.
The designated router is 1.3	Indicates the designated router on this particular LAN.
Sending HELLOs every 15 seconds	Indicates the frequency of hello packets.
routing updates 40 seconds	Indicates the frequency of routing updates.
Smallest router blocksize seen is 1498 bytes	Indicates the largest size of packets being sent on all routers on the LAN.
Routing input list is not set, output list is not set	Indicates that no access restrictions on incoming (or outgoing) router update or hello messages have been set for this interface.
Access list is not set	Indicates that no access lists have been configured for the interface.
DECnet fast switching is enabled	Indicates that fast switching is enabled.
Number of L1 router adjacencies is : 1	Indicates how many Level 1 adjacencies the router has on this interface.

Field	Description
Number of non-PhaseIV+ router adjacencies is : 3	Number of L1 and L2 routers on this interface that are not running Phase IV+.
Number of PhaseIV+ router adjacencies is : 0	Number of L2 routers on this interface that are running Phase IV+.
Router is bilingual	The router's MAC address on this interface is Phase IV-compatible (that is, it takes the form AA-00-04-00-xx-yy or 55-00-20-00-aa-bb on interfaces where the address is bit swapped). This means that the router will behave as both a Phase IV and a Phase IV Prime router.

Sample Display

The following is sample output from the **show decnet interface** command when you specify an interface:

```
Router# show decnet interface e 0
Ethernet0 is up, line protocol is up, encapsulation is ARPA
  Interface cost is 4, priority is 64, DECnet network: 0
  The designated router is 1.3
  Sending HELLOs every 15 seconds, routing updates 40 seconds
  Smallest router blocksize seen is 1498 bytes
  Routing input list is not set, output list is not set
  Access list is not set
  DECnet fast switching is enabled
  Number of L1 router adjacencies is : 1
  Number of non-PhaseIV+ router adjacencies is : 3
  Number of PhaseIV+ router adjacencies is : 0
  Router is bilingual
```

Table 16-8 describes significant fields shown in the display.

Table 16-8 Show DECnet Interface Field Descriptions when an Interface Is Specified

Field	Description
Ethernet 0 is up	Indicates whether the interface hardware is currently active and if it has been taken down by an administrator.
line protocol is up	Indicates whether the software processes that handle the line protocol believe the interface is usable (that is, whether keepalives are successful).
encapsulation is ARPA	Indicates the encapsulation type.
Interface cost is 4	Indicates the cost that has been assigned to this interface using the **decnet cost** interface configuration command. If there are multiple paths to a destination, the one with the lowest cost is selected.
priority is 64	Indicates the priority that has been assigned to this router on this interface. End systems will select the router with the highest priority as their designated router.
DECnet network: 0	Indicates that this interface is on DECnet network 0. This fact is significant only if Address Translation Gateway (ATG) is turned on.
The designated router is 1.3	Indicates the designated router on this particular LAN.
Sending HELLOs every 15 seconds	Indicates the frequency of hello packets.
routing updates 40 seconds	Indicates the frequency of routing updates.

Field	Description
Smallest router blocksize seen is 1498 bytes	Indicates the largest size of packets being sent on all routers on the LAN.
Routing input list is not set, output list is not set	Indicates that no access restrictions on incoming (or outgoing) router update or hello messages have been set for this interface.
Access list is not set	Indicates that no access lists have been configured for the interface.
DECnet fast switching is enabled	Indicates that fast switching is enabled.
Number of L1 router adjacencies is : 1	Indicates how many Level 1 adjacencies the router has on this interface.
Number of non-PhaseIV+ router adjacencies is : 3	Number of L1 and L2 routers on this interface that are not running Phase IV+.
Number of PhaseIV+ router adjacencies is : 0	Number of L2 routers on this interface that are running Phase IV+.
Router is bilingual	The router's MAC address on this interface is Phase IV-compatible (that is, it takes the form AA-00-04-00-xx-yy or 55-00-20-00-aa-bb on interfaces where the address is bit swapped). This means that the router will behave as both a Phase IV and a Phase IV Prime router.

show decnet map

Use the **show decnet map** EXEC command to display the address mapping information used by the DECnet Address Translation Gateway.

show decnet map

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show decnet map** command:

```
Router# show decnet map

Net Node   -> Net Node    Uses     Cost Hops
  0 1.100       1 2.100   0
```

Table 16-9 describes significant fields shown in the display.

Table 16-9 Show DECnet Map Field Descriptions

Field	Description
Net Node - Net Node	Net number and node address.
Uses	Number of times this map was used.
Cost	Cost associated with route.
Hop	Number of hops to destination mode.

show decnet neighbors

Use the **show decnet neighbors** privileged EXEC command to display all Phase IV and Phase IV Prime adjacencies and the MAC address associated with each neighbor.

show decnet neighbors

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Sample Display

The following is sample output from the **show decnet neighbors** command:

```
Router# show decnet neighbors
Net  Node     Interface     MAC address     Flags
0    3.11      Ethernet0     aa00.0400.0b0c  A
0    1.1       Ethernet0     aa00.0400.0104  V
0    1.3       Ethernet1     aa00.0400.0304  V
0    1.6       Ethernet1     aa00.0400.0604  V
0    2.2        TokenRing    5500.2000.4020  V IV-PRIME
```

Table 16-10 describes the fields shown in the display.

Table 16-10 Show DECnet Neighbors Field Descriptions

Field	Description
Net	Number of the DECnet network this adjacency is in.
Node	DECnet address of the adjacency.
Interface	Interface over which this adjacency was heard.
MAC address	MAC address that this adjacency is using on this interface.
Flags	A: L2 adjacency. V: L1 adjacency. IV-PRIME: DECnet Phase IV Prime adjacency.

show decnet route

Use the **show decnet route** EXEC command to display the DECnet routing table.

show decnet route [*decnet-address*]

Syntax Description

decnet-address	(Optional) DECnet address and, when specified, the first hop route to that address is displayed.

Command Mode

EXEC

Sample Display

The following is sample output from the **show decnet route** command when a DECnet address name was not specified, so the entire routing table is displayed:

```
Router# show decnet route

  Area        Cost  Hops   Next Hop to Node       Expires  Prio

 1              4     1   Ethernet1 -> 1.300         26     64    A
*1              4     1   Ethernet1 -> 1.400         37     64    A
*2              8     2   Ethernet1 -> 1.400
*5              0     0     (Local) -> 5.5
*10             4     1   Ethernet2 -> 10.1          36     64    A
*13            11     3   Ethernet1 -> 1.400
*44            22     6   Ethernet1 -> 1.400
*51            18     4   Ethernet1 -> 1.400
*61             1     1       (OSI) -> 5.5
*62             1     1       (OSI) -> 5.5
*3              0     0    (STATIC)   Ethernet0, snpa aa00.0400.0404
*4              0     0    (STATIC)     Serial0
*6              5     4    (STATIC)  forwarding to 2.3

  Node        Cost  Hops   Next Hop to Node       Expires  Prio

*(Area)         0     0     (Local) -> 5.5
*5.5            0     0     (Local) -> 5.5           32     64    A+

*DEFAULT* :     0     0     using next hop address of 2.100
Router#
```

As the display shows, the **show decnet route** command can display more than one route for a destination when equal-cost paths have been set with the **decnet max-paths** global configuration command, and when there is more than one equal-cost path to a destination. The display also shows that this node is an area router.

Table 16-11 describes significant fields shown in the display.

Table 16-11 Show DECnet Route Field Descriptions

Field	Description
*	Currently selected route for a particular destination. In interim mode, the selected route will never appear to change.
Node	DECnet address of this (reachable) destination.
(Area)	All Level 1 routes are displayed in this section except for this the first entry, which points to the nearest Level 2 router.
Cost	Assigned cost for the interface, based on a recommended value for the underlying media. Range: 1 through 63. No default.
Hops	Number of hops to this node from the router being monitored.
Next Hop to Node	DECnet address of the next hop a packet will take to get to the final destination as well as the interface.
(Local)	The address that the router is configured with.
(OSI)	Indicates that this entry was created by the **decnet advertise** command.
(STATIC)	Indicates that this entry was created by the **decnet route** command.
Expires	Displays how many seconds from now this entry expires.
Prio	Router priority of this node.
V	Adjacent Level 1 router.
A+	Adjacent Level 2 (area) router; A indicates that this is an adjacency created from a Phase IV hello, A+ indicates that thisis an adjacency created from a Phase IV+ hello.

show decnet static

Use the **show decnet static** privileged EXEC command to display all statically configured DECnet routes.

show decnet static

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Usage Guidelines

All static rotues are stored in a static route queue, which allows static routes to be reinstated when DECnet routing is turned off then on again.

Not all routes in the static route queue will show up in the routing table. This will happen under the following conditions:

- The router is a Level 1 router and any of the following apply. Assume the router DECnet address is 1.1:
  - — A Level 2 area static route is configured.

```
decnet route 2.0 1.2
```

  - — A static route is configured not in the same area as the router.

```
decnet route 3.10 1.200
```

  - — A static route is configured for the same address as the router.

```
decnet route 1.1 1.200
```

- The router is a Level 2 router and any of the following apply. Assume the router DECnet address is 2.1:
  - — A Level 1 static router is not in the same area as the router.

```
decnet route 4.1 10.200
```

    Note that the following route will appear because a Level 2 route is installed to area 4:

```
decnet route 4.0 10.200
```

  - — A Level 2 static route is configured for the router's own area.

```
decnet route 2.0 10.200
```

  - — A static route is configured for the same address as the router.

```
decnet route 2.1 5.4 s 1
```

Sample Display

The following is sample output from the **show decnet static** command:

```
Router# show decnet static

Address      Cost    Hops    Next hop         SNPA

 3             0       0     Ethernet0  aa00.0400.0404
 5             0       0       Serial0
 5.100         0       0       Serial2
 DEFAULT       0       0        2.100
 6             5       4        2.3
```

Note that this router is a Level 2 router with DECnet address of 1.2, so a static route configured for 5.100 is not relevant here. This route appears in the **show decnet static** display, but not in the routing table.

show decnet traffic

The **show decnet traffic** EXEC command shows the DECnet traffic statistics, including datagrams sent, received, and forwarded.

show decnet traffic

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show decnet traffic** command:

```
Router# show decnet traffic

Total: 42 received, 0 format errors, 0 unimplemented
0 not a gateway, 0 no memory, 0 no routing vector
0 congestion encountered
Hellos: 21 received, 0 bad, 0 other area, 16 sent
Level 1 routing: 14 received, 0 bad, 0 other area, 16 sent
Level 2 routing: 7 received, 0 not primary router, 8 sent
Data: 0 received, 0 not long format, 0 too many visits
0 forwarded, 0 returned, 0 converted, 0 local destination
0 access control failed, 0 no route, 0 encapsulation failed
0 inactive network, 0 incomplete map
```

Table 16-12 describes the fields shown in the display.

Table 16-12 Show DECnet Traffic Field Descriptions

Field	Description
Total:	Displays the totals of packet types received.
received	Total of all types of DECnet packets received.
format errors	Lists the number of packets that appeared to be DECnet, but were formatted incorrectly. The number in the received field includes these packets.
0 unimplemented	Reports the number of incoming packets that are DECnet control packets, and how many specify a service that the router does not implement. This includes services implemented to forward Level 1 and Level 2 routing information, and router and end-system hello packets.
0 not a gateway	Reports the total number of packets received while not routing DECnet.
0 no memory	Records transaction attempts when the system has run out of memory.
0 no routing vector	Indicates that either a routing update came in from another router when the router did not have an adjacency for it, or it had no routing vector for the type of routing update. Use the **debug decnet-routing** EXEC command for more information.
0 congestion encountered	Number of times the underlying physical layer detected congestion.
HELLOs:	Displays the number of hello messages received and sent.

Field	Description
received	Displays the total number of hello messages received. All protocol types are included.
bad	Displays the total number of "bad" hello messages received. Invoke the EXEC command **debug decnet** to display more information about why the hello message was judged as bad.
other area	Displays the total number of hello messages received from nodes on other areas when the router is a Level 1 router only.
sent	Displays the total number of hello messages sent.
Level 1 routing:	Displays the Level 1 routing updates received and sent.
received	Displays the total number of Level 1 routing updates received.
bad	Displays the total number of Level 1 updates received that were judged to be bad.
other area	Displays the total number of Level 1 updates from nodes in other areas.
sent	Displays the total number of Level 1 updates sent.
Level 2 routing:	Displays the Level 2 routing updates received and sent.
received	Displays the total number of Level 2 updates received.
not primary router	Should always be zero.
sent	Displays the total number of Level 2 updates sent.
Data:	Displays the number of data packets received and sent.
received	Displays the total number of noncontrol (data) packets received.
not long format	Displays the number of packets received which are not in the long DECnet format. This number should always be zero. If it is not, investigate the source of the improperly formatted packets.
too many visits	Lists the number of packets received which have visited too many routers and have been flushed.
forwarded	Lists the total number of packets forwarded.
returned	Lists the total number of packets returned to the sender at the senders' request.
converted	Displays the number of Phase IV packets converted to Phase V packets.
local destination	Packets received that are destined for this router.
access control failed	Lists the packets dropped because access control required it.
no route	Lists the total packets dropped because the router did not know where to forward them.
encapsulation failed	Lists the number of packets that could not be encapsulated. This usually happens where there are entries missing in a map for a public data network, such as X.25 or Frame Relay. This can also occur if an interface is set for an encapsulation for which there is no defined DECnet encapsulation (such as PPP on serial interfaces).
inactive network	Displays the number of packets that appear to come from a known interface, or that ATG returned because they did not make sense.
incomplete map	Counts the number of packets that failed address translation. This usually means a node that is not in the ATG map is trying to access a node in another network advertised by the ATG.

CHAPTER 17

IP Commands

The Internet Protocol (IP) is a packet-based protocol used to exchange data over computer networks. IP handles addressing, fragmentation, reassembly, and protocol demultiplexing. It is the foundation on which all other Internet protocols, collectively referred to as the Internet Protocol suite, are built. IP is a network-layer protocol that contains addressing information and some control information that allows data packets to be routed.

The Transmission Control Protocol (TCP) is built upon the IP layer. TCP is a connection-oriented protocol that specifies the format of data and acknowledgments used in the transfer of data. TCP also specifies the procedures that the computers use to ensure that the data arrives correctly. TCP allows multiple applications on a system to communicate concurrently because it handles all demultiplexing of the incoming traffic among the application programs.

Use the commands in this chapter to configure and monitor IP networks. For IP protocol configuration information and examples, refer to the "Configuring IP" chapter of the *Router Products Configuration Guide*.

access-class

To restrict incoming and outgoing connections between a particular virtual terminal line (into a Cisco device) and the addresses in an access list, use the **access-class** line configuration command. To remove access restrictions, use the **no** form of this command.

access-class *access-list-number* {**in** | **out**}
no access-class *access-list-number* {**in** | **out**}

Syntax Description

access-list-number	Number of an access list. This is a decimal number from 1 through 99.
in	Restricts incoming connections between a particular Cisco device and the addresses in the access list.
out	Restricts outgoing connections between a particular Cisco device and the addresses in the access list.

Default

No access lists are defined.

Command Mode

Line configuration

Usage Guidelines

Remember to set *identical restrictions* on all the virtual terminal lines because a user can connect to any of them.

To display the access lists for a particular terminal line, use the **show line** EXEC command and specify the line number.

Examples

The following example defines an access list that permits only hosts on network 192.89.55.0 to connect to the virtual terminal ports on the router:

```
access-list 12 permit 192.89.55.0  0.0.0.255
line 1 5
access-class 12 in
```

The following example defines an access list that denies connections to networks other than network 36.0.0.0 on terminal lines 1 through 5:

```
access-list 10 permit 36.0.0.0 0.255.255.255
line 1 5
access-class 10 out
```

Related Command

A dagger (†) indicates that the command is documented in another chapter.

show line †

access-list (standard)

To define a standard IP access list, use the standard version of the **access-list** global configuration command. To remove a standard access lists, use the **no** form of this command.

access-list *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
no access-list *access-list-number*

Caution Enhancements to this command are backward compatible; migrating from existing releases to Release 10.3 will convert your access lists automatically. However, releases prior to Release 10.3 are not upwardly compatible with these enhancements. Therefore, if you save an access list with these images and then use software prior to Release 10.3, the resulting access list will not be interpreted correctly. **This could cause you severe security problems.** Save your old configuration file before booting these images.

Syntax Description

access-list-number	Number of an access list. This is a decimal number from 1 through 99.
deny	Denies access if the conditions are matched.
permit	Permits access if the conditions are matched.
source	Number of the network or host from which the packet is being sent. There are two alternative ways to specify the source: • Use a 32-bit quantity in four-part dotted-decimal format. • Use the keyword **any** as an abbreviation for a *source* and *source-wildcard* of 0.0.0.0 255.255.255.255.
source-wildcard	(Optional) Wildcard bits to be applied to the *source*. There are two alternative ways to specify the source wildcard: • Use a 32-bit quantity in four-part dotted-decimal format. Place ones in the bit positions you want to ignore. • Use the keyword **any** as an abbreviation for a *source* and *source-wildcard* of 0.0.0.0 255.255.255.255.

Default

The access list defaults to an implicit deny statement for everything. The access list is always terminated by an implicit deny statement for everything.

Command Mode

Global configuration

Usage Guidelines

Plan your access conditions carefully and be aware of the implicit deny statement at the end of the access list.

You can use access lists to control the transmission of packets on an interface, control virtual terminal line access, and restrict the contents of routing updates.

Use the **show access-lists** EXEC command to display the contents of all access lists.

Use the **show ip access-list** EXEC command to display the contents of one access list.

Examples

The following example of a standard access list allows access for only those hosts on the three specified networks. The wildcard bits apply to the host portions of the network addresses. Any host with a source address that does not match the access list statements will be rejected.

```
access-list 1 permit 192.5.34.0  0.0.0.255
access-list 1 permit 128.88.0.0  0.0.255.255
access-list 1 permit 36.0.0.0  0.255.255.255
! (Note: all other access implicitly denied)
```

To specify a large number of individual addresses more easily, you can omit the wildcard if it is all zeros. Thus, the following two configuration commands are identical in effect:

```
access-list 2 permit 36.48.0.3
access-list 2 permit 36.48.0.3  0.0.0.0
```

Related Commands

access-class
access-list (extended)
distribute-list in
distribute-list out
ip access-group
priority-list
queue-list
show access-lists
show ip access-list

access-list (extended)

To define an extended IP access list, use the extended version of the **access-list** global configuration command. To remove the access lists, use the **no** form of this command.

access-list *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** *tos*]
no access-list *access-list-number*

For ICMP, you can also use the following syntax:

access-list *access-list-number* {**deny** | **permit**} **icmp** *source source-wildcard destination destination-wildcard* [*icmp-type* [*icmp-code*] | *icmp-message*] [**precedence** *precedence*] [**tos** *tos*]

For IGMP, you can also use the following syntax:

access-list *access-list-number* {**deny** | **permit**} **igmp** *source source-wildcard destination destination-wildcard* [*igmp-type*] [**precedence** *precedence*] [**tos** *tos*]

For TCP, you can also use the following syntax:

access-list *access-list-number* {**deny** | **permit**} **tcp** *source source-wildcard* [*operator port* [*port*]] *destination destination-wildcard* [*operator port* [*port*]] [**established**] [**precedence** *precedence*] [**tos** *tos*]

For UDP, you can also use the following syntax:

access-list *access-list-number* {**deny** | **permit**} **udp** *source source-wildcard* [*operator port* [*port*]] *destination destination-wildcard* [*operator port* [*port*]] [**precedence** *precedence*] [**tos** *tos*]

Caution Enhancements to this command are backward compatible; migrating from existing releases to Release 10.3 will convert your access lists automatically. However, releases prior to Release 10.3 are not upwardly compatible with these enhancements. Therefore, if you save an access list with these images and then use software prior to Release 10.3, the resulting access list will not be interpreted correctly. **This could cause you severe security problems.** Save your old configuration file before booting these images.

Syntax Description

access-list-number	Number of an access list. This is a decimal number from 100 through 199.
deny	Denies access if the conditions are matched.
permit	Permits access if the conditions are matched.
protocol	Name or number of an IP protocol. It can be one of the keywords **eigrp**, **gre**, **icmp**, **igmp**, **igrp**, **ip**, **ipinip**, **nos**, **ospf**, **tcp**, or **udp**, or an integer in the range 0 through 255 representing an IP protocol number. To match any Internet protocol, including ICMP, TCP, and UDP, use the keyword **ip**. Some protocols allow further qualifiers described below.

source	Number of the network or host from which the packet is being sent. There are three alternative ways to specify the source: • Use a 32-bit quantity in four-part dotted-decimal format. • Use the keyword **any** as an abbreviation for a *source* and *source-wildcard* of 0.0.0.0 255.255.255.255. • Use **host** *source* as an abbreviation for a *source* and *source-wildcard* of *source* 0.0.0.0.
source-wildcard	Wildcard bits to be applied to source. There are three alternative ways to specify the source wildcard: • Use a 32-bit quantity in four-part dotted-decimal format. Place ones in the bit positions you want to ignore. • Use the keyword **any** as an abbreviation for a *source* and *source-wildcard* of 0.0.0.0 255.255.255.255. • Use **host** *source* as an abbreviation for a *source* and *source-wildcard* of *source* 0.0.0.0.
destination	Number of the network or host to which the packet is being sent. There are three alternative ways to specify the destination: • Use a 32-bit quantity in four-part dotted-decimal format. • Use the keyword **any** as an abbreviation for the *destination* and *destination-wildcard* of 0.0.0.0 255.255.255.255. • Use **host** *destination* as an abbreviation for a *destination* and *destination-wildcard* of *destination* 0.0.0.0.
destination-wildcard	Wildcard bits to be applied to the destination. There are three alternative ways to specify the destination wildcard: • Use a 32-bit quantity in four-part dotted-decimal format. Place ones in the bit positions you want to ignore. • Use the keyword **any** as an abbreviation for a *destination* and *destination-wildcard* of 0.0.0.0 255.255.255.255. • Use **host** *destination* as an abbreviation for a *destination* and *destination-wildcard* of *destination* 0.0.0.0.
precedence *precedence*	(Optional) Packets can be filtered by precedence level, as specified by a number from 0 to 7 or by name as listed in the section "Usage Guidelines."
tos *tos*	(Optional) Packets can be filtered by type of service level, as specified by a number from 0 to 15 or by name as listed in the section "Usage Guidelines."
icmp-type	(Optional) ICMP packets can be filtered by ICMP message type. The type is a number from 0 to 255.
icmp-code	(Optional) ICMP packets which are filtered by ICMP message type can also be filtered by the ICMP message code. The code is a number from 0 to 255.

icmp-message	(Optional) ICMP packets can be filtered by an ICMP message type name or ICMP message type and code name. The possible names are found in the section "Usage Guidelines."
igmp-type	(Optional) IGMP packets can be filtered by IGMP message type or message name. A message type is a number from 0 to 15. IGMP message names are listed in the section "Usage Guidelines."
operator	(Optional) Compares source or destination ports. Possible operands include **lt** (less than), **gt** (greater than), **eq** (equal), **neq** (not equal), and **range** (inclusive range). If the operator is positioned after the *source* and *source-wildcard*, it must match the source port. If the operator is positioned after the *destination* and *destination-wildcard*, it must match the destination port. The **range** operator requires two port numbers. All other operators require one port number.
port	(Optional) The decimal number or name of a TCP or UDP port. A port number is a number from 0 to 65535. TCP port names are listed in the section "Usage Guidelines." TCP port names can only be used when filtering TCP. UDP port names are listed in the section "Usage Guidelines." UDP port names can only be used when filtering UDP. TCP port names can only be used when filtering TCP. UDP port names can only be used when filtering UDP.
established	(Optional) For the TCP protocol only: Indicates an established connection. A match occurs if the TCP datagram has the ACK or RST bits set. The nonmatching case is that of the initial TCP datagram to form a connection.

Default

An extended access list defaults to a list that denies everything. An extended access list is terminated by an implicit deny statement.

Command Mode

Global configuration

Usage Guidelines

You can use access lists to control the transmission of packets on an interface, control virtual terminal line access, and restrict contents of routing updates. The router stops checking the extended access list after a match occurs.

Fragmented IP packets, other than the initial fragment, are immediately accepted by any extended IP access list. Extended access lists used to control virtual terminal line access or restrict contents of routing updates must not match against the TCP source port, the type of service value, or the packet's precedence.

Note After an access list is created initially, any subsequent additions (possibly entered from the terminal) are placed at the end of the list. In other words, you cannot selectively add or remove access list command lines from a specific access list.

The following is a list of precedence names:

- **critical**
- **flash**
- **flash-override**
- **immediate**
- **internet**
- **network**
- **priority**
- **routine**

The following is a list of type of service (tos) names:

- **max-reliability**
- **max-throughput**
- **min-delay**
- **min-monetary-cost**
- **normal**

The following is a list of ICMP message type names and ICMP message type and code names:

- **administratively-prohibited**
- **alternate-address**
- **conversion-error**
- **dod-host-prohibited**
- **dod-net-prohibited**
- **echo**
- **echo-reply**
- **general-parameter-problem**
- **host-isolated**
- **host-precedence-unreachable**
- **host-redirect**
- **host-tos-redirect**

- **host-tos-unreachable**
- **host-unknown**
- **host-unreachable**
- **information-reply**
- **information-request**
- **mask-reply**
- **mask-request**
- **mobile-redirect**
- **net-redirect**
- **net-tos-redirect**
- **net-tos-unreachable**
- **net-unreachable**
- **network-unknown**
- **no-room-for-option**
- **option-missing**
- **packet-too-big**
- **parameter-problem**
- **port-unreachable**
- **precedence-unreachable**
- **protocol-unreachable**
- **reassembly-timeout**
- **redirect**
- **router-advertisement**
- **router-solicitation**
- **source-quench**
- **source-route-failed**
- **time-exceeded**
- **timestamp-reply**
- **timestamp-request**
- **traceroute**
- **ttl-exceeded**
- **unreachable**

The following is a list of IGMP message names:

- **dvmrp**
- **host-query**
- **host-report**
- **pim**
- **trace**

The following is a list of TCP port names that can be used instead of port numbers. Refer to the current Assigned Numbers RFC to find a reference to these protocols. Port numbers corresponding to these protocols can also be found by typing a **?** in the place of a port number.

- **bgp**
- **chargen**
- **daytime**
- **discard**
- **domain**
- **echo**
- **finger**
- **ftp**
- **ftp-data**
- **gopher**
- **hostname**
- **irc**
- **klogin**
- **kshell**
- **lpd**
- **nntp**
- **pop2**
- **pop3**
- **smtp**
- **sunrpc**
- **syslog**
- **tacacs-ds**
- **talk**
- **telnet**
- **time**
- **uucp**
- **whois**
- **www**

The following is a list of UDP port names that can be used instead of port numbers. Refer to the current Assigned Numbers RFC to find a reference to these protocols. Port numbers corresponding to these protocols can also be found by typing a **?** in the place of a port number.

- **biff**
- **bootpc**
- **bootps**
- **discard**
- **dns**
- **dnsix**
- **echo**
- **mobile-ip**
- **nameserver**
- **netbios-dgm**
- **netbios-ns**
- **ntp**
- **rip**
- **snmp**
- **snmptrap**
- **sunrpc**
- **syslog**
- **tacacs-ds**
- **talk**
- **tftp**
- **time**
- **who**
- **xdmcp**

Examples

In the following example, serial interface 0 is part of a Class B network with the address 128.88.0.0, and the mail host's address is 128.88.1.2. The keyword **established** is used only for the TCP protocol to indicate an established connection. A match occurs if the TCP datagram has the ACK or RST bits set, which indicate that the packet belongs to an existing connection.

```
access-list 102 permit tcp 0.0.0.0 255.255.255.255 128.88.0.0 0.0.255.255 established
access-list 102 permit tcp 0.0.0.0 255.255.255.255 128.88.1.2 0.0.0.0 eq 25
interface serial 0
ip access-group 102 in
```

The following example also permit DNS packets and ICMP echo and echo reply packets:

```
access-list 102 permit tcp any 128.88.0.0 0.0.255.255 established
access-list 102 permit tcp any host 128.88.1.2 eq smtp
access-list 102 permit tcp any any eq domain
access-list 102 permit udp any any eq domain
access-list 102 permit icmp any any echo
access-list 102 permit icmp any any echo-reply
```

Related Commands

access-class
access-list (standard)
distribute-list in
distribute-list out
ip access-group
priority-list
queue-list
show access-lists
show ip access-list

arp (global)

To add a permanent entry in the ARP cache, use the **arp** global configuration command. To remove an entry from the ARP cache, use the **no** form of this command.

arp *ip-address hardware-address type* [**alias**]
no arp *ip-address hardware-address type* [**alias**]

Syntax Description

ip-address	IP address in four-part dotted-decimal format corresponding to the local data link address.
hardware-address	Local data link address (a 48-bit address).
type	Encapsulation description. For Ethernet interfaces, this is typically the **arpa** keyword. For FDDI and Token Ring interfaces, this is always **snap**.
alias	(Optional) Indicates that the router should respond to ARP requests as if it were the owner of the specified address.

Default

No entries are permanently installed in the ARP cache.

Command Mode

Global configuration

Usage Guidelines

The router uses ARP cache entries to translate 32-bit Internet Protocol addresses into 48-bit hardware addresses.

Because most hosts support dynamic resolution, you generally do not need to specify static ARP cache entries.

To remove all nonstatic entries from the ARP cache, use the **clear arp-cache** privileged EXEC command**.**

Example

The following is an example of a static ARP entry for a typical Ethernet host:

```
arp 192.31.7.19 0800.0900.1834 arpa
```

Related Command

clear arp-cache

arp (interface)

To control the interface-specific handling of IP address resolution into 48-bit Ethernet, FDDI, and Token Ring hardware addresses, use the **arp** interface configuration command. To disable an encapsulation type, use the **no** form of this command.

arp {**arpa** | **probe** | **snap**}
no arp {**arpa** | **probe** | **snap**}

Syntax Description

arpa	Standard Ethernet-style ARP (RFC 826).
probe	HP Probe protocol for IEEE-802.3 networks.
snap	ARP packets conforming to RFC 1042.

Default

Standard Ethernet-style ARP

Command Mode

Interface configuration

Usage Guidelines

Unlike most commands that take multiple arguments, arguments to the **arp** command are not mutually exclusive. Each command enables or disables a specific type of ARP. For example, if you enter the **arp arpa** command followed by the **arp probe** command, the router would send three (two for **probe** and one for **arpa**) packets each time it needed to discover a MAC address.

The **arp probe** command allows the router to use the Probe protocol (in addition to ARP) whenever it attempts to resolve an IEEE-802.3 or Ethernet local data link address. The subset of Probe that performs address resolution is called Virtual Address Request and Reply. Using Probe, the router can communicate transparently with Hewlett-Packard IEEE-802.3 hosts that use this type of data encapsulation.

Note Cisco's support for HP Probe proxy support changed as of Software Release 8.3(2) and subsequent software releases. The **no arp probe** command is now the default. All interfaces that will use Probe must now be explicitly configured for **arp probe**.

The **show interfaces** EXEC command displays the type of ARP being used on a particular interface. To remove all nonstatic entries from the ARP cache, use the **clear arp-cache** privileged EXEC command.

Example

The following example enables probe services:

```
interface ethernet 0
arp probe
```

Related Commands

clear arp-cache
show interfaces

arp timeout

To configure how long an entry remains in the ARP cache, use the **arp timeout** interface configuration command. To restore the default value, use the **no** form of this command.

arp timeout *seconds*
no arp timeout *seconds*

Syntax Description

seconds	Time, in seconds, that an entry remains in the ARP cache. A value of zero means that entries are never cleared from the cache.

Default

14400 seconds (4 hours)

Command Mode

Interface configuration

Usage Guidelines

This command is ignored when issued on interfaces that do not use ARP. The **show interfaces** EXEC command displays the ARP timeout value. The value follows the "Entry Timeout:" heading, as seen in this sample **show interfaces** display:

```
ARP type: ARPA, PROBE, Entry Timeout: 14400 sec
```

Example

The following example illustrates how to set the ARP timeout to 12000 seconds to allow entries to time out more quickly than the default:

```
interface ethernet 0
arp timeout 12000
```

Related Command

show interfaces

clear arp-cache

To delete all dynamic entries from the ARP cache, to clear the fast-switching cache, and to clear the IP route cache, use the **clear arp-cache** EXEC command.

clear arp-cache

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Example

The following example removes all dynamic entries from the ARP cache and clears the fast-switching cache:

```
clear arp-cache
```

Related Commands

arp (global)
arp (interface)

clear host

To delete entries from the host-name-and-address cache, use the **clear host** EXEC command.

clear host {*name* | ***}

Syntax Description

name	Particular host entry to remove.
*****	Removes all entries.

Command Mode

EXEC

Usage Guidelines

The host name entries will not be removed from NVRAM, but will be cleared in running memory.

Example

The following example clears all entries from the host name-and-address cache:

```
clear host *
```

Related Commands

show hosts
ip host

clear ip accounting

To clear the active or checkpointed database when IP accounting is enabled, use the **clear ip accounting** EXEC command.

clear ip accounting [**checkpoint**]

Syntax Description

checkpoint	(Optional) Clears the checkpointed database.

Command Mode

EXEC

Usage Guidelines

You can also clear the checkpointed database by issuing the **clear ip accounting** command twice in succession.

Example

The following example clears the active database when IP accounting is enabled:

```
clear ip accounting
```

Related Commands

ip accounting
ip accounting-list
ip accounting-threshold
ip accounting-transits
show ip accounting

clear ip nhrp

To clear all dynamic entries from the Next Hop Resolution Protocol (NHRP) cache, use the **clear ip nhrp** EXEC command.

clear ip nhrp

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

This command does not clear any static (configured) IP-to-NBMA address mappings from the NHRP cache.

Example

In the following example, all dynamic entries are cleared from the NHRP cache for the interface:

```
clear ip nhrp
```

Related Command

show ip nhrp

clear ip route

To delete routes from the IP routing table, use the **clear ip route** EXEC command.

clear ip route {*network* [*mask*] | ***}

Syntax Description

network	Network or subnet address to remove.
mask	(Optional) Subnet address to remove.
*	Removes all routing table entries.

Default

All entries are removed.

Command Mode

EXEC

Example

The following example removes a route to network 132.5.0.0 from the IP routing table:

```
clear ip route 132.5.0.0
```

clear ip sse

To have the route processor recompute the SSE program for IP on the Cisco 7000 series, use the **clear ip sse**l EXEC command.

clear ip sse

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Privileged EXEC

Usage Guidelines

The silicon switching engine (SSE) is on the Silicon Switch Processor (SSP) board in the Cisco 7000.

This command also updates the SSE cache for IP.

Example

The following example causes the route processor to recompute the program for IP:

```
clear ip sse
```

clear sse

To reinitialize the route processor on the Cisco 7000 series, use the **clear sse** EXEC command.

clear sse

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

EXEC

Usage Guidelines

The silicon switching engine (SSE) is on the Silicon Switch Processor (SSP) board in the Cisco 7000.

Example

The following example causes the route processor to be reinitialized:

```
clear sse
```

dnsix-dmdp retries

To set the retransmit count used by the DNSIX Message Delivery Protocol (DMDP), use the **dnsix-dmdp retries** global configuration command. To restore the default numbe of retries, use the **no** form of this command.

dnsix-dmdp retries *count*
no dnsix-dmdp retries *count*

Syntax Description

count	Number of times DMDP will retransmit a message. It can be a decimal integer from 0 through 200. The default is 4 retries, or until acknowledged.

Default

Retransmits messages up to 4 times, or until acknowledged

Command Mode

Global configuration

Example

The following example sets the number of times DMDP will attempt to retransmit a message to 150:

```
dnsix-dmdp retries 150
```

Related Commands

dnsix-nat authorized-redirection
dnsix-nat primary
dnsix-nat secondary
dnsix-nat source
dnsix-nat transmit-count

dnsix-nat authorized-redirection

To specify the address of a collection center that is authorized to change the primary and secondary addresses of the host to receive audit messages, use the **dnsix-nat authorized-redirection** global configuration command. To delete an address, use the **no** form of this command.

dnsix-nat authorized-redirection *ip-address*
no dnsix-nat authorized-redirection *ip-address*

Syntax Description

ip-address	IP address of the host from which redirection requests are permitted.

Default

An empty list of addresses

Command Mode

Global configuration

Usage Guidelines

Use multiple **dnsix-nat authorized-redirection** commands to specify a set of hosts that are authorized to change the destination for audit messages. Redirection requests are checked against the configured list, and if the address is not authorized the request is rejected and an audit message is generated. If no address is specified, no redirection messages are accepted.

Example

The following example specifies that the address of the collection center that is authorized to change the primary and secondary addresses is 193.1.1.1.

```
dnsix-nat authorization-redirection 193.1.1.1.
```

dnsix-nat primary

To specify the IP address of the host to which DNSIX audit messages are sent, use the **dnsix-nat primary** global configuration command. To delete an entry, use the **no** form of this command.

dnsix-nat primary *ip-address*
no dnsix-nat primary *ip-address*

Syntax Description

ip-address	IP address for the primary collection center.

Default

Messages are not sent.

Command Mode

Global configuration

Usage Guidelines

An IP address must be configured before audit messages can be sent.

Example

The following example configures an IP address as the address of the host to which DNSIX audit messages are sent:

```
dnsix-nat primary 194.1.1.1
```

dnsix-nat secondary

To specify an alternate IP address for the host to which DNSIX audit messages are sent, use the **dnsix-nat secondary** global configuration command. To delete an entry, use the **no** form of this command.

dnsix-nat secondary *ip-address*
no dnsix-nat secondary *ip-address*

Syntax Description

ip-address	IP address for the secondary collection center.

Default

No alternate IP address is known.

Command Mode

Global configuration

Usage Guidelines

When the primary collection center is unreachable, audit messages are sent to the secondary collection center instead.

Example

The following example configures an IP address as the address of an alternate host to which DNSIX audit messages are sent:

```
dnsix-nat secondary 193.1.1.1
```

dnsix-nat source

To start the audit-writing module and to define audit trail source address, use the **dnsix-nat source** global configuration command. To disable the DNSIX audit trail writing module, use the **no** form of this command.

dnsix-nat source *ip-address*
no dnsix-nat source *ip-address*

Syntax Description

ip-address	Source IP address for DNSIX audit messages.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

You must issue the **dnsix-nat source** command before any of the other **dnsix-nat** commands. The configured IP address is used as the source IP address for DMDP protocol packets sent to any of the collection centers.

Example

The following example enables the audit trail writing module, and specifies that the source IP address for any generated audit messages should be the same as the primary IP address of interface Ethernet 0.

```
dnsix-nat source 128.105.2.5
interface ethernet 0
ip address 128.105.2.5 255.255.255.0
```

dnsix-nat transmit-count

To have the audit writing module collect multiple audit messages in the buffer before sending the messages to a collection center, use the **dnsix-nat transmit-count** global configuration command. To revert to the default audit message count, use the **no** form of this command.

dnsix-nat transmit-count *count*
no dnsix-nat transmit-count *count*

Syntax Description

count	Number of audit messages to buffer before transmitting to the server. Integer from 1 through 200.

Default

One message is sent at a time.

Command Mode

Global configuration

Usage Guidelines

An audit message is sent as soon as the message is generated by the IP packet-processing code. The audit writing module can, instead, buffer up to several audit messages before transmitting to a collection center.

Example

The following example configures the system to buffer five audit messages before transmitting them to a collection center:

```
dnsix-nat transmit-count 5
```

ip access-group

To control access to an interface, use the **ip access-group** interface configuration command. To remove the specified access group, use the **no** form of this command.

ip access-group *access-list-number* {**in** | **out**}
no ip access-group *access-list-number* {**in** | **out**}

Syntax Description

access-list-number	Number of an access lists. This is a decimal number from 1 through 199.
in	Filters on inbound packets.
out	Filters on outbound packets.

Default

Entering a keyword is strongly recommended, but if a keyword is not specified, **out** is the default.

Command Mode

Interface configuration

Usage Guidelines

For inbound access lists, after receiving a packet, the router checks the source address of the packet against the access list. If the access list permits the address, the router continues to process the packet. If the access list rejects the address, the router discards the packet and returns an ICMP *Host Unreachable* message.

For outbound access lists, after receiving and routing a packet to a controlled interface, the router checks the source address of the packet against the access list. If the access list permits the address, the router transmits the packet. If the access list rejects the address, the router discards the packet and returns an ICMP Host Unreachable message.

Access lists are applied on either outbound or inbound interfaces.

If the specified access list does not exist, all packets are passed.

When you enable outbound access lists, you automatically disable autonomous switching for that interface.When you enable input access lists on any cBus or CxBus interface, you automatically disable autonomous switching for all interfaces (with one exception; an SSE configured with simple access lists can still switch packets, on output only).

Example

The following example applies list 101 on packets outbound from Ethernet 0:

```
interface ethernet 0
ip access-group 101 out
```

Related Commands

access-list (extended)
show access-lists

ip accounting

To enable IP accounting on an interface, use the **ip accounting** interface configuration command. To disable IP accounting, use the **no** form of this command.

ip accounting [**access-violations**]
no ip accounting [**access-violations**]

Syntax Description

access-violations	(Optional) Enables IP accounting with the ability to identify IP traffic that fails IP access lists.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

IP accounting records the number of bytes (IP header and data) and packets switched through the system on a source and destination IP address basis. Only transit IP traffic is measured and only on an outbound basis; traffic generated by the router or terminating in the router is not included in the accounting statistics.

If you specify the **access-violations** keyword, this command provides information identifying IP traffic that fails IP access lists. Identifying IP source addresses that violate IP access lists alerts you to possible attempts to breach security. The data might also indicate that you should verify IP access list configurations.

Statistics are accurate even if IP fast switching or IP access lists are being used on the interface.

IP accounting disables autonomous switching and SSE switching on the interface.

Example

The following example enables IP accounting on Ethernet interface 0:

```
interface ethernet 0
ip accounting
```

Related Commands

clear ip accounting
ip accounting-list
ip accounting-threshold
ip accounting-transits
show ip accounting

ip accounting-list

To define filters to control the hosts for which IP accounting information is kept, use the **ip accounting-list** global configuration command. To remove a filter definition, use the **no** form of this command.

ip accounting-list *ip-address mask*
no ip accounting-list *ip-address mask*

Syntax Description

ip-address	IP address in dotted-decimal format.
mask	IP mask.

Default

No filters are defined.

Command Mode

Global configuration

Usage Guidelines

The source and destination address of each IP datagram is logically ANDed with ones-complement of the *mask* and compared with the *ip-address*. If there is a match, the information about the IP datagram will be entered into the accounting database. If there is no match, the IP datagram is considered a *transit* datagram and will be counted according to the setting of the **ip accounting-transits** global configuration command.

Example

The following example adds all hosts with IP addresses beginning with 192.31 to the list of hosts for which accounting information will be kept:

```
ip accounting-list 192.31.0.0 255.255.0.0
```

Related Commands

clear ip accounting
ip accounting
ip accounting-threshold
ip accounting-transits
show ip accounting

ip accounting-threshold

To set the maximum number of accounting entries to be created, use the **ip accounting-threshold** global configuration command. To restore the default number of entries, use the **no** form of this command.

ip accounting-threshold *threshold*
no ip accounting-threshold *threshold*

Syntax Description

threshold	Maximum number of entries (source and destination address pairs) that the router accumulates.

Default

512 entries

Command Mode

Global configuration

Usage Guidelines

The accounting threshold defines the maximum number of entries (source and destination address pairs) that the router accumulates, preventing IP accounting from possibly consuming all available free memory. This level of memory consumption could occur in a router that is switching traffic for many hosts. Overflows will be recorded; see the monitoring commands for display formats.

The default accounting threshold of 512 entries results in a maximumn table size of 12928 bytes. Active and checkpointed tables can reach this size independently.

Example

The following example sets the IP accounting threshold to only 500 entries:

```
ip accounting-threshold 500
```

Related Commands

clear ip accounting
ip accounting
ip accounting-list
ip accounting-transits
show ip accounting

ip accounting-transits

To control the number of transit records that are stored in the IP accounting database, use the **ip accounting-transits** global configuration command. To return to the default number of records, use the **no** form of this command.

ip accounting-transits *count*
no ip accounting-transits

Syntax Description

count	Number of transit records to store in the IP accounting database.

Default

0

Command Mode

Global configuration

Usage Guidelines

Transit entries are those that do not match any of the filters specified by **ip accounting-list** global configuration commands. If no filters are defined, no transit entries are possible.

To maintain accurate accounting totals, the router software maintains two accounting databases: an active and a checkpointed database.

Example

The following example specifies that no more than 100 transit records are stored:

```
ip accounting-transit 100
```

Related Commands

clear ip accounting
ip accounting
ip accounting-list
ip accounting-threshold
show ip accounting

ip address

To set an IP address for an interface, use the **ip address** interface configuration command. To remove an IP address, use the **no** form of this command.

ip address *ip-address mask*
no ip address *ip-address mask*

Syntax Description

ip-address	IP address.
mask	Mask for the associated IP subnet.

Default

No IP address is defined for an interface.

Command Mode

Interface configuration

Usage Guidelines

Hosts can determine subnet masks using the Internet Control Message Protocol (ICMP) Mask Request message. Routers respond to this request with an ICMP Mask Reply message.

You can disable IP processing on a particular interface by removing its IP address with the **no ip address** command. If the router detects another host using one of its IP addresses, it will print an error message on the console.

Example

In the following example, 131.108.1.27 is the primary address for Ethernet 0:

```
interface ethernet 0
ip address 131.108.1.27 255.255.255.0
```

ip address secondary

To set multiple IP addresses for an interface, use the **ip address secondary** interface configuration command. To remove an address, use the **no** form of this command.

ip address *ip-address mask* **secondary**
no ip address *ip-address mask* **secondary**

Syntax Description

ip-address	IP address.
mask	Mask for the associated IP subnet.

Default

No secondary IP addresses are defined.

Command Mode

Interface configuration

Usage Guidelines

Hosts can determine subnet masks using the Internet Control Message Protocol (ICMP) Mask Request message. Routers respond to this request with an ICMP Mask Reply message.

Packets generated by the router always use the primary interface IP address. Therefore, all routers on a segment should share the same primary network number.

Note When you are routing OSPF, ensure that all secondary addresses of an interface fall into the same OSPF area as the primary addresses.

Example

In the following example, 131.108.1.27 is the primary address and 192.31.7.17 and 192.31.8.17 are secondary addresses for Ethernet 0:

```
interface ethernet 0
ip address 131.108.1.27 255.255.255.0
ip address 192.31.7.17 255.255.255.0 secondary
ip address 192.31.8.17 255.255.255.0 secondary
```

ip broadcast-address

To define a broadcast address for an interface, use the **ip broadcast-address** interface configuration command. To restored the default IP broadcast address, use the **no** form of this command.

ip broadcast-address [*ip-address*]
no ip broadcast-address [*ip-address*]

Syntax Description

ip-address	(Optional) IP broadcast address for a network.

Default

Default address: 255.255.255.255 (all ones)

Command Mode

Interface configuration

Example

The following example specifies an IP broadcast address of 0.0.0.0:

```
ip broadcast-address 0.0.0.0
```

ip cache-invalidate-delay

To control the invalidation rate of the IP route cache, use the **ip cache-invalidate-delay** global configuration command. To allow the IP route cache to be immediately invalidated, use the **no** form of this command.

ip cache-invalidate-delay [*minimum maximum quiet threshold*]
no ip cache-invalidate-delay

Syntax Description

minimum	(Optional) Minimum time, in seconds, between invalidation request and actual invalidation. The default is 2 seconds.
maximum	(Optional) Maximum time, in seconds, between invalidation request and actual invalidation. The default is 5 seconds.
quiet	(Optional) Length of quiet period, in seconds, before invalidation.
threshold	(Optional) Maximum number of invalidation requests considered to be quiet.

Default

minimum = 2 seconds
maximum = 5 seconds, and 3 seconds with no more than zero invalidation requests

Command Mode

Global configuration

Usage Guidelines

All cache invalidation requests are honored immediately.

This command should typically not be used except under the guidance of technical support personnel. Incorrect settings can seriously degrade network performance.

The IP fast switching and autonomous switching features maintain a cache of IP routes for rapid access. When a packet is to be forwarded and the corresponding route is not present in the cache, the packet is process-switched and a new cache entry is built. However, when routing table changes occur (such as when a link or an interface goes down), the route cache must be flushed so that it can be rebuilt with up-to-date routing information.

This command controls how the route cache is flushed. The intent is to delay invalidation of the cache until after routing has settled down, since there tend to be many route table changes clustered in a short period of time, and the cache may be flushed repeatedly, which may put a high CPU load on the router.

When this feature is enabled, and the system requests that the route cache be flushed, the request is held for at least *minimum* seconds. Then the system determines whether the cache has been "quiet," that is, less than *threshold* invalidation requests in the last *quiet* seconds. If the cache has been quiet, the cache is then flushed. If the cache does not become quiet within *maximum* seconds after the first request, it is flushed unconditionally.

Manipulation of these parameters trades off CPU utilization versus route convergence time. Note that this does not affect the timing of the routing protocols, but only of the removal of stale cache entries.

Example

The following example sets a minimum delay of 5 seconds, a maximum delay of 30 seconds, and a quiet threshold of no more than 5 invalidation requests in the previous 10 seconds:

```
ip cache-invalidate-delay 5 30 10 5
```

Related Commands

ip route-cache
show ip cache

ip classless

At times the router might receive packets destined for a subnet of a network that has no network default route. To have the router forward such packets to the best supernet route possible, use the **ip classless** global configuration command. To disable this feature, use the **no** form of this command.

ip classless
no ip classless

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

This command allows the router to forward packets that are destined for unrecognized subnets of directly connected networks. By default, when a router receives packets for a subnet that numerically falls within its subnetwork addressing scheme, if there is no such subnet number in the routing table and there is no network default route, the router discards the packets. However, when the **ip classless** command is enabled, the router instead forwards those packets to the best supernet route.

Example

The following example configures the router to forward packets destined for an unrecognized subnet to the best supernet possible:

```
ip classless
```

ip default-gateway

To define a default gateway (router) when IP routing is disabled, use the **ip default-gateway** global configuration command. To disable this function, use the **no** form of this command.

ip default-gateway *ip-address*
no ip default-gateway *ip-address*

Syntax Description

ip-address	IP address of the router.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

The router sends any packets that need the assistance of a gateway to the address you specify. If another gateway has a better route to the requested host, the default gateway sends an ICMP redirect message to the router. The ICMP redirect message indicates which local router the router should use.

Example

The following example defines the router on IP address 192.31.7.18 as the default router:

```
ip default-gateway 192.31.7.18
```

Related Command

show ip redirects

ip directed-broadcast

To enable the translation of directed broadcast to physical broadcasts, use the **ip directed-broadcast** interface configuration command. To disable this function, use the **no** form of this command.

ip directed-broadcast [*access-list-number*]
no ip directed-broadcast [*access-list-number*]

Syntax Description

access-list-number	(Optional) Number of the access list. If specified, a broadcast must pass the access list to be forwarded. If not specified, all broadcasts are forwarded.

Default

Enabled, with no list specified

Command Mode

Interface configuration

Usage Guidelines

This feature is enabled only for those protocols configured using the **ip forward-protocol** global configuration command. An access list may be specified to control which broadcasts are forwarded. When an access list is specified, only those IP packets permitted by the access list are eligible to be translated from directed broadcasts to physical broadcasts.

Example

The following example enables forwarding of IP directed broadcasts on interface Ethernet 0:

```
interface ethernet 0
ip directed-broadcast
```

Related Command

ip forward-protocol

ip domain-list

To define a list of default domain names to complete unqualified host names, use the **ip domain-list** global configuration command. To delete a name from a list, use the **no** form of this command.

ip domain-list *name*
no ip domain-list *name*

Syntax Description

name	Domain name. Do not include the initial period that separates an unqualified name from the domain name.

Default

No domain names are defined.

Command Mode

Global configuration

Usage Guidelines

If there is no domain list, the domain name that you specified with the **ip domain-name** global configuration command is used. If there is a domain list, the default domain name is not used. The **ip domain-list** command is similar to the **ip domain-name** command, except that with **ip domain-list** you can define a list of domains, each to be tried in turn.

Examples

The following example adds several domain names to a list:

```
ip domain-list martinez.com
ip domain-list stanford.edu
```

The following example adds a name to and then deletes a name from the list:

```
ip domain-list sunya.edu
no ip domain-list stanford.edu
```

Related Command

ip domain-name

ip domain-lookup

To enable the IP Domain Name System-based host name-to-address translation, use the **ip domain-lookup** global configuration command. To disable the Domain Name System, use the **no** form of this command.

ip domain-lookup
no ip domain-lookup

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Global configuration

Example

The following example enables the IP Domain Name System-based host name-to-address translation:

```
ip domain-lookup
```

Related Commands

ip domain-lookup nsap
ip domain-name
ip name-server

ip domain-lookup nsap

To allow Domain Name System (DNS) queries for CLNS addresses, use the **ip domain-lookup nsap** global configuration command. To disable this feature, use the **no** form of this command.

ip domain-lookup nsap
no ip domain-lookup nsap

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Global configuration

Usage Guidelines

With both IP and ISO CLNS enabled on a router, this feature allows the router to dynamically determine a CLNS address given a host name. This feature is useful for the ISO CLNS **ping** EXEC command and when making CLNS Telnet connections.

Example

The following example disables DNS queries of CLNS addresses:

no ip domain-lookup nsap

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

ip domain-lookup
ping (for **ISO CLNS)** †

ip domain-name

To define a default domain name that the router uses to complete unqualified host names (names without a dotted-decimal domain name), use the **ip domain-name** global configuration command. To disable use of the Domain Name System, use the **no** form of this command.

ip domain-name *name*
no ip domain-name

Syntax Description

name	Default domain name used to complete unqualified host names. Do not include the initial period that separates an unqualified name from the domain name.

Default

Enabled

Command Mode

Global configuration

Usage Guidelines

Any IP host name that does not contain a domain name (that is, any name without a dot), will have the dot and cisco.com appended to it before being added to the host table.

Example

The following example defines cisco.com as the default domain name:

```
ip domain-name cisco.com
```

Related Commands

ip domain-list
ip domain-lookup
ip name-server

ip forward-protocol

To specify which protocols and ports the router forwards when forwarding broadcast packets, use the **ip forward-protocol** global configuration command. To remove a protocol or port, use the **no** form of this command.

ip forward-protocol {**udp** [*port*] | **nd** | **sdns**}
no ip forward-protocol {**udp** [*port*] | **nd** | **sdns**}

Syntax Description

udp	Forward User Datagram Protocol (UDP) datagrams. See the "Default" section below for a list of port numbers forwarded by default.
port	(Optional) Destination port that controls which UDP services are forwarded.
nd	Forward Network Disk (ND) datagrams. This protocol is used by older diskless SUN workstations.
sdns	Secure Data Network Service.

Default

If an IP helper address is defined, UDP forwarding is enabled on default ports. If UDP flooding is configured, UDP flooding is enabled on the default ports.

If a helper address is specified and UDP forwarding is enabled, broadcast packets destined to the following port numbers are forwarded by default:

- Trivial File Transfer (TFTP) (port 69)
- Domain Name System (port 53)
- Time service (port 37)
- NetBIOS Name Server (port 137)
- NetBIOS Datagram Server (port 138)
- Boot Protocol (BOOTP) client and server datagrams (ports 67 and 68)
- TACACS service (port 49)

Note Using the **ip directed-broadcast** interface configuration command with the optional *access-list-number* argument overrides the behavior of teh **ip forward-protocol** command.

Command Mode

Global configuration

Usage Guidelines

Enabling a helper address or UDP flooding on an interface causes the router to forward particular broadcast packets. You can use the **ip forward-protocol** command to specify exactly which types of broadcast packets you would like to have forwarded. A number of commonly forwarded applications are enabled by default. Enabling forwarding for some ports (for example, RIP) may be hazardous to your network.

If you use the **ip forward-protocol** command, specifying just UDP, without the port, enables forwarding and flooding on the default ports.

One common application that requires helper addresses is Dynamic Host Configuration Protocol (DHCP). DHCP is defined in RFC 1531. DHCP protocol information is carried inside of BOOTP packets. To enable BOOTP broadcast forwarding for a set of clients, configure a helper address on the router interface closest to the client. The helper address should specify the address of the DHCP server. If you have multiple servers, you can configure one helper address for each server. Since BOOTP packets are forwarded by default, DHCP information can now be forwarded by the router. The DHCP server now receives broadcasts from the DHCP clients.

Example

The following example uses the **ip forward-protocol** command to specify forwarding of UDP port 3001 in addition to the default ports, and then defines a helper address:

```
ip forward-protocol udp 3001
!
interface ethernet 1
ip helper-address 131.120.1.0
```

Related Commands

ip directed-broadcast
ip forward-protocol spanning-tree
ip forward-protocol turbo-flood
ip helper-address

ip forward-protocol any-local-broadcast

To forward any broadcasts including local subnet broadcasts, use the **ip forward-protocol any-local-broadcast** global configuration command. To disable this type of forwarding, use the **no** form of this command.

ip forward-protocol any-local-broadcast
no ip forward-protocol any-local-broadcast

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

The **ip forward-protocol any-local-broadcast** command forwards packets similarly to how the**ip forward-protocol spanning-tree** command does. That is, it forwards packets whose contents are all ones (255.255.255.255), all zeros (0.0.0.0), and, if subnetting is enabled, all networks (131.108.255.255 as an example in the network number 131.108.0.0. This mechanism also forwards packets whose contents are the zeros version of the all-networks broadcast when subnetting is enabled (for example, 131.108.0.0). In addition, it forwards any local subnet broadcast packets.

Example

Assume a router is directly connected to subnet 1 of network 131.108.0.0 and that the netmask is 255.255.255.0. The following command enables the forwarding of IP broadcasts destined to 131.108.1.255 and 131.108.1.0 in addition to the broadcast addresses mentioned in the “Usage Guidelines” section:

```
ip forward-protocol any-local-broadcast
```

Related Command

ip forward-protocol spanning-tree

ip forward-protocol spanning-tree

To permit IP broadcasts to be flooded throughout the internetwork in a controlled fashion, use the **ip forward-protocol spanning-tree** global configuration command. To disable the flooding of IP broadcasts, use the **no** form of this command.

ip forward-protocol spanning-tree
no ip forward-protocol spanning-tree

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

Packets must meet the following criteria to be considered for flooding:

- The packet must be a MAC-level broadcast.
- The packet must be an IP-level broadcast; that is , an all-network broadcast (255.255.255.255) or major network broadcast (131.108.255.255, for example).
- The packet must be a TFTP, DNS, Time, NetBIOS, ND, or BootP packet or a UDP protocol specified by the **ip forward-protocol udp** global configuration command.
- The packet's time-to-live (TTL) value must be at least two.

A flooded UDP datagram is given the destination address specified by the **ip broadcast-address** interface configuration command on the output interface. The destination address can be set to any desired address. Thus, the destination address may change as the datagram propagates through the network. The source address is never changed. The TTL value is decremented.

After a decision has been made to send the datagram out on an interface (and the destination address possibly changed), the datagram is handed to the normal IP output routines and is therefore subject to access lists, if they are present on the output interface.

The **ip forward-protocol spanning-tree** command uses the database created by the bridging spanning-tree protocol. Therefore, the transparent bridging option must be in the routing software, and bridging must be configured on each interface that is to participate in the flooding in order to support this capability.

If an interface does not have bridging configured, it still will be able to receive broadcasts, but it will never forward broadcasts received on that interface, and it will never use that interface to send broadcasts received on a different interface.

If no actual bridging is desired, you can configure a type-code bridging filter that will deny all packet types from being bridged. Refer to the Transparent Bridging chapter in the *Router Products Configuration Guide* for more information about using access lists to filter bridged traffic. The spanning-tree database is still available to the IP forwarding code to use for the flooding.

The spanning-tree-based flooding mechanism fowards packets whose contents are all ones (255.255.255.255), all zeros (0.0.0.0), and, if subnetting is enabled, all networks (131.108.255.255 as an example in the network number 131.108.0.0. This mechanism also forward packets whose contents are the zeros version of the all-networks braodcast when subnetting is enabled (for example, 131.108.0.0).

This command is an extension of the **ip helper-address** interface configuration command, in that the same packets that may be subject to the helper address and forwarded to a single network can now be flooded. Only one copy of the packet will be put on each network segment.

Example

The following example permits IP broadcasts to be flooded through the internetwork in a controlled fashion:

```
ip forward-protocol spanning-tree
```

Related Commands

ip broadcast-address
ip helper-address
ip forward-protocol
ip forward-protocol turbo-flood

ip forward-protocol turbo-flood

To speed up flooding of User Datagram Protocol (UDP) datagrams using the spanning-tree algorithm, use the **ip forward-protocol turbo-flood** global configuration command. To disable this feature, use the **no** form of this command.

ip forward-protocol turbo-flood
no ip forward-protocol turbo-flood

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

Used in conjunction with the **ip forward-protocol spanning-tree** global configuration command, this feature is supported over ARPA-encapsulated Ethernets, FDDI, and HDLC-encapsulated serials, but is not supported on Token Rings. As long as the Token Rings and the non-HDLC serials are not part of the bridge group being used for UDP flooding, turbo flooding will behave normally.

Example

The following is an example of a two-port router (2E) using this feature:

```
ip forward-protocol turbo-flood
ip forward-protocol spanning-tree
!
interface ethernet 0
ip address 128.9.1.1
bridge-group 1
!
interface ethernet 1
ip address 128.9.1.2
bridge-group 1
!
!
bridge 1 protocol dec
```

Related Commands

ip forward-protocol
ip forward-protocol spanning-tree

ip gdp gdp

To configure the router discovery feature using the Cisco Gateway Discovery Protocol (GDP) routing protocol, use the **ip gdp gdp** interface configuration command. To disable this feature, use the **no** form of this command.

ip gdp gdp
no ip gdp gdp

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

IP routing must be disabled before you can configure this feature.

Example

The following example configures router discovery using GDP on Ethernet interface 0:

```
interface ethernet 0
ip gdp gdp
```

ip gdp igrp

To configure the router discovery feature using the Cisco Interior Gateway Routing Protocol (IGRP), use the **ip gdp igrp** interface configuration command. To disable this feature, use the **no** form of this command.

ip gdp igrp
no ip gdp igrp

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

IP routing must be disabled before you can configure this feature.

Example

The following example configures router discovery using IGRP on Ethernet interface 1:

```
interface ethernet 1
ip gdp igrp
```

ip gdp irdp

To configure the router discovery feature using the ICMP Router Discovery Protocol (IRDP), use the **ip gdp irdp** interface configuration command. To disable this feature, use the **no** form of this command.

ip gdp irdp
no ip gdp irdp

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

IP routing must be disabled before you can configure this feature.

Example

The following example configures router discovery using IRDP on Ethernet interface 0:

```
interface ethernet 0
ip gdp irdp
```

ip gdp rip

To configure the router discovery feature using the Routing Information Protocol (RIP), use the **ip gdp rip** interface configuration command. To disable this feature, use the **no** form of this command.

ip gdp rip
no ip gdp rip

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

IP routing must be disabled before you can configure this feature.

Example

The following example configures router discovery using RIP on Ethernet interface 1:

```
interface ethernet 1
ip gdp rip
```

ip helper-address

To have the router forward User Datagram Protocol (UDP) broadcasts, including BOOTP, received on an interface, use the **ip helper-address** interface configuration command. To disable the forwarding of broadcast packets to specific addresses, use the **no** form of this command.

ip helper-address *address*
no ip helper-address *address*

Syntax Description

address	Destination broadcast or host address to be used when forwarding UDP broadcasts. You can have more than one helper address per interface.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

Combined with the **ip forward-protocol** global configuration command, the **ip helper-address** command allows you to control which broadcast packets and which protocols are forwarded.

One common application that requires helper addresses is Dynamic Host Configuration Protocol (DHCP). DHCP is defined in RFC 1531. DHCP protocol information is carried inside of BOOTP packets. To enable BOOTP broadcast forwarding for a set of clients, configure a helper address on the router interface closest to the client. The helper address should specify the address of the DHCP server. If you have multiple servers, you can configure one helper address for each server. Since BOOTP packets are forwarded by default, DHCP information can now be forwarded by the router. The DHCP server now receives broadcasts from the DHCP clients.

Example

The following example defines an address that acts as a helper address:

```
interface ethernet 1
ip helper-address 121.24.43.2
```

Related Command

ip forward-protocol

ip host

To define a static host name-to-address mapping in the host cache, use the **ip host** global configuration command. To remove the name-to-address mapping, use the **no** form of this command.

ip host *name* [*tcp-port-number*] *address1* [*address2...address8*]
no ip host *name address*

Syntax Description

name	Name of the host. The first character can be either a letter or a number, but if you use a number, the operations you can perform are limited.
tcp-port-number	(Optional) TCP port number to connect to when using the defined host name in conjunction with an EXEC connect or telnet command. The default is Telnet (port 23).
address1	Associated IP address.
address2...address8	(Optional) Additional associated IP address. You can bind up to eight addresses to a host name.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

The first character can be either a letter or a number, but if you use a number, the operations you can perform (such as ping) are limited.

Example

The following example uses the **ip host** command to define two static mappings:

```
ip host croff 192.31.7.18
ip host bisso-gw 10.2.0.2 192.31.7.33
```

ip hp-host

To enter into the host table the host name of an HP host to be used for HP Probe Proxy service, use the **ip hp-host** global configuration command. To remove a host name, use the **no** form of this command.

ip hp-host *hostname ip-address*
no ip hp-host *hostname ip-address*

Syntax Description

hostname	Name of the host.
ip-address	IP address of the host.

Default

No host names are defined.

Command Mode

Global configuration

Usage Guidelines

To use the HP Proxy service, you must first enter the host name of the HP host into the host table using this command.

Example

The following example specifies an HP host's name and address, and then enables Probe Proxy:

```
ip hp-host BCWjo 131.108.1.27
interface ethernet 0
ip probe proxy
```

Related Command

ip probe proxy

ip mask-reply

To have the router to respond to Internet Control Message Protocol (ICMP) mask requests by sending ICMP Mask Reply messages, use the **ip mask-reply** interface configuration command. To disable this function, use the **no** form of this command.

ip mask-reply
no ip mask-reply

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Example

The following example enables the sending of ICMP Mask Reply messages on interface Ethernet 0:

```
interface ethernet 0
ip address 131.108.1.0 255.255.255.0
ip mask-reply
```

ip mobile arp

To enable local-area mobility, use the **ip mobile arp** interface configuration command. To disable local-area mobility, use the **no** form of this command.

ip mobile arp [**timers** *keepalive hold-time*] [**access-group** *access-list-number*]
no ip mobile arp [**timers** *keepalive hold-time*] [**access-group** *access-list-number*]

Syntax Description

timers	(Optional) Indicates that you are setting local-area mobility timers.
keepalive	(Optional) Frequency, in seconds, at which the router sends unicast ARP messages to a relocated host to verify that the host is present and has not moved. The default keepalive time is 300 seconds (5 minutes).
hold-time	(Optional) Hold time, in seconds. This is the length of time the router considers that a relocated host is present without receiving some type of ARP broadast or unicast from the host. Normally, the hold time should be at least three times greater than the keepalive time. The default hold time is 900 seconds (15 minutes).
access-group	(Optional) Indicates that you are applying an access list. This access list applies only to local-area mobility.
access-list-number	(Optional) Number of a standard IP access list. It is a decimal number from 1 to 99. Only hosts with addresses permitted by this access list are accepted for local-area mobility.

Default

Local-area mobility is disabled.

If you enable local-area mobility:
keepalive: 300 seconds (5 minutes)
hold-time: 900 seconds (15 minutes)

Command Mode

Interface configuration

Usage Guidelines

Local-area mobility is supported on Ethernet, Token Ring, and FDDI interfaces only.

To create larger mobility areas, you must first redistribute the mobile routes into your IGP. The IGP must support host routes. You can use Enhanced IGRP, OSPF, or ISIS; you can also use RIP, but this is not recommended. The mobile area must consist of a contiguous set of subnets.

Using an access list to control the list of possible mobile nodes is strongly encouraged. Without an access list, misconfigured hosts can be taken for mobile nodes and disrupt normal operations.

Example

The following example configures local-area mobility on Ethernet interface 0:

```
bridge 1 protocol ieee
access-list 10 permit 198.92.37.114
interface ethernet 0
ip mobile arp access-group 10
bridge-group 1
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

access-list (standard)
bridge-group †
bridge protocol †
default-metric (BGP, EGP, OSPF, and RIP) †
network †
redistribute †
router eigrp †
router isis †
router ospf †

ip mtu

To set the maximum transmission unit (MTU) size of IP packets sent on an interface, use the **ip mtu** interface configuration command. To restore the default MTU size, use the **no** form of this command.

ip mtu *bytes*
no ip mtu

Syntax Description

bytes	MTU in bytes.

Default

Minimum is 128 bytes; maximum depends on interface medium.

Command Mode

Interface configuration

Usage Guidelines

If an IP packet exceeds the MTU set for the router's interface, the router will fragment it.

All devices on a physical medium must have the same protocol MTU in order to operate.

Note Changing the MTU value (with the **mtu** interface configuration command) can affect the IP MTU value. If the current IP MTU value is the same as the MTU value, and you change the MTU value, the IP MTU value will be modified automatically to match the new MTU. However, the reverse is not true; changing the IP MTU value has no effect on the value for the **mtu** command.

Example

The following example sets the maximum IP packet size for the first serial interface to 300 bytes:

```
interface serial 0
ip mtu 300
```

Related Command

A dagger (†) indicates that the command is documented in another chapter.

mtu †

ip name-server

To specify the address of one or more name servers to use for name and address resolution, use the **ip name-server** global configuration command. To remove the addresses specified, use the **no** form of this command.

ip name-server *server-address1* [[*server-address2*]... *server-address6*]
no ip name-server *server-address1* [[*server-address2*]... *server-address6*]

Syntax Description

server-address1	IP addresses of name server.
server-address2...server-address6	(Optional) IP addresses of additional name servers (a maximum of six name servers).

Default

No name server addresses are specified.

Command Mode

Global configuration

Example

The following example specifies host 131.108.1.111 as the primary name server and host 131.108.1.2 as the secondary server:

```
ip name-server 131.108.1.111 131.108.1.2
```

This command will be reflected in the configuration file as follows:

```
ip name-server 131.108.1.111
ip name-server 131.108.1.2
```

Related Commands

ip domain-lookup
ip domain-name

ip netmask-format

To specify the format in which netmasks are displayed in **show** command output, use the **ip netmask-format** line configuration command. To restore the default display format, use the **no** form of this command.

ip netmask-format {**bitcount** | **decimal** | **hexadecimal**}
no ip netmask-format [**bitcount** | **decimal** | **hexadecimal**]

Syntax Description

bitcount	Addresses are followed by a slash and the total number of bits in the netmask. For example, 131.108.11.0/24 indicates that the netmask is 24 bits.
decimal	Network masks are displayed in dotted decimal notation (for example, 255.255.255.0).
hexadecimal	Network masks are displayed in hexadecimal format, as indicated by the leading 0X (for example, 0XFFFFFF00).

Default

Netmasks are displayed in dotted decimal format.

Command Mode

Line configuration

Usage Guidelines

IP uses a 32-bit mask that indicates which address bits belong to the network and subnetwork fields and which bits belong to the host field. This is called a netmask. By default, **show** commands display an IP address and then its netmask in dotted decimal notation. For example, a subnet would be displayed as 131.108.11.0 255.255.255.0.

However, you can specify that the display of the network mask appear in hexadecimal format or bit count format instead. The hexadecimal format is commonly used on UNIX systems. The above example would be displayed as 131.108.11.0 0XFFFFFF00.

The bitcount format for displaying network masks is to append a slash (/) and the total number of bits in the netmask to the address itself. The above example would be displayed as 131.108.11.0/24.

Example

The following example configures network masks for the specified line to be displayed in bitcount notation in the output of **show** commands:

```
line vty 0 4
ip netmask-format bitcount
```

ip nhrp authentication

To configure the authentication string for an interface using Next Hop Resolution Protocol (NHRP), use the **ip nhrp authentication** interface configuration command. To remove the authentication string, use the **no** form of this command.

ip nhrp authentication *string*
no ip nhrp authentication [*string*]

Syntax Description

string	Authentication string configured for the source and destination stations that controls whether NHRP stations allow intercommunication. The string can be up to 8 characters long.

Default

No authentication string is configured; the router adds no authentication option to NHRP packets it generates.

Command Mode

Interface configuration

Usage Guidelines

All routers configured with NHRP on a fabric (for an interface) must share the same authentication string.

Example

In the following example, the authentication string *specialxx* must be configured in all routers using NHRP on the interface before NHRP communication occurs:

```
ip nhrp authentication specialxx
```

ip nhrp holdtime

To change the number of seconds that NHRP nonbroadcast, multiaccess (NBMA) addresses are advertised as valid in authoritative NHRP responses, use the **ip nhrp holdtime** interface configuration command. To restore the default value, use the **no** form of this command.

ip nhrp holdtime *seconds-positive* [*seconds-negative*]
no ip nhrp holdtime [*seconds-positive* [*seconds-negative*]]

Syntax Description

seconds-positive	Time in seconds that NBMA addresses are advertised as valid in positive authoritative NHRP responses.
seconds-negative	(Optional) Time in seconds that NBMA addresses are advertised as valid in negative authoritative NHRP responses.

Default

7200 seconds (2 hours) for both arguments

Command Mode

Interface configuration

Usage Guidelines

The **ip nhrp holdtime** command affects authoritative responses only. The advertised holding time is the length of time the router tells other routers to keep information that it is providing in authoritative NHRP responses. The cached IP-to-NBMA address mapping entries are discarded after the holding time expires.

The NHRP cache can contain static and dynamic entries. The static entries never expire. Dynamic entries expire regardless of whether they are authoritative or nonauthoritative.

If you want to change the valid time period for negative NHRP responses, you must also include a value for positive NHRP responses, as the arguments are position dependent.

Examples

In the following example, NHRP NBMA addresses are advertised as valid in positive authoritative NHRP responses for one hour:

```
ip nhrp holdtime 3600
```

In the following example, NHRP NBMA addresses are advertised as valid in negative authoritative NHRP responses for one hour and in positive authoritative NHRP responses for two hours:

```
ip nhrp holdtime 7200 3600
```

ip nhrp interest

To control which IP packets can trigger sending a Next Hop Resolution Protocol (NHRP) Request, use the **ip nhrp interest** interface configuration command. To restore the default value, use the **no** form of this command.

ip nhrp interest *access-list-number*
no ip nhrp interest [*access-list-number*]

Syntax Description

access-list-number	Standard or extended IP access list number in the range 1 through 199.

Default

All non-NHRP packets can trigger NHRP requests.

Command Mode

Interface configuration

Usage Guidelines

Use this command with the **access-list** command to control which IP packets trigger NHRP Requests.

Example

In the following example, any TCP traffic can cause NHRP Requests to be sent, but no other IP packets will cause NHRP Requests:

```
ip nhrp interest 101
access-list 101 permit tcp any any
```

Related Commands

access-list (standard)
access-list (extended)

ip nhrp map

To statically configure the IP-to-NBMA address mapping of IP destinations connected to a nonbroadcast, multiaccess (NBMA) network, use the **ip nhrp map** interface configuration command. To remove the static entry from NHRP cache, use the **no** form of this command.

ip nhrp map *ip-address nbma-address*
no ip nhrp map *ip-address nbma-address*

Syntax Description

ip-address	IP address of the destinations reachable through the NBMA network. This address is mapped to the NBMA address.
nbma-address	Nonbroadcast, multiaccess (NBMA) address which is directly reachable through the NBMA network. The address format varies depending on the medium you are using. For example, ATM has an NSAP address, Ethernet has a MAC address, and SMDS has an E.164 address. This address is mapped to the IP address.

Default

No static IP-to-NBMA cache entries exist.

Command Mode

Interface configuration

Usage Guidelines

You will probably have to configure at least one static mapping in order to reach the Next Hop Server. Repeat this command to statically configure multiple IP-to-NBMA address mappings.

Example

In the following example, this station in a multipoint tunnel network is statically configured to be served by two Next Hop Servers 100.0.0.1 and 100.0.1.3. The NBMA address for 100.0.0.1 is statically configured to be 11.0.0.1 and the NBMA address for 100.0.1.3 is 12.2.7.8.

```
interface tunnel 0
ip nhrp nhs 100.0.0.1
ip nhrp nhs 100.0.1.3
ip nhrp map 100.0.0.1 11.0.0.1
ip nhrp map 100.0.1.3 12.2.7.8
```

Related Command

clear ip nhrp

ip nhrp map multicast

To configure NBMA addresses used as destinations for broadcast or multicast packets to be sent over a tunnel network, use the **ip nhrp map multicast** interface configuration command. To remove the destinations, use the **no** form of this command.

ip nhrp map multicast *nbma-address*
no ip nhrp map multicast *nbma-address*

Syntax Description

nbma-address	Nonbroadcast, multiaccess (NBMA) address which is directly reachable through the NBMA network. The address format varies depending on the medium you are using.

Default

No NBMA addresses are configured as destinations for broadcast or multicast packets.

Command Mode

Interface configuration

Usage Guidelines

This command applies to tunnel interfaces only.

This command is useful for supporting broadcasts over a tunnel network when the underlying network does not support IP multicast. If the underlying network does support IP multicast, you should use the **tunnel destination** command to configure a multicast destination for transmission of tunnel broadcasts or multicasts.

When multiple NBMA addresses are configured, the system replicates the broadcast packet for each address.

Example

In the following example, if a packet is sent to 10.255.255.255, it is replicated to destinations 11.0.0.1 and 11.0.0.2. Addresses 11.0.0.1 and 11.0.0.2 are the IP addresses of two other routers that are part of the tunnel network, but those addresses are their addresses in the underlying network, not the tunnel network. They would have tunnel addresses that are in network 10.0.0.0.

```
interface tunnel 0
ip address 10.0.0.3 255.0.0.0
ip nhrp map multicast 11.0.0.1
ip nhrp map multicast 11.0.0.2
```

ip nhrp network-id

To enable the Next Hop Resolution Protocol (NHRP) on an interface, use the **ip nhrp network-id** interface configuration command. To disable NHRP on the interface, use the **no** form of this command.

ip nhrp network-id *number*
no ip nhrp network-id [*number*]

Syntax Description

number	Globally-unique, 32-bit network identifier for a nonbroadcast, multiaccess (NBMA) network. The range is 1 to 4294967295.

Default

NHRP is disabled on the interface.

Command Mode

Interface configuration

Usage Guidelines

In general, all NHRP stations within a fabric must be configured with the same network identifier.

Example

In the following example, NHRP is enabled on the interface:

```
ip nhrp network-id 1
```

ip nhrp nhs

To specify the address of one or more NHRP Next Hop Servers, use the **ip nhrp nhs** interface configuration command. To remove the address, use the **no** form of this command.

ip nhrp nhs *nhs-address* [*net-address* [*netmask*]]
no ip nhrp nhs *nhs-address* [*net-address* [*netmask*]]

Syntax Description

nhs-address	Address of the Next Hop Server being specified.
net-address	(Optional) IP address of a network served by the Next Hop Server.
netmask	(Optional) IP network mask to be associated with the *net* IP address. The *net* IP address is logically ANDed with the mask.

Default

No Next Hop Servers are explicitly configured, so NHRP fabric mode is assumed and normal IP routing decisions are used to forward NHRP traffic.

Command Mode

Interface configuration

Usage Guidelines

Use this command to specify the address of a Next Hop Server and the networks it serves. When Next Hop Servers are configured, server mode is assumed. In server mode, each Next Hop Server should be configured with information as to what networks are served by the other Next Hop Servers in the nonbroadcast, multiaccess (NBMA) network.

For any Next Hop Server that is configured, you can specify multiple networks that it serves by repeating this command with the same *nhs-address* address, but different *net-address* IP network addresses.

If no Next Hop Server is configured for an NBMA network, NHRP fabric mode is assumed.

Example

In the following example, the Next Hop Server with address 131.108.10.11 serves IP network 10.0.0.0. The mask is 255.0.0.0.

```
ip nhrp nhs 131.108.10.11 10.0.0.0 255.0.0.0
```

ip nhrp record

To re-enable the use of forward record and reverse record options in NHRP Request and Reply packets, use the **ip nhrp record** interface configuration command. To suppress the use of such options, use the **no** form of this command.

ip nhrp record
no ip nhrp record

Syntax Description

This command has no arguments or keywords.

Default

Forward record and reverse record options are used in NHRP Request and Reply packets.

Command Mode

Interface configuration

Usage Guidelines

Forward record and reverse record options provide loop detection and are enabled by default. Using the **no** form of this command disables this method of loop detection. For another method of loop detection, see the **ip nhrp responder** command.

Example

In the following example, forward record and reverse record options are suppressed:

```
no ip nhrp record
```

Related Command

ip nhrp responder

ip nhrp responder

To designate which interface's primary IP address the Next Hop Server will use in NHRP Reply packets when the NHRP requestor uses the Responder Address option, use the **ip nhrp responder** interface configuration command. To remove the designation, use the **no** form of this command.

ip nhrp responder *type number*
no ip nhrp responder [*type*] [*number*]

Syntax Description

type	Interface type whose primary IP address is used when a Next Hop Server complies with a Responder Address option (for example, **serial**, **tunnel**).
number	Interface number whose primary IP address is used when a Next Hop Server complies with a Responder Address option.

Default

The Next Hop Server uses the IP address of the interface where the NHRP Request was received.

Command Mode

Interface configuration

Usage Guidelines

If an NHRP requestor wants to know which Next Hop Server generates an NHRP Reply packet, it can request that information through the Responder Address option. The Next Hop Server that generates the NHRP Reply packet then complies by inserting its own IP address in the Responder Address option of the NHRP Reply. The Next Hop Server uses the primary IP address of the specified interface.

If an NHRP Reply packet being forwarded by a Next Hop Server contains that Next Hop Server's own IP address, the Next Hop Server generates an Error Indication of type "NHRP Loop Detected" and discards the Reply.

Example

In the following example, any NHRP requests for the Responder Address will cause this router acting as a Next Hop Server to supply the primary IP address of serial interface 0 in the NHRP Reply packet:

```
ip nhrp responder serial 0
```

ip probe proxy

To enable the HP Probe Proxy support, which allows a router to respond to HP Probe Proxy Name requests, use the **ip probe proxy** interface configuration command. To disable HP Prove Proxy, use the **no** form of this command.

ip probe proxy
no ip probe proxy

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

HP Probe Proxy Name requests are typically used at sites that have HP equipment and are already using HP Probe.

To use the HP Proxy service, you must first enter the host name of the HP host into the host table using the **ip hp-host** global configuration command.

Example

The following example specifies an HP host's name and address, and then enables Probe Proxy:

```
ip hp-host BCWjo 131.108.1.27
interface ethernet 0
ip probe proxy
```

Related Command

ip hp-host

ip proxy-arp

To enable proxy ARP on an interface, use the **ip proxy-arp** interface configuration command. To disable proxy ARP on the interface, use the **no** form of this command.

ip proxy-arp
no ip proxy-arp

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

Example

The following example enables proxy ARP on Ethernet interface 0:

```
interface ethernet 0
ip proxy-arp
```

ip redirects

To enable the sending of redirect messages if the router is forced to resend a packet through the same interface on which it was received, use the **ip redirects** interface configuration command. To disable the sending of redirect messages, use the **no** form of this command.

ip redirects
no ip redirects

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

Example

The following example enables the sending of IP redirects on Ethernet interface 0:

```
interface ethernet 0
ip redirects
```

Related Command

show ip redirects

ip route-cache

To control the use of a high-speed switching cache for IP routing as well as the use of autonomous switching, use the **ip route-cache** interface configuration command. To disable fast switching and autonomous switching, use the **no** form of this command.

ip route-cache [**cbus**]
no ip route-cache [**cbus**]

ip route-cache same-interface
no ip route-cache same-interface

ip route-cache sse
no ip route-cache sse

Syntax Description

cbus	(Optional) Enables both autonomous switching and fast switching.
same-interface	Enables fast switching packets back out the interface on which they arrived.
sse	Enables SSE switching on the SSP board on the Cisco 7000 series.

Default

IP autonomous switching is disabled.
Fast switching varies by interface and media.
SSE switching of IP is disabled.

Command Mode

Interface configuration

Usage Guidelines

Using the route cache is often called *fast switching*. The route cache allows outgoing packets to be load-balanced on a *per-destination* basis.

The **ip route-cache** command with not additional keywords, enables fast switching.

Our routers generally offer better packet transfer performance when fast switching is enabled, with one exception. On networks using slow serial links (64K and below), disabling fast switching to enable the per-packet load sharing is usually the best choice.

Autonomous switching gives a router faster packet processing by allowing the ciscoBus to switch packets independently without interrupting the system processor. It works only in Cisco 7000 series or AGS+ systems with high-speed network controller cards, and with a switch processor or ciscoBus controller card running microcode Version 1.4 or later.

You can enable IP fast switching when the input and output interfaces are the same interface, using the **ip route-cache same-interface** command. This normally is not recommended, though it is useful when you have partially meshed media, such as Frame Relay. You could use this feature on other interfaces, although it is not recommended because it would interfere with redirection.

SSE switching gives a router even faster packet processing than the is provided by the other **ip route-cache** commands by allowing the SSE to switch packets without interrupting the system processor. SSE switching is supported only in Cisco 7000 systems with an SSP board. Fast switching must be active to enable SSE switching. SSE switching requires that fast switching be enabled.

Examples

The following example enables both fast switching and autonomous switching:

```
ip route-cache cbus
```

The following example disables both fast switching and autonomous switching:

```
no ip route-cache
```

The following example turns off autonomous switching only:

```
no ip route-cache cbus
```

The following example returns the system to its defaults (fast switching enabled; autonomous switching disabled):

```
ip route-cache
```

Related Commands

ip cache-invalidate-delay
show ip cache

ip routing

To enable IP routing on the router, use the **ip routing** global configuration command. To disable IP routing on the router, use the **no** form of this command.

ip routing
no ip routing

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Global configuration

Usage Guidelines

If the system is running bridging software, the **no ip routing** command turns off IP routing when setting up a system to bridge (as opposed to route) IP packets.

Example

The following example shows how to enable IP routing:

```
ip routing
```

ip security add

To add a basic security option to all outgoing packets, use the **ip security add** interface configuration command. To disable the adding of a basic security option to all outgoing packets, use the **no** form of this command.

ip security add
no ip security add

Syntax Description

This command has no arguments or keywords.

Default

Disabled, when the security level of the interface is "Unclassified Genser" (or unconfigured). Otherwise, the default is enabled.

Command Mode

Interface configuration

Usage Guidelines

If an outgoing packet does not have a security option present, this interface configuration command will add one as the first IP option. The security label added to the option field is the label that was computed for this packet when it first entered the router. Because this action is performed after all the security tests have been passed, this label will either be the same as or will fall within the range of the interface.

Example

The following example adds a basic security option to each packet leaving Ethernet interface 0:

```
interface ethernet 0
ip security add
```

Related Commands

ip security dedicated
ip security extended-allowed
ip security first
ip security ignore-authorities
ip security implicit-labelling
ip security multilevel
ip security reserved-allowed
ip security strip

ip security aeso

To attach Auxiliary Extended Security Options (AESOs) to an interface, use the **ip security aeso** command. To disable AESO on an interface, use the **no** form of this command.

ip security aeso *source compartment-bits*
no ip security aeso *source compartment-bits*

Syntax Description

source	Extended Security Option (ESO) source. This can be an integer from 0 through 255.
compartment-bits	Compartment bits in hexadecimal.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

Compartment bits are specified only if this AESO is to be inserted in a packet. On every incoming packet at this level on this interface, these AESOs should be present.

Beyond being recognized, no further processing of AESO information is performed. AESO contents are not checked and are assumed to be valid if the source is listed in the configurable AESO table.

Configuring any per-interface extended IP security option (IPSO) information automatically enables **ip security extended-allowed** (disabled by default).

Example

In the following example, the extended security option source is defined as 5 and the compartments bits are set to 5.

```
interface ethernet 0
ip security aeso 5 5
```

Related Commands

ip security eso-info
ip security eso-max
ip security eso-min
ip security extended-allowed

ip security dedicated

To set the level of classification and authority on the interface, use the **ip security dedicated** interface configuration command. To reset the interface to the default classification and authorities, use the **no** form of this command.

ip security dedicated *level authority* [*authority...*]
no ip security dedicated *level authority* [*authority...*]

Syntax Description

level	Degree of sensitivity of information. The level keywords are listed in Table 17-1.
authority	Organization that defines the set of security levels that will be used in a network. The authority keywords are listed in Table 17-2.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

All traffic entering the system on this interface must have a security option that exactly matches this label. Any traffic leaving via this interface will have this label attached to it.

The following definitions apply to the descriptions of the IP security options (IPSO) in this section:

- **level**—The degree of sensitivity of information. For example, data marked TOPSECRET is more sensitive than data marked SECRET. The level keywords and their corresponding bit patterns are shown in Table 17-1.

Table 17-1 IPSO Level Keywords and Bit Patterns

Level Keyword	Bit Pattern
Reserved4	0000 0001
TopSecret	0011 1101
Secret	0101 1010
Confidential	1001 0110
Reserved3	0110 0110
Reserved2	1100 1100
Unclassified	1010 1011
Reserved1	1111 0001

- **authority**—An organization that defines the set of security levels that will be used in a network. For example, the Genser authority consists of level names defined by the U.S. Defense Communications Agency (DCA). The authority keywords and their corresponding bit patterns are shown in Table 17-2.

Table 17-2 IPSO Authority Keywords and Bit Patterns

Authority Keyword	Bit Pattern
Genser	1000 0000
Siop-Esi	0100 0000
DIA	0010 0000
NSA	0001 0000
DOE	0000 1000

- **label**—A combination of a security level and an authority or authorities.

Example

The following example sets a confidential level with Genser authority:

```
ip security dedicated confidential Genser
```

Related Commands

ip security add
ip security extended-allowed
ip security first
ip security ignore-authorities
ip security implicit-labelling
ip security multilevel
ip security reserved-allowed
ip security strip

ip security eso-info

To configure system-wide defaults for extended IP Security Option (IPSO) information, use the **ip security eso-info** global configuration command. To return to the default settings, use the **no** form of this command.

ip security eso-info *source compartment-size default-bit*
no ip security eso-info *source compartment-size default-bit*

Syntax Description

source	Hexadecimal or decimal value representing the extended IPSO source. This is an integer from 0 through 255.
compartment-size	Maximum number of bytes of compartment information allowed for a particular extended IPSO source. This is an integer from 1 through 16.
default-bit	Default bit value for any unsent compartment bits.

Default

Disabled

Command mode

Global configuration

Usage Guidelines

This command configures Extended Security Option (ESO) information, including Auxiliary Extended Security Option (AESO). Transmitted compartment info is padded to the size specified by the *compartment-size* argument.

Example

In the following example, system-wide defaults for source, compartment size, and the default bit value are set:

```
ip security eso-info 100 5 1
```

Related Commands

ip security eso-max
ip security eso-min

ip security eso-max

To specify the maximum sensitivity level for an interface, use the **ip security eso-max** interface configuration command. To return to the default, use the **no** form of this command.

ip security eso-max *source compartment-bits*
no ip security eso-max *source compartment-bits*

Syntax Description

source	Extended Security Option (ESO) source. This is an integer from 1 through 255.
compartment-bits	Compartment bits in hexadecimal.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

This command is used to specify the minimum sensitivity level for a particular interface. Before the per interface compartment information for a particular Network Level Extended Security Option (NLESO) source can be configured, the **ip security eso-info** global configuration command must be used to specify the default information.

On every incoming packet on the interface, these extended security options should be resent at the minimum level and should match the configured compartment bits. Every outgoing packet must have these ESOs.

On every packet transmitted or received on this interface, any NLESO sources present in the IP header should be bounded by the minimum sensitivity level and bounded by the maximum sensitivity level configured for the interface.

When transmitting locally generated traffic out this interface, or adding security information (with the **ip security add** command), the maximum compartment bit information can be used to construct the NLESO sources placed in the IP header.

A maximum of 16 NLESO sources can be configured per interface. Due to IP header length restrictions, a maximum of 9 of these NLESO sources appear in the IP header of a packet.

Example

In the following example, the specified ESO source is 240 and the compartment bits are specified as 500.

```
interface ethernet 0
ip security eso-max 240 500
```

Related Commands

ip security eso-info
ip security eso-min

ip security eso-min

To configure the minimum sensitivity for an interface, use the **ip security eso-min** interface configuration command. To return to the default, use the **no** form of this command.

ip security eso-min *source compartment-bits*
no ip security eso-min *source compartment-bits*

Syntax Description

source	Extended Security Option (ESO) source. This is an integer from 1 through 255.
compartment-bits	Compartment bits in hexadecimal.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

This command is used to specify the minimum sensitivity level for a particular interface. Before the per-interface compartment information for a particular Network Level Extended Security Option (NLESO) source can be configured, the **ip security eso-info** global configuration command must be used to specify the default information.

On every incoming packet on this interface, these extended security options should be resent at the minimum level and should match the configured compartment bits. Every outgoing packet must have these ESOs.

On every packet transmitted or received on this interface, any NLESO sources present in the IP header should be bounded by the minimum sensitivity level and bounded by the maximum sensitivity level configured for the interface.

When transmitting locally generated traffic out this interface, or adding security information (with the **iip security add** command), the maximum compartment bit information can be used to construct the NLESO sources placed in the IP header.

A maximum of 16 NLESO sources can be configured per interface. Due to IP header length restrictions, a maximum of 9 of these NLESO sources appear in the IP header of a packet.

Example

In the following example, the specified ESO source is 5 and the compartment bits are specified as 5.

```
interface ethernet 0
ip security eso-min 5 5
```

Related Commands

ip security eso-info
ip security eso-max

ip security extended-allowed

To accept packets on an interface that has an extended security option present, use the **ip security extended-allowed** interface configuration command. To restore the default, use the **no** form of this command.

ip security extended-allowed
no ip security extended-allowed

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

Packets containing extended security options are rejected.

Example

The following example allows interface Ethernet 0 to accept packets that have an extended security option present:

```
interface ethernet 0
ip security extended-allowed
```

Related Commands

ip security add
ip security dedicated
ip security first
ip security ignore-authorities
ip security implicit-labelling
ip security multilevel
ip security reserved-allowed
ip security strip

ip security first

To prioritize the presence of security options on a packet, use the **ip security first** interface configuration command. To disable this function, use the **no** form of this command.

ip security first
no ip security first

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

If a basic security option is present on an outgoing packet, but it is not the first IP option, then the packet is moved to the front of the options field when this interface configuration command is used.

Example

The following example ensures that, if a basic security option is present in the options field of a packet exiting interface Ethernet 0, the packet is moved to the front of the options field.

```
interface ethernet 0
ip security first
```

Related Commands

ip security add
ip security dedicated
ip security extended-allowed
ip security ignore-authorities
ip security implicit-labelling
ip security multilevel
ip security reserved-allowed
ip security strip

ip security ignore-authorities

To have the router ignore the authorities field of all incoming packets, use the **ip security ignore-authorities** interface configuration command. To disable this function, use the **no** form of this command.

ip security ignore-authorities
no ip security ignore-authorities

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

When the packet's authority field is ignored, the value used in place of this field is the authority value declared for the specified interface. IP security ignore-authorities can only be configured on interfaces with dedicated security levels.

Example

The following example causes interface Ethernet 0 to ignore the authorities field on all incoming packets:

```
interface ethernet 0
ip security ignore-authorities
```

Related Commands

ip security add
ip security dedicated
ip security extended-allowed
ip security first
ip security implicit-labelling
ip security multilevel
ip security reserved-allowed
ip security strip

ip security implicit-labelling

To force the router to accept packets on the interface, even if they do not include a security option, use the **ip security implicit-labelling** interface configuration command. To disable this function, use the **no** form of this command.

ip security implicit-labelling [*level authority* [*authority...*]]
no ip security implicit-labelling [*level authority* [*authority...*]]

Syntax Description

level	(Optional) Degree of sensitivity of information. If your interface has multilevel security set, you must specify this argument. The level keywords are listed in Table 17-1 (see the **ip security dedicated** interface configuration command).
authority	(Optional) Organization that defines the set of security levels that will be used in a network. If your interface has multilevel security set, you must specify this argument. You can specify more than one. The authority keywords are listed in Table 17-2 (see the **ip security dedicated** interface configuration command).

Default

Enabled, when the security level of the interface is “Unclassified Genser” (or unconfigured). Otherwise, the default is disabled.

Command Mode

Interface configuration

Usage Guidelines

If your interface has multilevel security set, you must use the expanded form of the command (with the optional arguments as noted in brackets) because the arguments are used to specify the precise level and authority to use when labeling the packet. If your interface has dedicated security set, the additional arguments are ignored.

Example

In the following example, an interface is set for security and will accept unlabeled packets:

```
ip security dedicated confidential genser
ip security implicit-labelling
```

Related Commands

ip security add
ip security dedicated
ip security extended-allowed
ip security first
ip security ignore-authorities

ip security multilevel
ip security reserved-allowed
ip security strip

ip security multilevel

To set the range of classifications and authorities on an interface, use the **ip security multilevel** interface configuration command. To disable this function, use the **no** form of this command.

ip security multilevel *level1* [*authority1*...] **to** *level2 authority2* [*authority2*...]
no ip security multilevel

Syntax Description

level1	Degree of sensitivity of information. The classification level of incoming packets must be equal to or greater than this value for processing to occur. The level keywords are found in Table 17-1 (see the **ip security dedicated**command).
authority1	(Optional) Organization that defines the set of security levels that will be used in a network. The authority bits must be a superset of this value. The authority keywords are listed in Table 17-2 (see the **ip security dedicated** command).
to	Separates the range of classifications and authorities.
level2	Degree of sensitivity of information. The classification level of incoming packets must be equal to or less than this value for processing to occur. The level keywords are found in Table 17-1 (see the **ip security dedicated** command).
authority2	Organization that defines the set of security levels that will be used in a network. The authority bits must be a proper subset of this value. The authority keywords are listed in Table 17-2 (see the **ip security dedicated** command).

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

All traffic entering or leaving the system must have a security option that falls within this range. Being within range requires that the following two conditions be met:

- The classification level must be greater than or equal to *level1* and less than or equal to *level2*.
- The authority bits must be a superset of *authority1* and a proper subset of *authority2*. That is, *authority1* specifies those authority bits that are required on a packet, while *authority2* specifies the required bits plus any optional authorities that also can be included. If the *authority1* field is the empty set, then a packet is required to specify any one or more of the authority bits in *authority2*.

Example

The following example specifies levels Unclassified to Secret and NSA authority:

```
ip security multilevel unclassified to secret nsa
```

Related Commands

ip security add
ip security dedicated
ip security extended-allowed
ip security first
ip security ignore-authorities
ip security implicit-labelling
ip security reserved-allowed
ip security strip

ip security reserved-allowed

To treat as valid any packets that have Reserved1 through Reserved4 security levels, use the **ip security reserved-allowed** interface configuration command. To disable this feature, use the **no** form of this command.

ip security reserved-allowed
no ip security reserved-allowed

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

When you set multilevel security on an interface, and indicate, for example, that the highest range allowed is Confidential, and the lowest is Unclassified, the router neither allows nor operates on packets that have security levels of Reserved3 and Reserved2 because they are undefined.

If you use the IP Security Option (IPSO) to block transmission out of unclassified interfaces, and you use one of the Reserved security levels, you *must* enable this feature to preserve network security.

Example

The following example allows a security level of Reserved through Ethernet interface 0:

```
interface ethernet 0
ip security reserved-allowed
```

Related Commands

ip security add
ip security dedicated
ip security extended-allowed
ip security first
ip security ignore-authorities
ip security implicit-labelling
ip security multilevel
ip security strip

ip security strip

To remove any basic security option on outgoing packets on an interface, use the **ip security strip** interface configuration command. To disable this function, use the **no** form of this command.

ip security strip
no ip security strip

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

This procedure is performed after all security tests in the router have been passed. This command is not allowed for multilevel interfaces.

Example

The following example removes any basic security options on outgoing packets on interface Ethernet 0:

```
interface ethernet 0
ip security strip
```

Related Commands

ip security add
ip security dedicated
ip security extended-allowed
ip security first
ip security ignore-authorities
ip security implicit-labelling
ip security multilevel
ip security reserved-allowed

ip source-route

To allow the router to handle IP datagrams with source routing header options, use the **ip source-route** global configuration command. To have the router discard any IP datagram containing a source-route option, use the **no** form of this command.

ip source-route
no ip source-route

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Global configuration

Example

The following example enables the handling of IP datagrams with source routing header options:

```
ip source-route
```

Related Commands

ping (privileged)
ping (user)

ip subnet-zero

To enable the use of subnet zero for interface addresses and routing updates, use the **ip subnet-zero** global configuration command. To restore the default, use the **no** form of this command.

ip subnet-zero
no ip subnet-zero

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

The **ip subnet-zero** command provides the ability to configure and route to subnet-zero subnets.

Subnetting with a subnet address of zero is discouraged because of the confusion inherent in having a network and a subnet with indistinguishable addresses.

Example

In the following example, subnet-zero is enabled for the router:

```
ip subnet-zero
```

ip tcp compression-connections

To specify the total number of header compression connections that can exist on an interface, uUse the **ip tcp compression-connections** interface configuration command. To restore the default, use the **no** form of this command.

ip tcp compression-connections *number*
no ip tcp compression-connections *number*

Syntax Description

number	Number of connections the cache supports. It can be a number from 3 through 256.

Default

16 connections

Command Mode

Interface configuration

Usage Guidelines

You should configure one connection for each TCP connection through the specified interface.

Each connection sets up a compression cache entry, so you are in effect specifying the maximum number of cache entries and the size of the cache. Too few cache entries for the specified interface can lead to degraded performance, while too many cache entries can lead to wasted memory.

Note Both ends of the serial connection must use the same number of cache entries.

Example

In the following example, the first serial interface is set for header compression with a maximum of ten cache entries:

```
interface serial 0
ip tcp header-compression
ip tcp compression-connections 10
```

Related Commands

ip tcp header-compression
show ip tcp header-compression

ip tcp header-compression

To enable TCP header compression, use the **ip tcp header-compression** interface configuration command. To disable compression, use the **no** form of this command.

ip tcp header-compression [**passive**]
no ip tcp header-compression [**passive**]

Syntax Description

passive	(Optional) Compresses outgoing TCP packets only if incoming TCP packets on the same interface are compressed. If you do not specify the **passive** keyword, the router compresses all traffic.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

You can compress the headers of your TCP/IP packets in order to reduce the size of your packets. TCP header compression is supported on serial lines using HDLC or PPP encapsulation. You must enable compression on both ends of a serial connection. RFC 1144 specifies the compression process. Compressing the TCP header can speed up Telnet connections dramatically. In general, TCP header compression is advantageous when your traffic consists of many small packets, not for traffic that consists of large packets. Transaction processing (usually using terminals) tends to use small packets while file transfers use large packets. This feature only compresses the TCP header, so it has no effect on UDP packets or other protocol headers.

When compression is enabled, fast switching is disabled. This means that fast interfaces like T1 can overload the router. Consider your network's traffic characteristics before using this command.

Example

In the following example, the first serial interface is set for header compression with a maximum of ten cache entries:

```
interface serial 0
ip tcp header-compression
ip tcp compression-connections 10
```

Related Commands

ip tcp compression-connections
show ip tcp header-compression

ip tcp path-mtu-discovery

To enable Path MTU Discovery for all new TCP connections from the router, use the **ip tcp path-mtu-discovery** interface configuration command. To disable the feature, use the **no** form of this command.

ip tcp path-mtu-discovery
no ip tcp path-mtu-discovery

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

Path MTU Discovery is a method for maximizing the use of available bandwidth in the network between the end points of a TCP connection. It is described in RFC 1191. Existing connections are not affected when this feature is turned on or off.

Customers using TCP connections to move bulk data between systems on distinct subnets would benefit most by enabling this feature. This might include customers using RSRB with TCP encapsulation, STUN, X.25 Remote Switching (also known as XOT or X.25 over TCP), and some protocol translation configurations.

Example

In the following example, Path MTU Discovery is enabled:

```
ip tcp path-mtu-discovery
```

ip tcp synwait-time

To set a period of time the router waits while attempting to establish a TCP connection before it times out, use the **ip tcp synwait-time** global configuration command. To restore the default time, use the **no** form of this command.

ip tcp synwait-time *seconds*
no ip tcp synwait-time *seconds*

Syntax Description

seconds	Time in seconds the router waits while attempting to establish a TCP connection. It can be an integer from 5 to 300 seconds. The default is 30 seconds.

Default

30 seconds

Command Mode

Global configuration

Usage Guidelines

In previous versions of router software, the system would wait a fixed 30 seconds when attempting to establish a TCP connection. If your network contains Public Switched Telephone Network Dial on Demand Routing (PSTN DDR), it is possible that the call setup time will exceed 30 seconds. This amount of time is not sufficient in networks that have dial-up asynchronous connections because it will affect your ability to Telnet over the link (from the router) if the link must be brought up. If you have this type of network, you might want to set this value to the UNIX value of 75.

Because this is a host parameter, it does not pertain to traffic going *through* the router, just for traffic originated *at* the router. Because UNIX has a fixed 75-second timeout, hosts are unlikely to see this problem.

Example

The following example configures the router to continue attempting to establish a TCP connection for 180 seconds:

```
ip tcp synwait-time 180
```

ip unnumbered

To enable IP processing on a serial interface without assigning an explicit IP address to the interface, use the **ip unnumbered** interface configuration command. To disable the IP processing on the interface, use the **no** form of this command.

ip unnumbered *interface-name*
no ip unnumbered *interface-name*

Syntax Description

interface-name	Name of another interface on which the router has an assigned IP address. It cannot be another unnumbered interface.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

Whenever the unnumbered interface generates a packet (for example, for a routing update), it uses the address of the specified interface as the source address of the IP packet. It also uses the address of the specified interface in determining which routing processes are sending updates over the unnumbered interface. Restrictions include the following:

- Serial interfaces using HDLC, PPP, LAPB, and Frame Relay encapsulations, as well as SLIP and tunnel interfaces can be unnumbered. It is not possible to use this interface configuration command with X.25 or SMDS interfaces.
- You cannot use the **ping** EXEC command to determine whether the interface is up, because the interface has no address. Simple Network Management Protocol (SNMP) can be used to remotely monitor interface status.
- You cannot netboot a runnable image over an unnumbered serial interface.
- You cannot support IP security options on an unnumbered interface.

The interface you specify by the *interface-name* argument must be enabled (listed as "up" in the **show interfaces** command display).

If you are configuring IS-IS across a serial line, you should configure the serial interfaces as unnumbered. This allows you to conform with RFC 1195, which states that IP addresses are not required on each interface.

Note Using an unnumbered serial line between different major networks (majornets) requires special care. If at each end of the link there are different majornets assigned to the interfaces you specified as unnumbered, then any routing protocol running across the serial line must not advertise subnet information.

Example

In the following example, the first serial interface is given Ethernet 0's address:

```
interface ethernet 0
ip address 131.108.6.6 255.255.255.0
interface serial 0
ip unnumbered ethernet 0
```

ip unreachables

To enable the generation of ICMP Unreachable messages, use the **ip unreachables** interface configuration command. To disable this function, use the **no** form of this command.

ip unreachables
no ip unreachables

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Interface configuration

Usage Guidelines

If the router receives a nonbroadcast packet destined for itself that uses a protocol it does not recognize, it sends an ICMP *Protocol Unreachable* message to the source.

If the router receives a datagram that it cannot deliver to its ultimate destination because it knows of no route to the destination address, it replies to the originator of that datagram with an ICMP *Host Unreachable* message.

This command affects all kinds of ICMP unreachable messages.

Example

The following example enables the generation of ICMP Unreachable messages, as appropriate, on an interface:

```
interface ethernet 0
ip unreachables
```

ping (user)

To check host reachability and network connectivity, use the **ping** (IP packet internet groper function) user EXEC command.

ping [*protocol*] {*host* | *address*}

Syntax Description

protocol	(Optional) Protocol keyword. The default is IP.
host	Host name of system to ping.
address	IP address of system to ping.

Command Mode

EXEC

Usage Guidelines

The **ping** command sends ICMP *Echo* messages. If the router receives an ICMP *Echo* message, it sends an ICMP *Echo Reply* message to the source of the ICMP *Echo* message.

The user ping feature provides a basic ping facility for IP users who do not have system privileges. This feature allows the router to perform the simple default ping functionality for the IP protocol. Only the nonverbose form of the **ping** command is supported for user pings.

If the system cannot map an address for a host name, it will return an "%Unrecognized host or address" error message.

To abort a ping session, type the escape sequence (by default, Ctrl-^ X, which is done by simultaneously pressing the Ctrl, Shift, and 6 keys, letting go, then pressing the X key).

Table 17-3 describes the test characters that the ping facility sends.

Table 17-3 Ping Test Characters

Char	Description
!	Each exclamation point indicates receipt of a reply.
.	Each period indicates the network server timed out while waiting for a reply.
U	Destination unreachable.
N	Network unreachable.
P	Protocol unreachable.
Q	Source quench.
M	Could not fragment.
?	Unknown packet type.

Sample Display Using an IP Host Name

The following display shows sample ping output when you ping a host named fred:

```
Router> ping fred
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.31.7.27, timeout is 2 seconds:
!!!!!
Success rate is 100 percent, round-trip min/avg/max = 1/3/4 ms
```

Sample Display Using the Broadcast Address

The following display shows sample ping output when you ping the broadcast address of 255.255.255.255:

```
Router> ping 255.255.255.255
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 255.255.255.255, timeout is 2 seconds:

Reply to request 0 from 160.89.48.15 (4 ms)
Reply to request 0 from 160.89.48.10 (4 ms)
Reply to request 0 from 160.89.48.19 (4 ms)
Reply to request 0 from 160.89.49.15 (4 ms)
Reply to request 1 from 160.89.48.15 (4 ms)
Reply to request 1 from 160.89.48.10 (4 ms)
Reply to request 1 from 160.89.48.19 (4 ms)
Reply to request 1 from 160.89.49.15 (4 ms)
Reply to request 2 from 160.89.48.15 (4 ms)
Reply to request 2 from 160.89.48.10 (4 ms)
Reply to request 2 from 160.89.48.19 (4 ms)
Reply to request 2 from 160.89.49.15 (4 ms)
Reply to request 3 from 160.89.48.15 (4 ms)
Reply to request 3 from 160.89.48.10 (4 ms)
Reply to request 3 from 160.89.48.19 (4 ms)
Reply to request 3 from 160.89.49.15 (4 ms)
Reply to request 4 from 160.89.48.15 (4 ms)
Reply to request 4 from 160.89.48.10 (4 ms)
Reply to request 4 from 160.89.48.19 (4 ms)
Reply to request 4 from 160.89.49.15 (4 ms)
```

Related Command

ping (privileged)

ping (privileged)

To check host reachability and network connectivity, use the **ping** (IP packet internet groper function) user EXEC command.

ping [*protocol*] {*host* | *address*}

Syntax Description

protocol	(Optional) Protocol keyword. The default is IP.
host	Host name of system to ping.
address	IP address of system to ping.

Command Mode

Privileged EXEC

Usage Guidelines

The **ping** command sends ICMP *Echo* messages. If the router receives an ICMP *Echo* message, it sends an ICMP *Echo Reply* message to the source of the ICMP *Echo* message.

You can use the IP **ping** command to diagnose serial line problems. By placing the local or remote CSU/DSU into loopback mode and pinging your own interface, you can isolate the problem to the router or leased line.

Multicast and broadcast pings are fully supported. When you ping the broadcast address of 255.255.255.255, the system will send out pings and print a list of all stations responding. You can also ping a local network to get a list of all systems that respond, as in the following example, where 128.111.3 is a local network:

```
ping 128.111.3.255
```

As a side-effect, you also can get a list of all multicast-capable hosts that are connected directly to the router from which you are pinging, as in the following example:

```
ping 224.0.0.1
```

To abort a ping session, type the escape sequence (by default, Ctrl-^ X, which is done by simultaneously pressing the Ctrl, Shift, and 6 keys, letting go, then pressing the X key).

Table 17-4 describes the test characters that the ping facility sends.

Table 17-4 Ping Test Characters

Char	Description
!	Each exclamation point indicates receipt of a reply.
.	Each period indicates the network server timed out while waiting for a reply.
U	Destination unreachable.
N	Network unreachable.
P	Protocol unreachable.
Q	Source quench.

Char	Description
M	Could not fragment.
?	Unknown packet type.

You can use the extended command mode of the **ping** command to specify the supported Internet header options, as shown in the following sample display.

Sample Display Showing Extended Command Sequence

To enter **ping** extended command mode, enter **yes** at the extended commands prompt of the **ping** command. The following display shows a sample **ping** extended command sequence.

```
Router# ping
Protocol [ip]:
Target IP address: 192.31.7.27
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address: 131.108.1.1
Type of service [0]:
Set DF bit in IP header? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.31.7.27, timeout is 2 seconds:
!!!!!
Success rate is 100 percent, round-trip min/avg/max = 1/3/4 ms
```

Table 17-5 describes significant fields shown in the display.

Table 17-5 IP Ping Internet Header Options Field Descriptions

Field	Description
Protocol [ip]:	Default is IP.
Target IP address:	Prompts for the IP address or host name of the destination node you plan to ping.
Repeat count [5]:	Number of ping packets that will be sent to the destination address. Default: 5.
Datagram size [100]:	Size of the ping packet (in bytes). Default: 100 bytes.
Timeout in seconds [2]:	Timeout interval. Default: 2 (seconds).
Extended commands [n]:	Specifies whether or not a series of additional commands appears. Many of the following displays and tables show and describe these commands. Default: no.
Source address:	IP address that appears in the ping packet as the source address.
Type of service [0]:	Internet service quality selection. See RFC 791 for more information. Default: 0.

Field	Description
Set DF bit in IP header?	Don't Fragment. Specifies that if the packet encounters a node in its path that is configured for a smaller MTU than the packet's MTU, that the packet is to be dropped and an error message is to be sent to the router at the packet's source address. If performance problems are encountered on the network, a node configured for a small MTU could be a contributing factor. This feature can be used to determine the smallest MTU in the path. Default: no.
Data pattern [0xABCD]:	Sets 16-bit hexadecimal data pattern. Default: 0xABCD. Varying the data pattern in this field (to all ones or all zeros for example) can be useful when debugging data sensitivity problems on CSU/DSUs, or detecting cable-related problems such as cross talk.
Loose, Strict, Record, Timestamp, Verbose [none]:	Supported Internet header options. The router examines the header options to every packet that passes through it. If it finds a packet with an invalid option, the router sends an ICMP *Parameter Problem* message to the source of the packet and discards the packet. The Internet header options follow: • Loose • Strict • Record—See the following section for more information on this helpful option. • Timestamp • Verbose Default: none. For more information on these header options, see RFC 791.
Sweep range of sizes [n]:	Allows you to vary the sizes of the echo packets being sent. This capability is useful for determining the minimum sizes of the MTUs configured on the nodes along the path to the destination address. Packet fragmentation contributing to performance problems can then be reduced.
!!!!!	Each exclamation point (!) indicates receipt of a reply. A period (.) indicates the network server timed out while waiting for a reply. Other characters may appear in the ping output display, depending on the protocol type.
Success rate is 100 percent	Percentage of packets successfully echoed back to the router. Anything less than 80 percent is usually considered problematic.
round-trip min/avg/max = 1/3/4 ms	Round-trip travel time intervals for the protocol echo packets, including minimum/average/maximum (in milliseconds).

Use the Record Route Option

Using the Record Route option to trace a path to a particular destination address. Be aware, however, that the **trace** EXEC command performs a similar function, but the latter does not have the nine-hop limitation.

Sample Display Showing the Record Route Option

The following display shows sample extended **ping** output when this option is specified:

```
Router# ping
Protocol [ip]:
Target IP address: fred
```

```
   Repeat count [5]:
   Datagram size [100]:
   Timeout in seconds [2]:
   Extended commands [n]: y
   Source address:
   Type of service [0]:
   Set DF bit in IP header? [no]:
   Data pattern [0xABCD]:
   Loose, Strict, Record, Timestamp, Verbose[none]: r
   Number of hops [ 9 ]:
   Loose, Strict, Record, Timestamp, Verbose[RV]:
   Sweep range of sizes [n]:
   Type escape sequence to abort.
   Sending 5, 100-byte ICMP Echos to 131.108.1.115, timeout is 2 seconds:
   Packet has IP options:  Total option bytes= 39, padded length=40
    Record route: <*> 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
            0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
```

The following display is a detail of the Echo packet section:

```
   0 in 4 ms.  Received packet has options
    Total option bytes= 40, padded length=40
    Record route: 160.89.80.31 131.108.6.10 131.108.1.7 131.108.1.115
            131.108.1.115 131.108.6.7 160.89.80.240 160.89.80.31 <*> 0.0.0.0
    End of list

   1 in 8 ms.  Received packet has options
    Total option bytes= 40, padded length=40
    Record route: 160.89.80.31 131.108.6.10 131.108.1.6 131.108.1.115
            131.108.1.115 131.108.6.7 160.89.80.240 160.89.80.31 <*> 0.0.0.0
    End of list

   2 in 4 ms.  Received packet has options
    Total option bytes= 40, padded length=40
    Record route: 160.89.80.31 131.108.6.10 131.108.1.7 131.108.1.115
   131.108.1.115 131.108.6.7 160.89.80.240 160.89.80.31 <*> 0.0.0.0
    End of list

   3 in 8 ms.  Received packet has options
    Total option bytes= 40, padded length=40
    Record route: 160.89.80.31 131.108.6.10 131.108.1.6 131.108.1.115
            131.108.1.115 131.108.6.7 160.89.80.240 160.89.80.31 <*> 0.0.0.0
    End of list

   4 in 4 ms.  Received packet has options
    Total option bytes= 40, padded length=40
    Record route: 160.89.80.31 131.108.6.10 131.108.1.7 131.108.1.115
            131.108.1.115 131.108.6.7 160.89.80.240 160.89.80.31 <*> 0.0.0.0
    End of list

   Success rate is 100 percent, round-trip min/avg/max = 4/5/8 ms
Router#
```

In this display, five ping echo packets are sent to the destination address 131.108.1.115. The echo packet detail section includes specific information about each of these echo packets.

The lines of **ping** output that are unique when the Record Route option is specified are described as follows.

The following line of output allows you to specify the number of hops that will be recorded in the route. Range: 1 through 9. Default: 9.

```
Number of hops [ 9 ]:
```

The following line of output indicates that IP header options have been enabled on the outgoing echo packets and shows the number of option bytes and padded bytes in the headers of these packets.

```
Packet has IP options:  Total option bytes= 39, padded length=40
```

The following lines of output indicate that the fields that will contain the IP addresses of the nodes in the routes have been zeroed out in the outgoing packets.

```
Record route: <*> 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
         0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
```

The following lines of output display statistics for the first of the five echo packets sent. 0 is the number assigned to this packet to indicate that it is the first in the series. 4 ms indicates the round trip travel time for the packet.

```
0 in 4 ms.  Received packet has options
 Total option bytes= 40, padded length=40
 Record route: 160.89.80.31 131.108.6.10 131.108.1.7 131.108.1.115
     131.108.1.115 131.108.6.7 160.89.80.240 160.89.80.31 <*> 0.0.0.0
```

The following line of output indicates that four nodes were included in the packet's route, including the router at source address 160.89.80.31, two intermediate nodes at addresses 131.108.6.10 and 131.108.1.7, and the destination node at address 131.108.1.115. The underlined address shows where the original route differs from the return route in the line that follows this line.

```
Record route: 160.89.80.31 131.108.6.10 131.108.1.7 131.108.1.115
```

The following line of output includes the addresses of the four nodes in the return path of the echo packet. The underlined address shows where the return route differs from the original route shown in the previous line of output.

```
131.108.1.115 131.108.6.7 160.89.80.240 160.89.80.31 <*> 0.0.0.0
```

Related Command

ping (user)

show access-lists

To display the contents of all current access lists, use the **show access-lists** privileged EXEC command.

show access-lists

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Sample Display

The following is sample output from the **show access-lists** command:

```
Router# show access-lists
Standard IP access list 19
    permit 131.108.19.0
    deny   0.0.0.0, wildcard bits 255.255.255.255
Standard IP access list 49
    permit 131.108.31.0, wildcard bits 0.0.0.255
    permit 131.108.194.0, wildcard bits 0.0.0.255
    permit 131.108.195.0, wildcard bits 0.0.0.255
    permit 131.108.196.0, wildcard bits 0.0.0.255
    permit 131.108.197.0, wildcard bits 0.0.0.255
Extended IP access list 101
    permit tcp 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255 eq 23
Type code access list 201
    permit 0x6001 0x0000
Type code access list 202
    permit 0x6004 0x0000
    deny   0x0000 0xFFFF
```

For information on how to configure access lists, refer to the "Configuring IP" chapter of the *Router Products Configuration Guide.*

Related Commands

access-list (extended)
access-list (standard)

show arp

To display the entries in the ARP table for the router, use the **show arp** privileged EXEC command.

show arp

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Sample Display

The following is sample output from the **show arp** command:

```
Router# show arp

Protocol   Address          Age (min)   Hardware Addr    Type    Interface

Internet   131.108.42.112   120         0000.a710.4baf   ARPA    Ethernet3
AppleTalk  4028.5           29          0000.0c01.0e56   SNAP    Ethernet2
Internet   131.108.42.114   105         0000.a710.859b   ARPA    Ethernet3
AppleTalk  4028.9           -           0000.0c02.a03c   SNAP    Ethernet2
Internet   131.108.42.121   42          0000.a710.68cd   ARPA    Ethernet3
Internet   131.108.36.9     -           0000.3080.6fd4   SNAP    TokenRing0
AppleTalk  4036.9           -           0000.3080.6fd4   SNAP    TokenRing0
Internet   131.108.33.9     -           0000.0c01.7bbd   SNAP    Fddi0
```

Table 17-6 describes significant fields shown in the first line of output in the display.

Table 17-6 Show ARP Field Descriptions

Field	Description
Protocol	Indicates the type of network address this entry includes.
Address	Network address that is mapped to the MAC address in this entry.
Age (min)	Indicates the interval (in minutes) since this entry was entered in the table, rather than the interval since the entry was last used. (The timeout value is 4 hours.)
Hardware Addr	MAC address mapped to the network address in this entry.
Type	Indicates the encapsulation type the router is using for the network address in this entry. Possible values include: • ARPA • SNAP • ETLK (EtherTalk) • SMDS
Interface	Indicates the interface associated with this network address.

show dnsix

To display state information and the current configuration of the DNSIX audit writing module, use the **show dnsix** privileged EXEC command.

show dnsix

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Sample Display

The following is sample output from the **show dnsix** command:

```
Router# show dnsix
   Audit Trail Enabled with  Source 128.105.2.5
          State: PRIMARY
          Connected to 128.105.2.4
          Primary 128.105.2.4
          Transmit Count 1
          DMDP retries 4
          Authorization Redirection List:
               128.105.2.4
          Record count: 0
          Packet Count: 0
          Redirect Rcv: 0
```

show hosts

To display the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of host names and addresses, use the **show hosts** EXEC command.

show hosts

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show hosts** command:

```
Router# show hosts

Default domain is CISCO.COM
Hame/address lookup uses domain service
Hame servers are 255.255.255.255
Host              Flag        Age   Type       Address(es)
SLAG.CISCO.COM    (temp, OK)  1     IP         131.108.4.10
CHAR.CISCO.COM    (temp, OK)  8     IP         192.31.7.50
CHAOS.CISCO.COM   (temp, OK)  8     IP         131.108.1.115
DIRT.CISCO.COM    (temp, EX)  8     IP         131.108.1.111
DUSTBIN.CISCO.COM (temp, EX)  0     IP         131.108.1.27
DREGS.CISCO.COM   (temp, EX)  24    IP         131.108.1.30
```

Table 17-7 describes significant fields shown in the display.

Table 17-7 Show Hosts Field Descriptions

Field	Description
Flag	A temporary entry is entered by a name server; the router removes the entry after 72 hours of inactivity. A perm entry is entered by a configuration command and is not timed out. Entries marked OK are believed to be valid. Entries marked ?? are considered suspect and subject to revalidation. Entries marked EX are expired.
Age	Indicates the number of hours since the router last referred to the cache entry.
Type	Identifies the type of address, for example, IP, CLNS, or X.121. If you have used the **ip hp-host** global configuration command, the **show hosts** command will display these host names as type HP-IP.
Address(es)	Shows the address of the host. One host may have up to eight addresses.

Related Command

clear host

show ip access-list

To display the contents of all current IP access lists, use the **show ip access-list** EXEC command.

show ip access-list [*access-list-number*]

Syntax Description

access-list-number	(Optional) Number of the IP access list to display. This is a decimal number from 1 to 199.

Defaults

Displays all standard and extended IP access lists.

Command Mode

EXEC

Usage Guidelines

The **show ip access-list** command provides output identical to the **show access-lists** command, except that it is IP-specific and allows you to specify a particular access list.

Sample Display

The following is sample output from the **show ip access-list** command:

```
Router# show ip access-list
Extended IP access list 101
   deny udp any any eq ntp
   permit tcp any any
   permit udp any any eq tftp
   permit icmp any any
   permit udp any any eq domain
```

show ip accounting

To display the active accounting or checkpointed database or to display access-list violations, use the **show ip accounting** EXEC command.

show ip accounting [**checkpoint**] [**output-packets** | **access-violations**]

Syntax Description

checkpoint	(Optional) Indicates that the checkpointed database should be displayed.
output-packets	(Optional) Indicates that information pertaining to packets that passed access control and were successfully routed should be displayed. If neither the **output-packets** nor **access-violations** keyword is specified, **output-packets** is the default.
access-violations	(Optional) Indicates that information pertaining to packets that failed access lists and were not routed should be displayed. If neither the **output-packets** nor **access-violations** keyword is specified, **output-packets** is the default.

Defaults

If neither the **output-packets** nor **access-violations** keyword is specified, show ip accounting displays information pertaining to packets that passed access control and were successfully routed.

Command Mode

EXEC

Usage Guidelines

If you do not specify any keywords, the **show ip accounting** command displays information about the active accounting database.

To display IP access violations, you must give the **access-violations** keyword on the command. If you do not specify the keyword, the command defaults to displaying the number of packets that have passed access lists and were routed.

To use this command, you must first enable IP accounting on a per-interface basis.

Sample Display

Following is sample output from the **show ip accounting** command:

```
Router# show ip accounting

   Source           Destination              Packets               Bytes
 131.108.19.40    192.67.67.20                   7                 306
 131.108.13.55    192.67.67.20                  67                2749
 131.108.2.50     192.12.33.51                  17                1111
 131.108.2.50     130.93.2.1                     5                 319
 131.108.2.50     130.93.1.2                   463               30991
 131.108.19.40    130.93.2.1                     4                 262
 131.108.19.40    130.93.1.2                    28                2552
```

```
131.108.20.2     128.18.6.100                39                2184
131.108.13.55    130.93.1.2                  35                3020
131.108.19.40    192.12.33.51              1986               95091
131.108.2.50     192.67.67.20               233               14908
131.108.13.28    192.67.67.53               390               24817
131.108.13.55    192.12.33.51            214669             9806659
131.108.13.111   128.18.6.23              27739             1126607
131.108.13.44    192.12.33.51             35412             1523980
192.31.7.21      130.93.1.2                  11                 824
131.108.13.28    192.12.33.2                 21                1762
131.108.2.166    192.31.7.130               797              141054
131.108.3.11     192.67.67.53                 4                 246
192.31.7.21      192.12.33.51             15696              695635
192.31.7.24      192.67.67.20                21                 916
131.108.13.111   128.18.10.1                 16                1137
```

The following is sample output from the **show ip accounting access-violations** command. The output pertains to packets that failed access lists and were not routed:

```
Router# show ip accounting access-violations

  Source           DestinationPacketsBytesACL
131.108.19.40    192.67.67.20       7             306  77
131.108.13.55    192.67.67.20       67            2749185
131.108.2.50     192.12.33.5117           1111140
131.108.2.50     130.93.2.1         5           319140
131.108.19.40    130.93.2.1 4            26277
Accounting data age is 41
```

Table 17-8 describes the fields shown in the displays.

Table 17-8 Show IP Accounting (and Access-Violation) Field Descriptions

Field	Description
Source	Source address of the packet.
Destination	Destination address of the packet.
Packets	Number of packets transmitted from the source address to the destination address. With the **access-violations** keyword, the number of packets transmitted from the source address to the destination address that violated an access control list.
Bytes	Sum of the total number of bytes (IP header and data) of all IP packets transmitted from the source address to the destination address. With the **access-violations** keyword, the total number of bytes transmitted from the source address to the destination address that violated an access-control list.
ACL	Number of the access list of the last packet transmitted from the source to the destination that failed an access list filter.

Related Commands

clear ip accounting
ip accounting
ip accounting-list
ip accounting-threshold
ip accounting-transits

show ip aliases

To display the router's IP addresses mapped to TCP ports (aliases) and SLIP addresses, which are treated similarly to aliases, use the **show ip aliases** EXEC command.

show ip aliases

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

To distinguish a SLIP address from a normal alias address, the command output uses the form SLIP TTY1 for the "port" number, where 1 is the auxiliary port.

Sample Display

The following is sample output from the **show ip aliases** command:

```
Router# show ip aliases

  IP Address    Port
131.108.29.245  SLIP TTY1
```

The display lists the IP address and corresponding port number.

Related Command

A dagger (†) indicates that the command is documented in another chapter.

show line †

show ip arp

To display the Address Resolution Protocol (ARP) cache, where SLIP addresses appear as permanent ARP table entries, use the **show ip arp** EXEC command.

show ip arp

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

ARP establishes correspondences between network addresses (an IP address, for example) and LAN hardware addresses (Ethernet addresses). A record of each correspondence is kept in a cache for a predetermined amount of time and then discarded.

Sample Display

The following is sample output from the **show ip arp** command:

```
Router# show ip arp

Protocol  Address          Age (min)     Hardware Addr   Type      Interface
Internet  131.108.62.192         187     0800.2010.a3b6  ARPA      Ethernet3
Internet  131.108.62.245          68     0800.200e.28f8  ARPA      Ethernet3
Internet  131.108.1.140          139     0000.0c01.2812  ARPA      Ethernet0
Internet  131.108.62.160         187     0800.200e.4dab  ARPA      Ethernet3
Internet  131.108.1.111           27     0800.2007.8866  ARPA      Ethernet0
Internet  131.108.1.117          119     0000.0c00.f346  ARPA      Ethernet0
Internet  131.108.1.115           28     0000.0c01.0509  ARPA      Ethernet0
Internet  131.108.1.77             1     0800.200e.57ce  ARPA      Ethernet0
Internet  192.31.7.29            225     aa00.0400.0234  ARPA      Ethernet2
Internet  192.31.7.17            118     2424.c01f.0711  ARPA      Ethernet2
Internet  192.31.7.18            135     0000.0c01.2817  ARPA      Ethernet2
Internet  192.31.7.21            119     2424.c01f.0715  ARPA      Ethernet2
Internet  131.108.1.33             1     0800.2008.c52e  ARPA      Ethernet0
Internet  131.108.62.1             -     0000.0c00.750f  ARPA      Ethernet3
Internet  131.108.31.35          119     0800.2010.8c5b  ARPA      Ethernet7
Internet  131.108.62.7            14     0000.0c00.33ce  ARPA      Ethernet3
Internet  131.108.1.55           155     0800.200e.e443  ARPA      Ethernet0
```

Table 17-9 describes significant fields shown in the display.

Table 17-9 Show IP ARP Field Displays

Field	Description
Protocol	Protocol for network address in the Address field.
Address	The network address that corresponds to Hardware Addr.
Age (min)	Age, in minutes, of the cache entry.
Hardware Addr	LAN hardware address a MAC address that corresponds to network address.
Type	Type of encapsulation: • ARPA—Ethernet • SNAP—RFC 1042 • SAP—IEEE 802.3
Interface	Interface to which this address mapping has been assigned.

show ip cache

To display the routing table cache used to fast switch IP traffic, use the **show ip cache** EXEC command.

show ip cache [*prefix mask*] [*type number*]

Syntax Description

prefix	(Optional) Display only the entries in the cache that match the prefix and mask combination.
mask	(Optional) Display only the entries in the cache that match the prefix and mask combination.
type	(Optional) Display only the entries in the cache that match the interface type and number combination.
number	(Optional) Display only the entries in the cache that match the interface type and number combination.

Command Mode

EXEC

Usage Guidelines

The **show ip cache** display shows MAC headers up to 92 bytes.

Sample Displays

The following is sample output from the **show ip cache** command:

```
Router# show ip cache

IP routing cache version 4490, 141 entries, 20772 bytes, 0 hash overflows
Minimum invalidation interval 2 seconds, maximum interval 5 seconds,
   quiet interval 3 seconds, threshold 0 requests
Invalidation rate 0 in last second, 0 in last 3 seconds
Last full cache invalidation occurred 0:06:31 ago

Prefix/Length       Age       Interface       MAC Header
131.108.1.1/32      0:01:09   Ethernet0/0     AA0004000134000000C0357430800
131.108.1.7/32      0:04:32   Ethernet0/0     00000C01281200000C0357430800
131.108.1.12/32     0:02:53   Ethernet0/0     00000C029FD000000C0357430800
131.108.2.13/32     0:06:22   Fddi2/0         00000C05A3E000000C035753AAAA0300
                                              00000800
131.108.2.160/32    0:06:12   Fddi2/0         00000C05A3E000000C035753AAAA0300
                                              00000800
131.108.3.0/24      0:00:21   Ethernet1/2     00000C026BC600000C03574D0800
131.108.4.0/24      0:02:00   Ethernet1/2     00000C026BC600000C03574D0800
131.108.5.0/24      0:00:00   Ethernet1/2     00000C04520800000C03574D0800
131.108.10.15/32    0:05:17   Ethernet0/2     00000C025FF500000C0357450800
131.108.11.7/32     0:04:08   Ethernet1/2     00000C010E3A00000C03574D0800
131.108.11.12/32    0:05:10   Ethernet0/0     00000C01281200000C0357430800
131.108.11.57/32    0:06:29   Ethernet0/0     00000C01281200000C0357430800
```

Table 17-10 describes significant fields shown in the display.

Table 17-10 Show IP Cache Field Descriptions

Field	Description
IP routing cache version	Version number of this table. This number is incremented any time the table is flushed.
entries	Number of valid entries.
bytes	Number of bytes of processor memory for valid entries.
hash overflows	Number of times autonomous switching cache overflowed.
Minimum invalidation interval	Minimum time delay between cache invalidation request and actual invalidation.
maximum interval	Maximum time delay between cache invalidation request and actual invalidation.
quiet interval	Length of time between cache flush requests before the cache will be flushed.
threshold n requests	Maximum number of requests that can occur while the cache is considered quiet.
Invalidation rate *n* in last *m* seconds	Number of cache invalidations during the last m second.s
0 in last 3 seconds	Number of cache invalidation requests during the last quiet interval.
Last full cache invalidation occurred nn:nn:nn ago	Time since last full cache invalidation was performed.
Prefix/Length	Network reachability information for cache entry.
Age	Age of cache entry.
Interface	Output interface type and number.
MAC Header	Layer 2 encapsulation information for cache entry.

The following is sample output from the **show ip cache** command with a prefix and mask specified:

```
Router# show ip cache 131.108.5.0 255.255.255.0

IP routing cache version 4490, 119 entries, 17464 bytes, 0 hash overflows
Minimum invalidation interval 2 seconds, maximum interval 5 seconds,
   quiet interval 3 seconds, threshold 0 requests
Invalidation rate 0 in last second, 0 in last 3 seconds
Last full cache invalidation occurred 0:11:56 ago

Prefix/Length       Age       Interface       MAC Header
131.108.5.0/24      0:00:34   Ethernet1/2     00000C04520800000C03574D0800
```

The following is sample output from the **show ip cache** command with an interface specified:

```
Router# show ip cache e0/2

IP routing cache version 4490, 141 entries, 20772 bytes, 0 hash overflows
Minimum invalidation interval 2 seconds, maximum interval 5 seconds,
   quiet interval 3 seconds, threshold 0 requests
Invalidation rate 0 in last second, 0 in last 3 seconds
Last full cache invalidation occurred 0:06:31 ago

Prefix/Length       Age       Interface       MAC Header
131.108.10.15/32    0:05:17   Ethernet0/2     00000C025FF500000C0357450800
```

show ip interface

To display the usability status of interfaces configured for IP, use the **show ip interface** EXEC command.

show ip interface [*type number*]

Syntax Description

type	(Optional) Interface type.
number	(Optional) Interface number.

Command Mode

EXEC

Usage Guidelines

A router automatically enters a directly connected route in the routing table if the interface is usable. A usable interface is one through which the router can send and receive packets. If the router determines that an interface is not usable, it removes the directly connected routing entry from the routing table. Removing the entry allows the router to use dynamic routing protocols to determine backup routes to the network (if any).

If the interface can provide two-way communication, the line protocol is marked "up." If the interface hardware is usable, the interface is marked "up."

If you specify an optional interface type, you will see only information on that specific interface.

If you specify no optional arguments, you will see information on all the interfaces.

Sample Display

The following is sample output from the **show ip interface** command:

```
Router# show ip interface

Ethernet0 is up, line protocol is up
  Internet address is 192.195.78.24, subnet mask is 255.255.255.240
  Broadcast address is 255.255.255.255
  Address determined by non-volatile memory
  MTU is 1500 bytes
  Helper address is not set
  Secondary address 131.192.115.2, subnet mask 255.255.255.0
  Directed broadcast forwarding is enabled
  Multicast groups joined: 224.0.0.1 224.0.0.2
  Outgoing access list is not set
  Inbound  access list is not set
  Proxy ARP is enabled
  Security level is default
  Split horizon is enabled
  ICMP redirects are always sent
  ICMP unreachables are always sent
  ICMP mask replies are never sent
  IP fast switching is enabled
  IP fast switching on the same interface is disabled
  IP SSE switching is disabled
  Router Discovery is disabled
```

```
IP output packet accounting is disabled
IP access violation accounting is disabled
TCP/IP header compression is disabled
Probe proxy name replies are disabled
```

Table 17-11 describes the fields shown in the display.

Table 17-11 Show IP Interface Field Descriptions

Field	Description
Ethernet0 is up	If the interface hardware is usable, the interface is marked “up.” For an interface to be usable, both the interface hardware and line protocol must be up.
line protocol is up	If the interface can provide two-way communication, the line protocol is marked “up.” For an interface to be usable, both the interface hardware and line protocol must be up.
Broadcast address	Shows the broadcast address.
Address determined by ...	Indicates how the IP address of the interface was determined.
MTU	Shows the MTU value set on the interface.
Helper address	Shows a helper address if one has been set.
Secondary address	Shows a secondary address if one has been set.
Directed broadcast forwarding	Indicates whether directed broadcast forwarding is enabled.
Multicast groups joined	List which multicast groups this interface is a member of.
Outgoing access list	Indicates whether the interface has an outgoing access list set.
Inbound access list	Indicates whether the interface has an incoming access list set.
Proxy ARP	Indicates whether Proxy ARP is enabled for the interface.
Security level	Specifies the IPSO security level set for this interface.
ICMP redirects	Specifies whether redirects will be sent on this interface.
ICMP unreachables	Specifies whether unreachable messages will be sent on this interface.
ICMP mask replies	Specifies whether mask replies will be sent on this interface.
IP fast switching	Specifies whether fast switching has been enabled for this interface. It is generally enabled on serial interfaces, such as this one.
IP SSE switching	Specifies whether IP SSE switching is enabled.
Router Discovery	Specifies whether the discovery process has been enabled for this interface. It is generally disabled on serial interfaces.
IP output packet accounting	Specifies whether IP accounting is enabled for this interface and what the threshold (maximum number of entries) is.
TCP/IP header compression	Indicates whether compression is enabled or disabled.
Probe proxy name	Indicates whether HP Probe proxy name replies are generated.

show ip masks

To display the masks used for network addresses and the number of subnets using each mask, use the **show ip masks** EXEC command.

show ip masks *address*

Syntax Description

address	Network address for which a mask is required.

Command Mode

EXEC

Usage Guidelines

The **show ip masks** command is useful for debugging when variable-length subnet masks (VLSM) are used. It shows the number of masks associated with the network and the number of routes for each mask.

Sample Display

The following is sample output from the **show ip masks** command:

```
Router# show ip masks 131.108.0.0
Mask            Reference count
255.255.255.255 2
255.255.255.0   3
255.255.0.0     1
```

show ip nhrp

To display the Next Hop Resolution Protocol (NHRP) cache, use the **show ip nhrp** EXEC command.

show ip nhrp [**dynamic** | **static**] [*type number*]

Syntax Description

dynamic	(Optional) Displays only the dynamic (learned) IP-to-NBMA address cache entries.
static	(Optional) Displays only the static IP-to-NBMA address entries in the cache (configured through the **ip nhrp map** command).
type	(Optional) Interface type about which to display the NHRP cache (for example, **atm**, **tunnel**).
number	(Optional) Interface number about which to display the NHRP cache.

Command Mode

EXEC

Sample Display

The following is sample output from the **show ip nhrp** command:

```
Router# show ip nhrp

10.0.0.2 255.255.255.255, ATM0/0 created 0:00:43 expire 1:59:16
  Type: dynamic Flags: authoritative
  NBMA address: 11.1111.1111.1111.1111.1111.1111.1111.1111.1111.11
10.0.0.1 255.255.255.255, Tunnel0 created 0:10:03 expire 1:49:56
  Type: static Flags: authoritative
  NBMA address: 11.1.1.2
Router#
```

Table 17-12 describes the fields in the display.

Table 17-12 Show IP NHRP Field Descriptions

Field	Description
100.0.0.2 255.255.255.255	IP address and its network mask in the IP-to-NBMA address cache. The mask is currently always 255.255.255.255 because we do not support aggregation of NBMA information through NHRP.
ATM0/0 created 0:00:43	Interface type and number (in this case, ATM slot and port numbers) and how long ago it was created (hours:minutes:seconds).
expire 1:59:16	Time in which the positive and negative authoritative NBMA address will expire (hours:minutes:seconds). This value is based on the **ip nhrp holdtime** command.

Field	Description
Type	Value can be one of the following: • dynamic—NBMA address was obtained from NHRP Request packet. • static—NBMA address was statically configured.
Flags	Value can be one of the following: • authoritative—Indicates that the NHRP information was obtained from the Next Hop Server or router that maintains the NBMA-to-IP addresss mapping for a particular destination. • implicit—Indicates that the information was learned not from an NHRP request generated from the local router, but from an NHRP packet being forwarded or from an NHRP request being received by the local router. • negative—For negative caching; indicates that the requested NBMA mapping could not be obtained.
NBMA address	Nonbroadcast, multiaccess address. The address format is appropriate for the type of network being used (for example, ATM, Ethernet, SMDS, multipoint tunnel).

Related Command

ip nhrp map

show ip nhrp traffic

To display Next Hop Resolution Protocol (NHRP) traffic statistics, use the **show ip nhrp traffic** EXEC command.

show ip nhrp traffic

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show ip nhrp traffic** command:

```
Router# show ip nhrp traffic

Tunnel0
  request packets sent: 2
  request packets received: 4
  reply packets sent: 4
  reply packets received: 2
  register packets sent: 0
  register packets received: 0
  error packets sent: 0
  error packets received: 0
Router#
```

Table 17-13 describes the fields in the display.

Table 17-13 Show IP NHRP Traffic Field Descriptions

Field	Description
Tunnel 0	Interface type and number.
request packets sent	Number of NHRP Request packets originated from this station.
request packets received	Number of NHRP Request packets received by this station.
reply packets sent	Number of NHRP Reply packets originated from this station.
reply packets received	Number of NHRP Reply packets received by this station.
register packets sent	Number of NHRP Register packets originated from this station. Currently, our routers do not send Register packets, so this value is 0.
register packets received	Number of NHRP Register packets received by this station. Currently, our routers do not send Register packets, so this value is 0.
error packets sent	Number of NHRP Error packets originated by this station.
error packets received	Number of NHRP Error packets received by this station.

show ip redirects

To display the address of a default gateway (router) and the address of hosts for which a redirect has been received, use the **show ip redirects** EXEC command.

show ip redirects

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show ip redirects** command:

```
Router# show ip redirects
Default gateway is 160.89.80.29

Host               Gateway           Last Use    Total Uses  Interface
131.108.1.111      160.89.80.240         0:00             9  Ethernet0
128.95.1.4         160.89.80.240         0:00             4  Ethernet0
Router#
```

Related Command

ip redirects

show ip route

To display the entries in the routing table, use the **show ip route** EXEC command.

show ip route [*address* [*mask*]] | [*protocol*]

Syntax Description

address	(Optional) Address about which routing information should be displayed.
mask	(Optional) Argument for a subnet mask.
protocol	(Optional) Argument for a particular routing protocol, or **static** or **connected**.

Command Mode

EXEC

Sample Displays

The following is sample output from the **show ip route** command when entered when you do not specify an address:

```
Router# show ip route

Codes: I - IGRP derived, R - RIP derived, O - OSPF derived
       C - connected, S - static, E - EGP derived, B - BGP derived
       * - candidate default route, IA - OSPF inter area route
       E1 - OSPF external type 1 route, E2 - OSPF external type 2 route

Gateway of last resort is 131.119.254.240 to network 129.140.0.0

O E2 150.150.0.0 [160/5] via 131.119.254.6, 0:01:00, Ethernet2
E    192.67.131.0 [200/128] via 131.119.254.244, 0:02:22, Ethernet2
O E2 192.68.132.0 [160/5] via 131.119.254.6, 0:00:59, Ethernet2
O E2 130.130.0.0 [160/5] via 131.119.254.6, 0:00:59, Ethernet2
E    128.128.0.0 [200/128] via 131.119.254.244, 0:02:22, Ethernet2
E    129.129.0.0 [200/129] via 131.119.254.240, 0:02:22, Ethernet2
E    192.65.129.0 [200/128] via 131.119.254.244, 0:02:22, Ethernet2
E    131.131.0.0 [200/128] via 131.119.254.244, 0:02:22, Ethernet2
E    192.75.139.0 [200/129] via 131.119.254.240, 0:02:23, Ethernet2
E    192.16.208.0 [200/128] via 131.119.254.244, 0:02:22, Ethernet2
E    192.84.148.0 [200/129] via 131.119.254.240, 0:02:23, Ethernet2
E    192.31.223.0 [200/128] via 131.119.254.244, 0:02:22, Ethernet2
E    192.44.236.0 [200/129] via 131.119.254.240, 0:02:23, Ethernet2
E    140.141.0.0 [200/129] via 131.119.254.240, 0:02:22, Ethernet2
E    141.140.0.0 [200/129] via 131.119.254.240, 0:02:23, Ethernet2
```

The following is sample output that includes some IS-IS Level 2 routes learned:

```
Router# show ip route
Codes: I - IGRP derived, R - RIP derived, O - OSPF derived
       C - connected, S - static, E - EGP derived, B - BGP derived
       i - IS-IS derived
       * - candidate default route, IA - OSPF inter area route
E1 - OSPF external type 1 route, E2 - OSPF external type 2 route
       L1 - IS-IS level-1 route, L2 - IS-IS level-2 route
```

```
Gateway of last resort is not set

     160.89.0.0 is subnetted (mask is 255.255.255.0), 3 subnets
C       160.89.64.0 255.255.255.0 is possibly down,
          routing via 0.0.0.0, Ethernet0
i L2    160.89.67.0 [115/20] via 160.89.64.240, 0:00:12, Ethernet0
i L2    160.89.66.0 [115/20] via 160.89.64.240, 0:00:12, Ethernet0
```

Table 17-14 describes the fields shown in the displays.

Table 17-14 Show IP Route Field Descriptions

Field	Description
Codes	Codes defining how the route was learned and the type of route.
I	Route learned via IGRP.
R	Route learned from a RIP update.
O	Route learned from an OSPF update.
C	Directly connected network.
S	Statically defined route via the **ip route** command.
E	Route learned from EGP.
B	Route learned from BGP.
i	Router learned from IS-IS.
D	Route leaved via Enhanced IGRP.
*	Candidate default route. In the list of routes, the asterisk is the robin pointer. It indicates the last path used when a packet was forwarded. It applies only to non-fast-switched packets. The asterisk does not give an indication of which path will be used next when forwarding a non-fast-switched packet except when the paths are equal-cost paths. Paths can be equal cost only when runnign RIP.
IA	OSPF interarea route.
E1	OSPF external type 1 route.
E2	OSPF external type 2 route.
L1	IS-IS Level 1 route.
L2	IS-IS Level 2 route.
EX	External enhanced IGRP route.
150.150.0.0	Indicates the address of the remote network.
[160/5]	The first number in the brackets is the administrative distance of the information source; the second number is the metric for the route.
via 131.119.254.6	Specifies the address of the next router to the remote network.
0:01:00	Specifies the last time the route was updated in hours:minutes:seconds.
Ethernet 2	Specifies the interface through which the specified network can be reached.

The following is sample output from the **show ip route** command when you specify an address:

```
Router# show ip route 160.89.6.0
Routing entry for 160.89.6.0 (mask 255.255.255.0)
  Known via "connected", distance 0, metric 0 (connected)
  Tag 0
  Routing Descriptor Blocks:
  * directly connected, via Ethernet1
      Route metric is 0, traffic share count is 1
```

Table 17-15 describes the significant field shown in the display.

Table 17-15 Show IP Route Field Descriptions When You Specify an Address

Field	Description
Mask	Network mask associated with the route.
Connected	Routing protocol name, or **connected** or **static**.
Distance	Administrative distance.
Metric	Route metric that was either configured or learned from the particular route.
Routing Descriptor Blocks	Up to 4: Indicates the IP address of the next hop or the interface to which the particular route is connected.

show ip route summary

To display summary information about entries in the routing table, use the **show ip route summary** EXEC command.

show ip route summary

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show ip route summary** command:

```
Router# show ip route summary
Route Source    Networks    Subnets     Overhead    Memory (bytes)
connected       0           3           126         360
static          1           2           126         360
igrp 109        747         12          31878       91080
internal        3                                   360
Total           751         17          32130       92160
Router#
```

Table 17-16 describes the fields shown in the display:

Table 17-16 Show IP Route Summary Field Descriptions

Field	Description
Route Source	Routing protocol name, or **connected**, **static**, or **internal**. Internal—those routes that are in the primary routing table merely as markers to hold subnet routes. These routes are not owned by any routing protocol. There should be one of these internal routes for each subnetted network in the routing table.
Networks	The number of Class A, B, or C networks that are present in the routing table for each route source.
Subnets	The number of subnets that are present in the routing table for each route source, including host routes.
Overhead	Any additional memory involved in allocating the routes for the particular route source other than the memory specified under “Memory.”
Memory	The number of bytes allocated to maintain all the routes for the particular route source.

Related Command

show ip route

show ip tcp header-compression

To display statistics about TCP header compression, use the **show ip tcp header-compression** EXEC command.

show ip tcp header-compression

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show ip tcp header-compression** command:

```
Router# show ip tcp header-compression

TCP/IP header compression statistics:
  Interface Serial1: (passive, compressing)
    Rcvd:    4060 total, 2891 compressed, 0 errors
             0 dropped, 1 buffer copies, 0 buffer failures
    Sent:    4284 total, 3224 compressed,
             105295 bytes saved, 661973 bytes sent
             1.15 efficiency improvement factor
    Connect: 16 slots, 1543 long searches, 2 misses, 99% hit ratio
             Five minute miss rate 0 misses/sec, 0 max misses/sec
```

Table 17-17 describes significant fields shown in the display.

Table 17-17 Show IP TCP Header-Compression Field Descriptions

Field	Description
Rcvd:	
total	Total number of TCP packets received.
compressed	Total number of TCP packets compressed.
errors	Unknown packets.
dropped	Number of packets dropped due to invalid compression.
buffer copies	Number of packets that had to be copied into bigger buffers for decompression.
buffer failures	Number of packets dropped due to a lack of buffers.
Sent:	
total	Total number of TCP packets sent.
compressed	Total number of TCP packets compressed.
bytes saved	Number of bytes reduced.
bytes sent	Number of bytes sent.
efficiency improvement factor	Improvement in line efficiency because of TCP header compression.

Field	Description
Connect:	
number of slots	Size of the cache.
long searches	Indicates the number of times the software had to look to find a match.
misses	Indicates the number of times a match could not be made. If your output shows a large miss rate, then the number of allowable simultaneous compression connections may be too small.
hit ratio	Percentage of times the software found a match and was able to compress the header.
Five minute miss rate	Calculates the miss rate over the previous 5 minutes for a longer-term (and more accurate) look at miss rate trends.
max misses/sec	Maximum value of the previous field.

Related Command

ip tcp header-compression

show ip traffic

To display statistics about IP traffic, use the **show ip traffic** EXEC command.

show ip traffic

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show ip traffic** command:

```
Router# show ip traffic

IP statistics:
  Rcvd: 98 total, 98 local destination
        0 format errors, 0 checksum errors, 0 bad hop count
        0 unknown protocol, 0 not a gateway
        0 security failures, 0 bad options
  Frags:0 reassembled, 0 timeouts, 0 too big
        0 fragmented, 0 couldn't fragment
  Bcast:38 received, 52 sent
  Sent: 44 generated, 0 forwarded
        0 encapsulation failed, 0 no route
ICMP statistics:
  Rcvd: 0 checksum errors, 0 redirects, 0 unreachable, 0 echo
        0 echo reply, 0 mask requests, 0 mask replies, 0 quench
        0 parameter, 0 timestamp, 0 info request, 0 other
  Sent: 0 redirects, 3 unreachable, 0 echo, 0 echo reply
        0 mask requests, 0 mask replies, 0 quench, 0 timestamp
        0 info reply, 0 time exceeded, 0 parameter problem
UDP statistics:
  Rcvd: 56 total, 0 checksum errors, 55 no port
  Sent: 18 total, 0 forwarded broadcasts
TCP statistics:
  Rcvd: 0 total, 0 checksum errors, 0 no port
  Sent: 0 total
EGP statistics:
  Rcvd: 0 total, 0 format errors, 0 checksum errors, 0 no listener
  Sent: 0 total
IGRP statistics:
  Rcvd: 73 total, 0 checksum errors
  Sent: 26 total
HELLO statistics:
  Rcvd: 0 total, 0 checksum errors
  Sent: 0 total
ARP statistics:
  Rcvd: 20 requests, 17 replies, 0 reverse, 0 other
  Sent: 0 requests, 9 replies (0 proxy), 0 reverse
Probe statistics:
  Rcvd: 6 address requests, 0 address replies
0 proxy name requests, 0 other
  Sent: 0 address requests, 4 address replies (0 proxy)
        0 proxy name replies
```

Table 17-18 describes significant fields shown in the display.

Table 17-18 Show IP Traffic Field Descriptions

Field	Description
format errors	A gross error in the packet format, such as an impossible Internet header length.
bad hop count	Occurs when a packet is discarded because its time-to-live (TTL) field was decremented to zero.
encapsulation failed	Usually indicates that the router had no ARP request entry and therefore did not send a datagram.
no route	Counted when the router discards a datagram it did not know how to route.
proxy name reply	Counted when the router sends an ARP or Probe Reply on behalf of another host. The display shows the number of probe proxy requests that have been received and the number of responses that have been sent.

show sse summary

To display a summary of Silicon Switch Processor (SSP) statistics, use the **show sse summary** EXEC command.

show sse summary

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show sse summary** command:

```
Router# show sse summary
SSE utilization statistics

               Program words  Rewrite bytes  Internal nodes  Depth
Overhead                 499              1               8
IP                         0              0               0      0
IPX                        0              0               0      0
SRB                        0              0               0      0
CLNP                       0              0               0      0
IP access lists            0              0               0
Total used               499              1               8
Total free             65037         262143
Total available        65536         262144

Free program memory
  [499..65535]
Free rewrite memory
  [1..262143]

Internals
  75032 internal nodes allocated, 75024 freed
  SSE manager process enabled, microcode enabled, 0 hangs
  Longest cache computation 4ms, longest quantum 160ms at 0x53AC8
```

show standby

To display Hot Standby Router Protocol information, use the **show standby** EXEC command.

show standby

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show standby** command:

```
Router# show standby

Ethernet0 - Group 0
  Local state is Active, priority 100, may preempt
  Hellotime 3 holdtime 10
  Next hello sent in 0:00:00
  Hot standby IP address is 198.92.72.29 configured
  Active router is local
  Standby router is 198.92.72.21 expires in 0:00:07
  Tracking interface states for 2 interfaces, 2 up:
    Up    Ethernet0
    Up    Serial0
```

Table 17-19 describes the fields in the display.

Table 17-19 Show Standby Field Descriptions

Field	Description
Ethernet0 - Group 0	Interface type and number and Hot Standby group number for the interface.
Local state is ...	State of local router; can be one of the following: • Active—Current Hot Standby router • Standby—Router next in line to be the Hot Standby router
priority	Priority value of the router based on the **standby priority** command.
may preempt	Indicates that the router will attempt to assume control as the active router if its priority is greater than the current active router.
Hellotime	Time between hello packets in seconds, based on the **standby timers** command.
holdtime	Time (in seconds) before other routers declare the active or standby router to be down, based on the **standby timers** command.
Next hello sent in ...	Time in which the router will send the next hello packet (in hours:minutes:seconds).

Field	Description
Hot Standby IP address is ... configured	IP address of the current Hot Standby router. The word “configured” indicates that this address is known through the **standby ip** command. Otherwise, the address was learned dynamically through HSRP hello packets from other routers that do have the HSRP IP address configured.
Active router is ...	Value can be “local” or an IP address. Address of the current active Hot Standby router.
Standby router is ...	Value can be “local” or an IP address. Address of the “standby” router (the router that is next in line to be the Hot Standby router).
expires in	Time (in hours:minutes:seconds) in which the standby router will no longer be the standby router if the local router receives no hello packets from it.
Tracking interface states for ...	List of interfaces that are being tracked and their corresponding states. Based on the **standby track** command.

standby authentication

To configure an authentication string for the Hot Standby Router Protocol, use the **standby authentication** interface configuration command. To delete an authentication string, use the **no** form of this command.

standby [*group-number*] **authentication** *string*
no standby [*group-number*] **authentication** *string*

Syntax Description

group-number	(Optional) Group number on the interface to which this authentication string applies.
string	Authentication string. It can be up to eight characters in length. The default string is **cisco**.

Defaults

group-number: 0
string: **cisco**

Command Mode

Interface configuration

Usage Guidelines

The authentication string is transmitted unencrypted in all Hot Standby Router Protocol messages. The same authentication string must be configured on all routers on a cable to ensure interoperation. Authentication mismatch prevents a router from learning the designated Hot Standby IP address and the Hot Standby timer values from other routers configured with the Hot Standby Router Protocol. Authentication mismatch does not prevent protocol events such as one router taking over as the designated router.

When group number 0 is used, no group number is written to NVRAM, providing backward compatibility.

Example

In the following example, “word” is configured as the authentication string required to allow Hot Standby routers in group 1 to interoperate.

```
interface ethernet 0
standby 1 authentication word
```

standby ip

To activate the Hot Standby Router Protocol, use the **standby ip** interface configuration command. To disable the Hot Standby Router Protocol, use the **no** form of this command.

standby [*group-number*] **ip** [*ip-address*]
no standby [*group-number*] **ip** [*ip-address*]

Syntax Description

group-number	(Optional) Group number on the interface for which the Hot Standby Router Protocol is being activated.
ip-address	(Optional) IP address of the Hot Standby Router interface.

Defaults

group-number: 0
Hot Standby Router Protocol is disabled.

Command Mode

Interface configuration

Usage Guidelines

The **standby ip** command activates the Hot Standby Router Protocol on the configured interface. If an IP address is specified, that address is used as the designated address for the Hot Standby group. If no IP address is specified, the designated address is learned through the standby function. For the Hot Standby Router Protocol to elect a designated router, at least one router on the cable must have been configured with, or learned, the designated address. Configuring the designated address on the active router always overrides a designated address that is currently in use.

When the **standby ip** command is enabled on an interface, the handling of proxy ARP requests is changed (unless proxy ARP was disabled). If the interface's Hot Standby state is active, proxy ARP requests are answered using the Hot Standby group's MAC address. If the interface is in a different state, proxy ARP responses are suppressed.

When group number 0 is used, no group number is written to NVRAM, providing backward compatibility.

Example

In the following example, the Hot Standby protocol is enabled for group 1 on Ethernet interface 0. The IP address used by the Hot Standby group will be learned using the Hot Standby Router Protocol.

```
interface ethernet 0
standby 1 ip
```

standby preempt

To indicate that, when the local router has a Hot Standby priority higher than the current active router, the local router should attempt to assume control as the active router, use the **standby preempt** interface configuration command. To have the local router assume control as the active router only if it receives information indicating that there is no router currently in the active state (acting as the designated router), use the **no** form of this command.

standby [*group-number*] **preempt**
no standby [*group-number*] **preempt**

Syntax Description

group-number	(Optional) Group number on the interface for which the Hot Standby preemptive feature is being activated.

Defaults

group-number: 0
The local router assumes control as the active router only if it receives information indicating that there is no router currently in the active state.

Command Mode

Interface configuration

Usage Guidelines

When group number 0 is used, no group number is written to NVRAM, providing backward compatibility.

Example

In the following example, group 1 on Ethernet interface 0 is configured to preempt the current leader if the interface has a higher priority:

```
interface ethernet 0
standby 1 preempt
```

Related Commands

standby priority
standby track

standby priority

To prioritize a potential Hot Standby router, use the **standby priority** interface configuration command. To restore the priority to the default, use the **no** form of this command.

standby [*group-number*] **priority** *priority-number*
no standby [*group-number*] **priority** *priority-number*

Syntax Description

group-number	(Optional) Group number on the interface to which the priority number applies.
priority-number	Priority value. It is an integer from 0 through 255. The default is 100.

Defaults

group-number: 0
priority-number: 100

Command Mode

Interface configuration

Usage Guidelines

The assigned priority is used to help select the active and standby routers. Assuming preemption is enabled, the router with the highest priority becomes the designated active router. In case of ties, the primary IP addresses are compared, and the higher IP address has priority.

Note that the router's priority can change dynamically if an interface is configured with the **standby track** command and another interface on the router goes down.

When group number 0 is used, no group number is written to NVRAM, providing backward compatibility.

Example

In the following example, group number 1 on Ethernet interface 0 is assigned with priority 150:

```
interface ethernet 0
standby 1 priority 150
```

Related Commands

standby preempt
standby track

standby timers

To configure the time between hellos and the time before other routers declare the active Hot Standby or standby router to be down, use the **standby timers** interface configuration command. To restore the timers to their default values, use the **no** form of this command.

standby [*group-number*] **timers** *hellotime holdtime*
no standby [*group-number*] **timers** *hellotime holdtime*

Syntax Description

group-number	(Optional) Group number on the interface to which the timers apply. The default is 0.
hellotime	Hello interval in seconds. This is an integer from 1 through 255. The default is 1 second.
holdtime	Time in seconds before the active or standby router is declared to be down. This is an integer from 1 through 255. The default is 3 seconds.

Defaults

group-number: 0
hellotime: 1 second
holdtime: 3 seconds

Command Mode

Interface configuration

Usage Guidelines

The **standby timers** command configures the time between standby hellos and the time before other routers declare the active or standby router to be down. Routers on which timer values are not configured can learn timer values from the active or standby router. The timers configured on the active router always override any other timer settings. All routers in a Hot Standby group should use the same timer values. Normally, holdtime is greater than or equal to 3 times *hellotime* (*holdtime* >= 3 x *hellotime*).

When group number 0 is used, no group number is written to NVRAM, providing backward compatibility.

Example

In the following example, for group number 1 on Ethernet interface 0, the time between hello packets is set to 5 seconds, and the time after which a router is considered to be down is set to 15 seconds:

```
interface ethernet 0
standby 1 ip
standby 1 timers 5 15
```

standby track

To configure an interface so that the router's Hot Standby priority changes based on the availability of other interfaces, use the **standby track** interface configuration command. To remove the tracking, use the **no** form of this command.

standby [*group-number*] **track** *type number* [*interface-priority*]
no standby [*group-number*] **track** *type number* [*interface-priority*]

Syntax Description

group-number	(Optional) Group number on the interface to which the tracking applies.
type	Interface type (combined with interface number) that will be tracked.
number	Interface number (combined with interface type) that will be tracked.
interface-priority	(Optional) Amount by which the Hot Standby priority for the router is decremented (or incremented) when the interface goes down (or comes back up). The default value is 10.

Defaults

group-number: 0
interface-priority: 10

Command Mode

Interface configuration

Usage Guidelines

This command ties the router's Hot Standby priority to the availability of its interfaces. It is useful for tracking interfaces that are not configured for the Hot Standby Router Protocol.

When a tracked interface goes down, the Hot Standby priority of the router decreases by 10. If an interface is not tracked, its state changes do not affect the Hot Standby priority of the router. For each interface configured for Hot Standby, you can configure a separate list of interfaces to be tracked.

The optional argument *interface-priority* specifies how much to decrement the router's Hot Standby priority by when a tracked interface goes down. When the tracked interface comes back up, the router's priority is incremented by the same amount.

When multiple tracked interfaces are down and *interface-priority* values have been configured, these configured priority decrements are cumulative. If tracked interfaces are down, but none of them were configured with priority decrements, the default decrement is 10 and it is noncumulative.

When group number 0 is used, no group number is written to NVRAM, providing backward compatibility.

Example

In the following example, Ethernet interface 1 tracks Ethernet interface 0 and serial interface 0. If one or both of these two interfaces go down, the Hot Standby priority of the router decreases by 10. Because the default Hot Standby priority is 100, the priority becomes 90 when one or both of the tracked interfaces go down.

```
interface ethernet 1
ip address 198.92.72.37 255.255.255.240
no ip redirects
standby track ethernet 0
standby track serial 0
standby preempt
standby ip 198.92.72.46
```

Related Commands

standby preempt
standby priority

term ip netmask-format

To specify the format in which netmasks are displayed in **show** command output, use the **term ip netmask-format** EXEC command. To restore the default display format, use the **no** form of this command.

term ip netmask-format {**bitcount** | **decimal** | **hexadecimal**}
term no ip netmask-format [**bitcount** | **decimal** | **hexadecimal**]

Syntax Description

bitcount	Addresses are followed by a slash and the total number of bits in the netmask. For example, 131.108.11.55/24 indicates that the netmask is 24 bits.
decimal	Netmasks are displayed in dotted decimal notation (for example, 255.255.255.0).
hexadecimal	Netmasks are displayed in hexadecimal format, as indicated by the leading 0X (for example, 0XFFFFFF00).

Default

Netmasks are displayed in dotted decimal format.

Command Mode

EXEC

Usage Guidelines

IP uses a 32-bit mask that indicates which address bits belong to the network and subnetwork fields and which bits belong to the host field. This is called a netmask. By default, **show** commands display an IP address and then its netmask in dotted decimal notation. For example, a subnet would be displayed as 131.108.11.55 255.255.255.0.

However, you can specify that the display of the network mask appear in hexadecimal format or bit count format instead. The hexadecimal format is commonly used on UNIX systems. The above example would be displayed as 131.108.11.55 0XFFFFFF00.

The bitcount format for displaying network masks is to append a slash (/) and the total number of bits in the netmask to the address itself. The above example would be displayed as 131.108.11.55/24.

Example

The following example specifies that network masks for the session be displayed in bitcount notation in the output of **show** commands:

```
term ip netmask-format bitcount
```

trace (user)

To discover the routes the router's packets follow when traveling to their destination, use the **trace** user EXEC command.

trace ip *destination*

Syntax Description

destination	Destination address or host name on the command line. The default parameters for the appropriate protocol are assumed and the tracing action begins.

Command Mode

EXEC

Usage Guidelines

The **trace** command works by taking advantage of the error messages generated by routers when a datagram exceeds its time-to-live (TTL) value.

The **trace** command starts by sending probe datagrams with a TTL value of one. This causes the first router to discard the probe datagram and send back an error message. The **trace** command sends several probes at each TTL level and displays the round-trip time for each.

The **trace** command sends out one probe at a time. Each outgoing packet may result in one or two error messages. A *time exceeded* error message indicates that an intermediate router has seen and discarded the probe. A *destination unreachable* error message indicates that the destination node has received the probe and discarded it because it could not deliver the packet. If the timer goes off before a response comes in, **trace** prints an asterisk (*).

The **trace** command terminates when the destination responds, when the maximum TTL is exceeded, or when the user interrupts the trace with the escape sequence. By default, to invoke the escape sequence, press Ctrl-^ X, which is done by simultaneously pressing the Ctrl, Shift, and 6 keys, letting go, then pressing the X key.

Common Trace Problems

Due to bugs in the IP implementation of various hosts and routers, the IP **trace** command may behave in odd ways.

Not all destinations will respond correctly to a *probe* message by sending back an *ICMP port unreachable* message. A long sequence of TTL levels with only asterisks, terminating only when the maximum TTL has been reached, may indicate this problem.

There is a known problem with the way some hosts handle an *ICMP TTL exceeded* message. Some hosts generate an *ICMP* message but they reuse the TTL of the incoming packet. Since this is zero, the ICMP packets do not make it back. When you trace the path to such a host, you may see a set of TTL values with asterisks (*). Eventually the TTL gets high enough that the *ICMP* message can get back. For example, if the host is six hops away, **trace** will time out on responses 6 through 11.

Sample Display Showing Trace IP Routes

The following display shows sample IP **trace** output when a destination host name has been specified:

```
Router# trace ip ABA.NYC.mil
Type escape sequence to abort.
Tracing the route to ABA.NYC.mil (26.0.0.73)
  1 DEBRIS.CISCO.COM (131.108.1.6) 1000 msec 8 msec 4 msec
  2 BARRNET-GW.CISCO.COM (131.108.16.2) 8 msec 8 msec 8 msec
  3 EXTERNAL-A-GATEWAY.STANFORD.EDU (192.42.110.225) 8 msec 4 msec 4 msec
  4 BB2.SU.BARRNET.NET (131.119.254.6) 8 msec 8 msec 8 msec
  5 SU.ARC.BARRNET.NET (131.119.3.8) 12 msec 12 msec 8 msec
  6 MOFFETT-FLD-MB.in.MIL (192.52.195.1) 216 msec 120 msec 132 msec
  7 ABA.NYC.mil (26.0.0.73) 412 msec 628 msec 664 msec
```

Table 17-20 describes the fields shown in the display.

Table 17-20 Trace Field Descriptions

Field	Description
1	Indicates the sequence number of the router in the path to the host.
DEBRIS.CISCO.COM	Host name of this router.
131.108.1.61	Internet address of this router.
1000 msec 8 msec 4 msec	Round-trip time for each of the three probes that are sent.

Table 17-21 describes the characters that can appear in **trace** output.

Table 17-21 IP Trace Text Characters

Char	Description
nn msec	For each node, the round-trip time in milliseconds for the specified number of probes.
*	The probe timed out.
?	Unknown packet type.
Q	Source quench.
P	Protocol unreachable.
N	Network unreachable.
U	Port unreachable.
H	Host unreachable.

Related Command

trace (privileged)

trace (privileged)

To discover the routes the router's packets follow when traveling to their destination, use the **trace** privileged EXEC command.

trace [*destination*]

Syntax Description

destination	(Optional) Destination address or host name on the command line. The default parameters for the appropriate protocol are assumed and the tracing action begins.

Command Mode

Privileged EXEC

Usage Guidelines

The **trace** command works by taking advantage of the error messages generated by routers when a datagram exceeds its time-to-live (TTL) value.

The **trace** command starts by sending probe datagrams with a TTL value of one. This causes the first router to discard the probe datagram and send back an error message. The **trace** command sends several probes at each TTL level and displays the round-trip time for each.

The **trace** command sends out one probe at a time. Each outgoing packet may result in one or two error messages. A *time exceeded* error message indicates that an intermediate router has seen and discarded the probe. A *destination unreachable* error message indicates that the destination node has received the probe and discarded it because it could not deliver the packet. If the timer goes off before a response comes in, **trace** prints an asterisk (*).

The **trace** command terminates when the destination responds, when the maximum TTL is exceeded, or when the user interrupts the trace with the escape sequence. By default, to invoke the escape sequence, press Ctrl-^ X, which is done by simultaneously pressing the Ctrl, Shift, and 6 keys, letting go, then pressing the X key.

To use nondefault parameters and invoke an extended **trace** test, enter the command without a destination argument. You will be stepped through a dialog to select the desired parameters.

Common Trace Problems

Due to bugs in the IP implementation of various hosts and routers, the IP **trace** command may behave in odd ways.

Not all destinations will respond correctly to a *probe* message by sending back an *ICMP port unreachable* message. A long sequence of TTL levels with only asterisks, terminating only when the maximum TTL has been reached, may indicate this problem.

There is a known problem with the way some hosts handle an *ICMP TTL exceeded* message. Some hosts generate an *ICMP* message but they reuse the TTL of the incoming packet. Since this is zero, the ICMP packets do not make it back. When you trace the path to such a host, you may see a set of TTL values with asterisks (*). Eventually the TTL gets high enough that the *ICMP* message can get back. For example, if the host is six hops away, **trace** will time out on responses 6 through 11.

Sample Display Showing Trace IP Routes

The following display shows sample IP **trace** output when a destination host name has been specified:

```
Router# trace ABA.NYC.mil
Type escape sequence to abort.
Tracing the route to ABA.NYC.mil (26.0.0.73)
  1 DEBRIS.CISCO.COM (131.108.1.6) 1000 msec 8 msec 4 msec
  2 BARRNET-GW.CISCO.COM (131.108.16.2) 8 msec 8 msec 8 msec
  3 EXTERNAL-A-GATEWAY.STANFORD.EDU (192.42.110.225) 8 msec 4 msec 4 msec
  4 BB2.SU.BARRNET.NET (131.119.254.6) 8 msec 8 msec 8 msec
  5 SU.ARC.BARRNET.NET (131.119.3.8) 12 msec 12 msec 8 msec
  6 MOFFETT-FLD-MB.in.MIL (192.52.195.1) 216 msec 120 msec 132 msec
  7 ABA.NYC.mil (26.0.0.73) 412 msec 628 msec 664 msec
```

Table 17-22 describes the fields shown in the display.

Table 17-22 Trace Field Descriptions

Field	Description
1	Indicates the sequence number of the router in the path to the host.
DEBRIS.CISCO.COM	Host name of this router.
131.108.1.61	Internet address of this router.
1000 msec 8 msec 4 msec	Round-trip time for each of the three probes that are sent.

Sample Display Showing Extended IP Trace Dialog

The following display shows a sample **trace** session involving the extended dialog of the **trace** command:

```
Router# trace
Protocol [ip]:
Target IP address: mit.edu
Source address:
Numeric display [n]:
Timeout in seconds [3]:
Probe count [3]:
Minimum Time to Live [1]:
Maximum Time to Live [30]:
Port Number [33434]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Type escape sequence to abort.
Tracing the route to MIT.EDU (18.72.2.1)
  1 ICM-DC-2-V1.ICP.NET (192.108.209.17) 72 msec 72 msec 88 msec
  2 ICM-FIX-E-H0-T3.ICP.NET (192.157.65.122) 80 msec 128 msec 80 msec
  3 192.203.229.246 540 msec 88 msec 84 msec
  4 T3-2.WASHINGTON-DC-CNSS58.T3.ANS.NET (140.222.58.3) 84 msec 116 msec 88 msec
  5 T3-3.WASHINGTON-DC-CNSS56.T3.ANS.NET (140.222.56.4) 80 msec 132 msec 88 msec
  6 T3-0.NEW-YORK-CNSS32.T3.ANS.NET (140.222.32.1) 92 msec 132 msec 88 msec
  7 T3-0.HARTFORD-CNSS48.T3.ANS.NET (140.222.48.1) 88 msec 88 msec 88 msec
  8 T3-0.HARTFORD-CNSS49.T3.ANS.NET (140.222.49.1) 96 msec 104 msec 96 msec
  9 T3-0.ENSS134.T3.ANS.NET (140.222.134.1) 92 msec 128 msec 92 msec
 10 W91-CISCO-EXTERNAL-FDDI.MIT.EDU (192.233.33.1) 92 msec 92 msec 112 msec
 11 E40-RTR-FDDI.MIT.EDU (18.168.0.2) 92 msec 120 msec 96 msec
 12 MIT.EDU (18.72.2.1) 96 msec 92 msec 96 msec
```

Table 17-23 describes the fields that are unique to the extended trace sequence, as shown in the display.

Table 17-23 Trace Field Descriptions

Field	Description
Target IP address	You must enter a host name or an IP address. There is no default.
Source address	One of the interface addresses of the router to use as a source address for the probes. The router will normally pick what it feels is the best source address to use.
Numeric display	The default is to have both a symbolic and numeric display; however, you can suppress the symbolic display.
Timeout in seconds	The number of seconds to wait for a response to a probe packet. The default is 3 seconds.
Probe count	The number of probes to be sent at each TTL level. The default count is 3.
Minimum Time to Live [1]	The TTL value for the first probes. The default is 1, but it can be set to a higher value to suppress the display of known hops.
Maximum Time to Live [30]	The largest TTL value that can be used. The default is 30. The **trace** command terminates when the destination is reached or when this value is reached.
Port Number	The destination port used by the UDP probe messages. The default is 33434.
Loose, Strict, Record, Timestamp, Verbose	IP header options. You may specify any combination. The **trace** command issues prompts for the required fields. Note that **trace** will place the requested options in each probe; however, there is no guarantee that all routers (or end nodes) will process the options.
Loose Source Routing	Allows you to specify a list of nodes that must be traversed when going to the destination.
Strict Source Routing	Allows you to specify a list of nodes that must be the only nodes traversed when going to the destination.
Record	Allows you to specify the number of hops to leave room for.
Timestamp	Allows you to specify the number of time stamps to leave room for.
Verbose	If you select any option, the verbose mode is automatically selected and **trace** prints the contents of the option field in any incoming packets. You can prevent verbose mode by selecting it again, toggling its current setting.

Table 17-24 describes the characters that can appear in **trace** output.

Table 17-24 IP Trace Text Characters

Char	Description
nn msec	For each node, the round-trip time in milliseconds for the specified number of probes.
*	The probe timed out.
?	Unknown packet type.

Char	Description
Q	Source quench.
P	Protocol unreachable.
N	Network unreachable.
U	Port unreachable.
H	Host unreachable.

Related Command

trace (user)

transmit-interface

To assign a transmit interface to a receive-only interface, use the **transmit-interface** interface configuration command. To return to normal duplex Ethernet interfaces, use the **no** form of this command.

transmit-interface *type number*
no transmit-interface

Syntax Description

type	Transmit interface type to be linked with the (current) receive-only interface.
number	Transmit interface number to be linked with the (current) receive-only interface.

Default

Disabled

Command Mode

Interface configuration

Usage Guidelines

Receive-only interfaces are used commonly with microwave Ethernet links.

Example

The following example specifies Ethernet interface 0 as a simplex Ethernet interface:

```
interface ethernet 1
ip address 128.9.1.2
transmit-interface ethernet 0
```

tunnel mode

To set the encapsulation mode for the tunnel interface, use the **tunnel mode** interface configuration command. To set to the default, use the **no** form of this command.

tunnel mode {**aurp** | **cayman** | **dvmrp** | **eon** | **gre ip** [**multipoint**] | **nos**}
no tunnel mode

Syntax Description

aurp	AppleTalk Update Routing Protocol (AURP).
cayman	Cayman TunnelTalk AppleTalk encapsulation.
dvmrp	Distance Vector Multicast Routing Protocol.
eon	EON compatible CLNS tunnel.
gre ip	Generic route encapsulation (GRE) protocol over IP.
multipoint	(Optional) Enables a GRE tunnel to be used in a multipoint fashion. Can be used with the **gre ip** keyword only, and requires the use of the **tunnel key** command.
nos	KA9Q/NOS compatible IP over IP.

Default

GRE tunneling

Command Mode

Interface configuration

Usage Guidelines

You cannot have two tunnels using the same encapsulation mode with exactly the same source and destination address. The workaround is to create a loopback interface and source packets off of the loopback interface.

Cayman tunneling implements tunneling as designed by Cayman Systems. This enables our routers to interoperate with Cayman GatorBoxes. With Cayman tunneling, you can establish tunnels between two routers or between our router and a GatorBox. When using Cayman tunneling, you must not configure the tunnel with an AppleTalk network address. This means that there is no way to ping the other end of the tunnel.

Use DVMRP when a router connects to a mrouted router to run DVMRP over a tunnel. It is required to configure Protocol-Independent Multicast (PIM) and an IP address on a DVMRP tunnel.

Generic route encapsulation (GRE) tunneling can be done between our routers only. When using GRE tunneling for AppleTalk, you configure the tunnel with an AppleTalk network address. This means that you can ping the other end of the tunnel.

For multipoint GRE tunnels, a tunnel key must be configured. Unlike other tunnels, the tunnel destination is optional. However, if the tunnel destination is supplied, it must map to an IP multicast address.

Examples

The following example enables Cayman tunneling:

```
interface tunnel 0
tunnel source ethernet 0
tunnel destination 131.108.164.19
tunnel mode cayman
```

The following example enables GRE tunneling:

```
interface tunnel 0
appletalk cable-range 4160-4160 4160.19
appletalk zone Engineering
tunnel source ethernet0
tunnel destination 131.108.164.19
tunnel mode gre ip
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

appletalk cable-range †
appletalk zone †
tunnel destination †
tunnel source †

Index

INDEX

Symbols

Numerics

A

B

C

D

E

F

G

H

I

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Z