XPEDITION

NATIVE Command Line Interface Reference Manual

Revision Date: 09.15.03

ENTERASYS NETWORKS™

9032553-23

NOTICE

Enterasys Networks reserves the right to make changes in specifications and other information contained in this document and its web site without prior notice. The reader should in all cases consult Enterasys Networks to determine whether any such changes have been made.

The hardware, firmware, or software described in this document is subject to change without notice.

IN NO EVENT SHALL ENTERASYS NETWORKS BE LIABLE FOR ANY INCIDENTAL, INDIRECT, SPECIAL, OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING BUT NOT LIMITED TO LOST PROFITS) ARISING OUT OF OR RELATED TO THIS DOCUMENT, WEB SITE, OR THE INFORMATION CONTAINED IN THEM, EVEN IF ENTERASYS NETWORKS HAS BEEN ADVISED OF, KNEW OF, OR SHOULD HAVE KNOWN OF, THE POSSIBILITY OF SUCH DAMAGES.

Enterasys Networks, Inc.
50 Minuteman Road
Andover, MA 01810

Part Number: 9032553-23 September 2003

ENTERASYS NETWORKS, INC.
PROGRAM LICENSE AGREEMENT

BEFORE OPENING OR UTILIZING THE ENCLOSED PRODUCT, CAREFULLY READ THIS LICENSE AGREEMENT.

This document is an agreement ("Agreement") between the end user ("You") and Enterasys Networks, Inc. on behalf of itself and its Affiliates (as hereinafter defined) ("Enterasys") that sets forth Your rights and obligations with respect to the Enterasys software program (including any accompanying documentation, hardware or media) ("Program") in the package and prevails over any additional, conflicting or inconsistent terms and conditions appearing on any purchase order or other document submitted by You. "Affiliate" means any person, partnership, corporation, limited liability company, or other form of enterprise that directly or indirectly through one or more intermediaries, controls, or is controlled by, or is under common control with the party specified. This Agreement constitutes the entire understanding between the parties, and supersedes all prior discussions, representations, understandings or agreements, whether oral or in writing, between the parties with respect to the subject matter of this Agreement. The Program may be contained in firmware, chips or other media.

BY INSTALLING OR OTHERWISE USING THE PROGRAM, YOU REPRESENT THAT YOU ARE AUTHORIZED TO ACCEPT THESE TERMS ON BEHALF OF THE END USER (IF THE END USER IS AN ENTITY ON WHOSE BEHALF YOU ARE AUTHORIZED TO ACT, "YOU" AND "YOUR" SHALL BE DEEMED TO REFER TO SUCH ENTITY) AND THAT YOU AGREE THAT YOU ARE BOUND BY THE TERMS OF THIS AGREEMENT, WHICH INCLUDES, AMONG OTHER PROVISIONS, THE LICENSE, THE DISCLAIMER OF WARRANTY AND THE LIMITATION OF LIABILITY. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT OR ARE NOT AUTHORIZED TO ENTER INTO THIS AGREEMENT, ENTERASYS IS UNWILLING TO LICENSE THE PROGRAM TO YOU AND YOU AGREE TO RETURN THE UNOPENED PRODUCT TO ENTERASYS OR YOUR DEALER, IF ANY, WITHIN TEN (10) DAYS FOLLOWING THE DATE OF RECEIPT FOR A FULL REFUND.

IF YOU HAVE ANY QUESTIONS ABOUT THIS AGREEMENT, CONTACT ENTERASYS NETWORKS, LEGAL DEPARTMENT AT (978) 684-1000.

You and Enterasys agree as follows:

1. LICENSE. You have the non-exclusive and non-transferable right to use only the one (1) copy of the Program provided in this package subject to the terms and conditions of this Agreement.

2. RESTRICTIONS. Except as otherwise authorized in writing by Enterasys, You may not, nor may You permit any third party to:

(i) Reverse engineer, decompile, disassemble or modify the Program, in whole or in part, including for reasons of error correction or interoperability, except to the extent expressly permitted by applicable law and to the extent the parties shall not be permitted by that applicable law, such rights are expressly excluded. Information necessary to achieve interoperability or correct errors is available from Enterasys upon request and upon payment of Enterasys' applicable fee.

(ii) Incorporate the Program, in whole or in part, in any other product or create derivative works based on the Program, in whole or in part.

(iii) Publish, disclose, copy, reproduce or transmit the Program, in whole or in part.

(iv) Assign, sell, license, sublicense, rent, lease, encumber by way of security interest, pledge or otherwise transfer the Program, in whole or in part.

(v) Remove any copyright, trademark, proprietary rights, disclaimer or warning notice included on or embedded in any part of the Program.

3. APPLICABLE LAW. This Agreement shall be interpreted and governed under the laws and in the state and federal courts of the Commonwealth of Massachusetts without regard to its conflicts of laws provisions. You accept the personal jurisdiction and venue of the Commonwealth of Massachusetts courts. None of the 1980 United Nations Convention on Contracts for the International Sale of Goods, the United Nations Convention on the Limitation Period in the International Sale of Goods, and the Uniform Computer Information Transactions Act shall apply to this Agreement.

4. EXPORT RESTRICTIONS. You understand that Enterasys and its Affiliates are subject to regulation by agencies of the U.S. Government, including the U.S. Department of Commerce, which prohibit export or diversion of certain technical products to certain countries, unless a license to export the Program is obtained from the U.S. Government or an exception from obtaining such license may be relied upon by the exporting party.

If the Program is exported from the United States pursuant to the License Exception CIV under the U.S. Export Administration Regulations, You agree that You are a civil end user of the Program and agree that You will use the Program for civil end uses only and not for military purposes.

If the Program is exported from the United States pursuant to the License Exception TSR under the U.S. Export Administration Regulations, in addition to the restriction on transfer set forth in Sections 1 or 2 of this Agreement, You agree not to (i) reexport or release the Program, the source code for the Program or technology to a national of a country in Country Groups D:1 or E:2 (Albania, Armenia, Azerbaijan, Belarus, Bulgaria, Cambodia, Cuba, Estonia, Georgia, Iraq, Kazakhstan, Kyrgyzstan, Laos, Latvia, Libya, Lithuania, Moldova, North Korea, the People's Republic of China, Romania, Russia, Rwanda, Tajikistan, Turkmenistan, Ukraine, Uzbekistan, Vietnam, or such other countries as may be designated by the United States Government), (ii) export to Country Groups D:1 or E:2 (as defined herein) the direct product of the Program or the technology, if such foreign produced direct product is subject to national security controls as identified on the U.S. Commerce Control List, or (iii) if the direct product of the technology is a complete plant or any major component of a plant, export to Country Groups D:1 or E:2 the direct product of the plant or a major component thereof, if such foreign produced direct product is subject to national security controls as identified on the U.S. Commerce Control List or is subject to State Department controls under the U.S. Munitions List.

5. UNITED STATES GOVERNMENT RESTRICTED RIGHTS. The enclosed Program (i) was developed solely at private expense; (ii) contains "restricted computer software" submitted with restricted rights in accordance with section 52.227-19 (a) through (d) of the Commercial Computer Software-Restricted Rights Clause and its successors, and (iii) in all respects is proprietary data belonging to Enterasys and/or its suppliers. For Department of Defense units, the Program is considered commercial computer software in accordance with DFARS section 227.7202-3 and its successors, and use, duplication, or disclosure by the Government is subject to restrictions set forth herein.

6. DISCLAIMER OF WARRANTY. EXCEPT FOR THOSE WARRANTIES EXPRESSLY PROVIDED TO YOU IN WRITING BY ENTERASYS, ENTERASYS DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, SATISFACTORY QUALITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON- INFRINGEMENT WITH RESPECT TO THE PROGRAM. IF IMPLIED WARRANTIES MAY NOT BE DISCLAIMED BY APPLICABLE LAW, THEN ANY IMPLIED WARRANTIES ARE LIMITED IN DURATION TO THIRTY (30) DAYS AFTER DELIVERY OF THE PROGRAM TO YOU.

7. LIMITATION OF LIABILITY. IN NO EVENT SHALL ENTERASYS OR ITS SUPPLIERS BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS, PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR RELIANCE DAMAGES, OR OTHER LOSS) ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM, EVEN IF ENTERASYS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THIS FOREGOING LIMITATION SHALL APPLY REGARDLESS OF THE CAUSE OF ACTION UNDER WHICH DAMAGES ARE SOUGHT.

THE CUMULATIVE LIABILITY OF ENTERASYS TO YOU FOR ALL CLAIMS RELATING TO THE PROGRAM, IN CONTRACT, TORT OR OTHERWISE, SHALL NOT EXCEED THE TOTAL AMOUNT OF FEES PAID TO ENTERASYS BY YOU FOR THE RIGHTS GRANTED HEREIN.

8. AUDIT RIGHTS. You hereby acknowledge that the intellectual property rights associated with the Program are of critical value to Enterasys and, accordingly, You hereby agree to maintain complete books, records and accounts showing (i) license fees due and paid, and (ii) the use, copying and deployment of the Program. You also grant to Enterasys and its authorized representatives, upon reasonable notice, the right to audit and examine during Your normal business hours, Your books, records, accounts and hardware devices upon which the Program may be deployed to verify compliance with this Agreement, including the verification of the license fees due and paid Enterasys and the use, copying and deployment of the Program. Enterasys' right of examination shall be exercised reasonably, in good faith and in a manner calculated to not unreasonably interfere with Your business. In the event such audit discovers non-compliance with this Agreement, including copies of the Program made, used or deployed in breach of this Agreement, You shall promptly pay to Enterasys the appropriate license fees. Enterasys reserves the right, to be exercised in its sole discretion and without prior notice, to terminate this license, effective immediately, for failure to comply with this Agreement. Upon any such termination, You shall immediately cease all use of the Program and shall return to Enterasys the Program and all copies of the Program.

9. OWNERSHIP. This is a license agreement and not an agreement for sale. You acknowledge and agree that the Program constitutes trade secrets and/or copyrighted material of Enterasys and/or its suppliers. You agree to implement reasonable security measures to protect such trade secrets and copyrighted material. All right, title and interest in and to the Program shall remain with Enterasys and/or its suppliers. All rights not specifically granted to You shall be reserved to Enterasys.

10. ENFORCEMENT. You acknowledge and agree that any breach of Sections 2, 4, or 9 of this Agreement by You may cause Enterasys irreparable damage for which recovery of money damages would be inadequate, and that Enterasys may be entitled to seek timely injunctive relief to protect Enterasys' rights under this Agreement in addition to any and all remedies available at law.

11. ASSIGNMENT. You may not assign, transfer or sublicense this Agreement or any of Your rights or obligations under this Agreement, except that You may assign this Agreement to any person or entity which acquires substantially all of Your stock or assets. Enterasys may assign this Agreement in its sole discretion. This Agreement shall be binding upon and inure to the benefit of the parties, their legal representatives, permitted transferees, successors and assigns as permitted by this Agreement. Any attempted assignment, transfer or sublicense in violation of the terms of this Agreement shall be void and a breach of this Agreement.

12. WAIVER. A waiver by Enterasys of a breach of any of the terms and conditions of this Agreement must be in writing and will not be construed as a waiver of any subsequent breach of such term or condition. Enterasys' failure to enforce a term upon Your breach of such term shall not be construed as a waiver of Your breach or prevent enforcement on any other occasion.

13. SEVERABILITY. In the event any provision of this Agreement is found to be invalid, illegal or unenforceable, the validity, legality and enforceability of any of the remaining provisions shall not in any way be affected or impaired thereby, and that provision shall be reformed, construed and enforced to the maximum extent permissible. Any such invalidity, illegality or unenforceability in any jurisdiction shall not invalidate or render illegal or unenforceable such provision in any other jurisdiction.

14. TERMINATION. Enterasys may terminate this Agreement immediately upon Your breach of any of the terms and conditions of this Agreement. Upon any such termination, You shall immediately cease all use of the Program and shall return to Enterasys the Program and all copies of the Program.

Contents

About this Manual

This manual provides reference information for the commands in the Enterasys X-Pedition Command Line Interface (CLI). For product information not available in this manual, see the manuals listed in *Related Documentation* on page xxviii.

What's New

The content of this manual includes the addition of new and extended capabilities for the following:

ACL

acl logging set deny-trap on page 15

DHCP

dhcp define parameters on page 213

DVMRP

dvmrp start on page 246

FDDI

fddi set fddi-fdx-mode on page 282

IGMP

igmp enable vlan on page 357
igmp start-snooping on page 371

IP-ROUTER

ip-router policy aggr-gen destination on page 465
ip-router policy create bgp-import-source on page 475
ip-router policy create rip-import-source on page 487
ip-router policy create ospf-import-source on page 484
ip-router policy import source on page 493
ip-router policy summarize route on page 497
ip-router show route on page 502

PIM-SM

pim Commands on page 699

PORT

port bmon on page 748

STATISTICS

statistics clear on page 1151

SYSTEM

system set idle-timeout on page 1252

Who should Read this Manual?

Read this manual if you are a network administrator responsible for configuring or managing the X-Pedition.

How to Use this Manual

The CLI commands and facilities are organized alphabetically in this manual. To locate information about a command, go to the chapter for the command or for the facility that contains the command. For example, to find information about the **configure** command, go to *configure Command* on page 205. To find information about the **interface add** command, go to *interface Commands* on page 373, then locate the description of the **interface add** command within that chapter.

Related Documentation

The X-Pedition documentation set includes the following items. Refer to these other documents to learn more about your product.

For Information About...	See the...
Installing and setting up the X-Pedition	*Enterasys X-Pedition Getting Started Guide*
How to use CLI (Command Line Interface) commands to configure and manage the X-Pedition	*Enterasys X-Pedition User Reference Manual*
SYSLOG messages and SNMP traps	*Enterasys X-Pedition Error Reference Manual*

CLI Parameter Types

The following table describes all the parameter types you can use with the CLI.

Data Type	Descripton	Example
conditional	A numerical conditional expression. Special symbols are used to describe a numerical condition: > (greater than), < (less than) and != (not equal to).	<1024 or >2048 or !=4096
hexadecimal	A hexadecimal number	a7 or 0xa7
hostname	Hostname of an IP host	gauguin or john-pc
hostname/IP	Hostname or IP address of a host	nagasaki or 10.43.1.4
keyword	A keyword described in the list of acceptable keywords in the online help	on or off
interface name or IP address	Name of an interface or its IP address **Note:** Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses.	int1 or 10.1.4.33
interface name list	A list of one or more interface names delimited by commas **Note:** Enterasys recommends that you use only alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.	int1 or int1,int2,int3
IP address	An IP address of the form x.x.x.x. Some commands may explicitly require a unicast or multicast address.	*10.1.2.3*

Data Type	Descripton	Example
IP address/mask	A pair of IP address and mask values. Depending on the command, the mask may be a network mask or filtering mask. The mask can be described using the traditional IP address syntax (255.0.0.0) or a CIDR syntax (/8).	10.1.4.0/255.255.255.0 or 10.1.4.0/24
IP address list	A list of IP addresses separated by spaces but enclosed in quotes.	*"10.1.4.4 10.1.5.5 10.1.6.6"*
IPX network address	An IPX network address in hexadecimal	
IPX network**.**node address	An IPX network and node address of the form <netaddr>.<macaddr> where <netaddr> is the network address of a host and <macaddr> is the node or MAC address of the IPX host. For some commands, if the node address is not given, the node address is assumed to be a wildcard.	a1b2c3d4.0820a1:f3:38:11 or aa89f383
IPX SAP server name	An alphanumeric string representing a valid IPX SAP server name where the following characters are illegal: "*./:;<=>?[]|	server1
MAC address	A MAC address specified in one of two forms: xx:xx:xx:xx:xx:xx or xxxxxx:xxxxxx	08:00:50:1a:2b:c3 or 080050:1a2bc3
number	An integer number	*100*
numerical range	A number or a range of numbers	5 or 7-10
port	A single port	et.1.4, gi.2.1, hs.3.1.100, or se.4.2.200
port list	A list of one or more ports. To specify a range of ports within a module, describe the range in parenthesis. You can also specify non-consecutive ports by using commas to separate them. The wildcard character (*) can also be used to specify all modules or all ports within a module	et.1.(3-8) or et.1.(1,3,5), hs.(1-2).1.100, or se.4.(1-3).200, gi.2.*
slot number	A list of one or more occupied slots in the X-Pedition	1 or 7

Data Type	Descripton	Example
string	A character string. To include spaces in a string, specify the entire string in double quotes (“).	abc or “abc def”
URL	A Uniform Resource Locator. The type of URL depends on the command where the URL is used. Currently, two URLs are supported: TFTP: *tftp://host/pathname* RCP: *rcp://username@host/pathname*	tftp://10.1.4.5/test/abc.txt rcp://dave@rtr/test/abc.txt

Getting Help

For additional support related to the Common CLI syntax or this document, contact Enterasys Networks using one of the following methods:

World Wide Web	http://www.enterasys.com
Phone	603-332-9400 1-800-872-8440 (toll-free in U.S. and Canada) For the Enterasys Networks Support toll-free number in your country: http://www.enterasys.com/support/gtac-all.html
Internet mail	support@enterasys.com
To send comments or suggestions concerning this document to the Technical Writing Department: **TechWriting@enterasys.com** Make sure to include the document Part Number in the email message.	

Before contacting Enterasys Networks for technical support, have the following information ready:

- Your Enterasys Networks service contract number
- A description of the failure
- A description of any action(s) already taken to resolve the problem (e.g., changing mode switches, rebooting the unit, etc.)
- The serial and revision numbers of all involved Enterasys Networks products in the network
- A description of your network environment (layout, cable type, etc.)
- Network load and frame size at the time of trouble (if known)
- The device history (i.e., have you returned the device before, is this a recurring problem, etc.)
- Any previous Return Material Authorization (RMA) numbers

Chapter 1

acl Commands

The **acl** commands allow you to create ACLs (Access Control Lists) and apply them to IP and IPX interfaces on the X-Pedition. An ACL permits or denies switching of packets based on criteria such as the packet's source address and destination address, TCP or UDP port number, and so on. When you apply an ACL to an interface, you can specify whether the ACL affects incoming traffic or outgoing traffic. You also can enable a log of the ACL's use.

Note: Using ACLs for packet filtering directly impacts NetFlow performance. Specific performance will vary based on the number and complexity of the ACLs.

Command Summary

Table 1 lists the **acl** commands. The sections following the table describe the command syntax.

Table 1. acl commands

acl <*name*> **apply interface** <*InterfaceName*> **input| output [logging on| off| deny-only| permit-only| on-syslog| deny-syslog| permit-syslog] [report-denied periodic| all] [policy local| external]**
acl <*name*> **apply interface-are** <*InterfaceName*> **input|output**
acl <*name*> **apply port** <*port list*> **input| output [logging on| off| on-syslog| deny-syslog| permit-syslog] [report-denied periodic| all] [policy local| external]**
acl <*name*> **apply service** <*ServiceName*> **[logging on|off|on-syslog|deny-syslog|permit-syslog]**
acl <*name*> **clearCounters aclname| all| interface| service| port**
acl logging set deny-trap or... **acl** <*number*> **apply interface** <*name*> **logging deny-only deny-trap**

Table 1. acl commands (Continued)

acl logging set syslog-only
acl logging set deny-report-frequency [<*number*>**]**
acl <*name*> **permit|deny appletalk [nbp-brrq] [nbp-fwdrq] [nbp-lookup] [nbp-object** <*ObjectName*>**] [nbp-type** <*TypeName*>**] [nbp-zone** <*ZoneName*>**]**
acl <*name*> **permit|deny appletalk cable-range** <*range*>
acl <*name*> **permit|deny appletalk zone** <*ZoneName*>
acl <*name*> **permit|deny icmp** <*SrcAddr/Mask*> <*DstAddr/Mask*> **[log]**
acl <*name*> **permit|deny igmp** <*SrcAddr/Mask*> <*DstIP/mask*> **[log]**
acl <*name*> **permit|deny ip** <*SrcAddr/Mask*> <*DstAddr/Mask*> <*SrcPort*> <*DstPort*> <*tos*> [**accounting 5-minutes|15-minutes|hourly**] **[log]**
acl <*name*> **permit|deny ip-protocol** <*proto-num*> <*SrcAddr/Mask*> <*DstAddr/Mask*> <*tos*> **[log]**
acl <*name*> **permit|deny ipx** <*SrcAddr*> <*SrcSocket*> <*DstAddr*> <*DstSocket*> <*SrcNetMask*> <*DstNetMask*>
acl <*name*> **permit|deny ipxgns** <*ServerAddr*> <*ServiceType*> <*ServiceName*>
acl <*name*> **permit|deny ipxrip** <*FromNetwork*> <*ToNetwork*>
acl <*name*> **permit|deny ipxsap** <*ServerAddr*> <*ServiceType*> <*ServiceName*>
acl <*name*> **permit|deny ipxtype20**
acl <*name*> **permit|deny tcp** <*SrcAddr/Mask*> <*DstAddr/Mask*> <*SrcPort*> <*DstPort*> <*tos*> <*tos-mask*> [**accounting 5-minutes|15-minutes|hourly**] [**established**] **[log]**
acl <*name*> **permit|deny udp** <*SrcAddr/Mask*> <*DstAddr/Mask*> <*SrcPort*> <*DstPort*> <*tos*> <*tos-mask*> [**accounting 5-minutes|15-minutes|hourly**] **[log]**
acl-policy enable external| policy-routing-external
acl show [aclname <*string*>|**all] | [interface** <*string*>|**all-ip] | [service] | [port** <*port list*>|**all-ports] | [all]**

acl apply interface

Purpose

Apply an ACL to an interface.

Format

acl <*name*> **apply interface** <*InterfaceName*> **input| output**
[logging on| off| deny-only| permit-only| on-syslog| deny-syslog| permit-syslog]
[report-denied periodic| all] [policy local| external]

Mode

Configure

Description

The **acl apply interface** command applies a previously defined ACL to an interface. When you apply an ACL to an interface, you implicitly enable access control on that interface. You can apply an ACL to filter out inbound traffic, outbound traffic, or both inbound and outbound traffic. Inbound traffic is packets coming into the interface while outbound traffic is packets going out of that interface.

When you apply an ACL, you also can enable ACL Logging by using the **logging** keyword. When you enable ACL Logging on an interface, the X-Pedition displays ACL Logging messages on the console. The ACL log provides information such as the interface name, the ACL name, whether the packet is forwarded or not, and the internal details of the packet.

You can also specify if the ACL is allowed to be modified or removed from the interface by an external agent (such as a policy manager application) by using the **policy** keyword. If you do not specify the **policy** keyword, an external agent is allowed to modify or remove the applied ACL. Note that the **acl-policy enable external** command must be in the configuration before an external agent can modify or remove an applied ACL.

Parameters

<*name*> Name of the ACL. The ACL must already be defined. To define an ACL, use one of the commands described in other sections in this chapter.

<*InterfaceName*> Name of the interface to which you are applying the ACL.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

input Applies the ACL to filter out inbound traffic.

output Applies the ACL to filter out outbound traffic.

logging on|off|deny-only|permit-only

Enables or disables ACL logging for this interface. You can specify one of the following keywords:

off Disables all logging.

on Enables logging of packets that are dropped or forwarded because of ACL. If logging is turned on when you apply the ACL to an interface, you will override the rule-based logging values. In order to enable rule-based logging, the log option must remain off.

deny-only Enables logging of dropped packets only.

permit-only Enables logging of forwarded packets only.

on-syslog Enables logging of packets that are dropped or forwarded. Sends logging messages only to Syslog server.

deny-syslog Enables logging of dropped packets only. Sends logging messages only to Syslog server.

permit-syslog Enables logging of forwarded packets only. Sends logging messages only to Syslog server.

report-denied periodic| all

Enables reporting of all denied traffic for this ACL. Without this option, only the first denied packet of a particular traffic stream is reported. Subsequent packets are dropped at the line module with no reporting. You can specify one of the following keywords:

Note: The report-denied option is currently available for IP ACL's only.

periodic The XP will periodically report how many denied packets have been dropped by this ACL. The reporting period is configurable with the command 'acl logging set deny-report-frequency X'

all All packets dropped by this ACL will be reported as they are received. This option may severely impact forwarding performance, and is not recommended for normal network operation.

policy local|external

Allows or prevents an external agent from modifying or removing the applied ACL. You can specify one of the following keywords:

local External agent cannot modify or remove the applied ACL.

external External agent can modify or remove the applied ACL. This is the default.

Restrictions

- You can apply only one ACL at a time (IP or IPX) to inbound or outbound traffic on an interface. For example, if you define two ACLs, “ipacl1” and “ipacl2”, you cannot apply them both to inbound traffic. However, if you define one ACL for *inbound* traffic and one for *outbound* traffic, you can apply both ACLs. This does not limit the number of rules you may apply, it means only that you must include all rules in a single ACL.
- You can apply IP ACLs to IP interfaces only and IPX ACLs to IPX interfaces only.
- You may not apply ACLs to interface EN0 of the control module.

Examples

To apply ACL “100” to interface *int4* to filter out inbound traffic:

```
xp(config)# acl 100 apply interface int4 input
```

To apply ACL “nonfs” to interface *int16* to filter out outbound traffic and enable logging:

```
xp(config)# acl nonfs apply interface int16 output logging on
```

To apply ACL “100” to interface *int10* to filter out inbound traffic and enable logging to the Syslog server:

```
xp(config)# acl 100 apply interface int10 input logging on-syslog
```

acl apply interface-are

Purpose

Applies an Appletalk/ARE (Advanced Routing Engine) ACL to an interface.

Format

acl <*name*> **apply interface-are** <*InterfaceName*> **input|output**

Mode

ARE-Configure

Description

The **acl apply interface-are command** works very similarly to the **acl apply interface** command, with the following exceptions:

- If an Appletalk/ARE ACL contains Names Binding Protocol (NBP) rules (such as nbp-brrq, nbp-fwdrq, nbp-lookup, nbp-object, nbp-type, or nbp-zone), you may only apply it to the *input* of an interface.
- Like other ACLs for the X-Pedition, each direction (input and output) on an interface must have only one Appletalk/ARE ACL applied to it at a time. For example, although you can define two ARE ACLs, “areacl1” and “areacl2”, you cannot apply them both to the same interface. Unlike other ACLs, however, a single ARE ACL may be applied to *both* directions on one interface. In addition, a single ARE ACL may be applied to multiple interfaces.
- After applying an ARE ACL containing zone or cable-range rules, it may be necessary to reboot all network X-Peditions in order to see the new ACL. Alternatively, you may disable or disconnect the Routing Maintenance Protocol (RTMP) until all routes have been removed from each network X-Pedition routing table.

  Note: No special action is required for ACLs containing only NBP rules.
- There are no logging options available for ARE ACLs at this time.

Parameters

<*name*>	Name of the ACL. The ACL must already be defined. To define an ACL, use one of the commands described in other sections in this chapter.
<*InterfaceName*>	Name of the interface to which you are applying the ACL.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

input Applies the ACL to filter out inbound traffic.

output Applies the ACL to filter out outbound traffic.

Restrictions

- You may only use this command in ARE-Configure mode. For more about this mode, please see Chapter 6, *are Commands*.
- You may apply only one ARE ACL to an interface at a time.
- You may not apply ACLs to interface EN0 of the control module.

Examples

To apply ACL “ar10” to interface *int4* to filter out inbound traffic:

```
xp(are-config)# acl ar10 apply interface-are int4 input
```

To apply ACL “ar12” to interface *int16* to filter out outbound traffic:

```
xp(are-config)# acl ar12 apply interface-are int16 output
```

acl apply port

Purpose

Apply an ACL to one or more ports operating in Layer-4 bridging mode. The **acl apply port** applies a previously defined ACL to one or more ports. This command applies only to ports operating in Layer-4 bridging mode. The ACLs applied to a Layer-4 bridging port are only used with bridged traffic. Routed traffic is still subject to the ACLs attached to the interface.

Format

acl <*name*> **apply port** <*port list*> **input| output [logging on| off| on-syslog| deny-syslog| permit-syslog] [report-denied periodic| all] [policy local| external]**

Mode

Configure

Parameters

<*name*> Name of the ACL. The ACL must already be defined. To define an ACL, use one of the commands described in other sections in this chapter.

<*port list*> Specifies the port(s) in the Layer-4 bridging VLAN to which you are applying the ACL.

input Applies the ACL to filter out inbound traffic.

output Applies the ACL to filter out outbound traffic.

logging on|off

Enables or disables ACL logging for this port. You can specify one of the following keywords:

- **on** Enables logging of packets that are dropped or forwarded because of ACL.
- **off** Disables all logging.
- **on-syslog** Enables logging of packets that are dropped or forwarded. Sends logging messages only to Syslog server.
- **deny-syslog** Enables logging of dropped packets only. Sends logging messages only to Syslog server.
- **permit-syslog** Enables logging of forwarded packets only. Sends logging messages only to Syslog server.

report-denied periodic| all

Enables reporting of all denied traffic for this ACL. Without this option, only the first denied packet of a particular traffic stream is reported. Subsequent packets are dropped at the line module with no reporting. You can specify one of the following keywords:

Note: The report-denied option is currently available for IP ACL's only.

periodic The XP will periodically report how many denied packets have been dropped by this ACL. The reporting period is configurable with the command 'acl logging set deny-report-frequency X'

all All packets dropped by this ACL will be reported as they are received. This option may severely impact forwarding performance, and is not recommended for normal network operation.

policy local|external

Allows or prevents an external agent from modifying or removing the applied ACL. You can specify one of the following keywords:

local External agent cannot modify or remove the applied ACL.

external External agent can modify or remove the applied ACL. This is the default.

Restrictions

The line cards that contain the specified ports must support Layer 4 bridging. The X-Pedition software checks the line card(s) and displays an error message if new line card(s) are necessary.

Examples

To apply ACL “l4” to slot 1, gigabit port 3 and slot 3, 10/100 port 6 for inbound traffic:

```
xp(config)# acl l4 apply port gi.1.2 et.3.6 input
```

To apply ACL “l4out” to slot 5, all ports for outbound traffic and enable logging:

```
xp(config)# acl l4out apply port et.5.* output logging on
```

To apply ACL “l4” to slot 3, all ports for outbound traffic and enable logging to the Syslog server:

```
xp(config)# acl 14 apply port et.3.* output logging on-syslog
```

acl apply service

Purpose

Apply an ACL to a service on the X-Pedition.

Format

acl <*name*> **apply service** <*ServiceName*> **[logging on|off|on-syslog|deny-syslog|permit-syslog]**

Mode

Configure

Description

The **acl apply service** command applies a previously defined ACL to a service provided by the X-Pedition. A service is typically a server or agent running on the X-Pedition, for example, a Telnet server or SNMP agent. By applying an ACL to a service, you can control which host can access individual services on the X-Pedition. This type of ACL is known as a Service ACL. It does not control packets going *through* the X-Pedition. It only controls packets that are *destined* for the X-Pedition, specifically, one of the services provided by the X-Pedition. As a result, a Service ACL, by definition, is applied only to check for inbound traffic to the X-Pedition. The destination host of a Service ACL is by definition the X-Pedition. The destination port is the well-known port of the service.

When you apply an ACL, you also can enable ACL Logging by using the **logging** keyword. When you enable ACL Logging on an interface, the X-Pedition displays ACL Logging messages on the console. The ACL log provides information such as the interface name, the ACL name, whether the packet is forwarded or not, and the internal details of the packet.

In addition, you may apply an ACL to a service on a per-interface basis, based on the destination address defined by the ACL.

Parameters

<*name*> Name of the Service ACL. The ACL must already be defined. To define an ACL, use one of the commands described in other sections in this chapter.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses.

<*ServiceName*> Name of the service on the X-Pedition to which you are applying the ACL. Currently, the following services are supported:

http HTTP web server

snmp SNMP agent

telnet Telnet server

[logging [on|off]] Enables or disables ACL logging for this interface. You can specify one of the following keywords:

off Disables logging.

on Enables logging.

on-syslog
Enables logging of packets that are dropped or forwarded. Sends logging messages only to Syslog server.

deny-syslog
Enables logging of dropped packets only. Sends logging messages only to Syslog server.

permit-syslog
Enables logging of forwarded packets only. Sends logging messages only to Syslog server.

Restrictions

You can apply only one ACL of each type (IP or IPX) to a service at one time. For example, although you can define two ACLs, “ipacl1” and “ipacl2”, you cannot apply them both to the same service.

Examples

To permit access to the SNMP agent only from the host 10.4.3.33 (presumably an SNMP management station):

```
xp(config)# acl 100 permit udp 10.4.3.33
xp(config)# acl 100 apply service snmp
```

The following commands permit access to the Telnet server from hosts on the subnet 10.4.7.0/24 with a privileged source port. In addition, with logging enabled, all incoming Telnet accesses are logged to the console.

```
xp(config)# acl 120 permit tcp 10.4.7.0/24 any <1024
xp(config)# acl 120 apply service telnet logging on
```

The following commands permit access to the HTTP web server from subnet 10.12.4.0/24. Notice that even though the destination address and port are specified for this ACL (*10.12.7.44* and *any*

port), they are ignored. This service ACL will match only packets destined for the X-Pedition itself and the well-known port of the service (port 80 for HTTP).

```
xp(config)# acl 140 permit ip 10.12.4.0/24 any 10.12.7.44 any
xp(config)# acl 120 apply service http
```

acl clearCounters

Purpose

Clears one or all ACL counters.

Format

acl clearCounters aclname <*string*> **| all| interface| service| port**

Mode

Enable

Description

The **acl clearCounters** commands allows the user to clear ACL counters. With ACL logging enabled, the router prints out a message about whether a packet is forwarded or dropped and counters keep track of these statistics. With this command, the user can clear these ACL counters.

Parameters

aclname <*string*> Clears the counter based on the name of the ACL. Specify **all** to clear all ACLs.

all Clears all ACL counters.

interface <*string*> Clears ACL counters attached to specific interfaces. Specify **all-ip** to clear all counters with IP interfaces.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

service Clears ACL counters that are applied to services.

port <*port list*> Clears ACL counters on specific ports. Specify **all-ports** to clear counters on all ports.

Restrictions

None

Examples

To clear the ACL counters for ACL ‘engacl’:

```
xp# acl clearCounters aclname engacl
```

acl logging set deny-trap

Purpose

Allows sending of SNMP traps when an acl denies traffic.

Format

acl logging set deny-trap

or...

acl <*number*> **apply interface** <*name*> **logging deny-only deny-trap**

Mode

Configure

Description

This command enables the router to send the polACLDenied traps to a configured snmp target. The trap is generated when an ACL denies access to traffic matching the specified "deny" pattern in the ACL.

Restrictions

None.

acl logging set syslog-only

Purpose

Directs all acl reports only to the Syslog server.

Format

acl logging set syslog-only

Mode

Configure

Description

The **acl logging set syslog-only** commands allows the user to globally direct acl reporting to the Syslog server only (if defined). In order to view messages at the Syslog server, the Syslog level should be set to accept Informational Messages.

Restrictions

None.

acl logging set deny-report-frequency

Purpose

Changes the reporting interval (in seconds) for all ACL's using the **report-denied periodic** option.

Format

acl logging set deny-report-frequency [<*number*>**]**

Mode

Configure

Description

The **acl logging set deny-report-frequency** command allows users to set how often the router will report denied traffic with an ACL.

Parameters

<*number*>	The interval (15–3,600 seconds) at which to report denied traffic. By default, this interval is 15 seconds.

Restrictions

None.

acl permit|deny appletalk nbp

Purpose

Creates an Appletalk/ARE (Advanced Routing Engine) ACL with Name Binding Protocol (NBP) rules.

Format

acl <*name*> **permit|deny appletalk [nbp-brrq] [nbp-fwdrq] [nbp-lookup] [nbp-object** <*ObjectName*>**] [nbp-type** <*TypeName*>**] [nbp-zone** <*ZoneName*>**]**

Mode

ARE-Configure

Description

The **acl permit appletalk** and **acl deny appletalk** commands creates and defines an ACL to allow or block specific types Appletalk/ARE traffic from entering or leaving the X-Pedition. In this case, you use the commands to define Name Binding Protocol (NBP) rules. As with other ACLs for the X-Pedition, you may use both deny and permit commands within the same NBP-ruled ACL.

Parameters

<*name*> Name of this ACL. You may use a string of characters or a number.

nbp-prrq Name Binding Protocol Broadcast Request. Permit or deny all broadcast request packets.

nbp-fwdrq Name Binding Protocol Forward Request. Permit or deny all forward request packets.

nbp-lookup Name Binding Protocol Lookup Request. Permit or deny all lookup request packets.

nbp-object <*ObjectName*>
Permit or deny a specific machine. Specify the name of the machine (up to 32 characters). A single "~" can be used to request a match for 0 or more characters. If no machine is specified, the default "**any**" will be applied.

nbp-type <*TypeName*>
Permit or deny a specific type of machine. Specify the name of the machine (up to 32 characters). A single "~" can be used to request a match for 0 or more characters. If no machine type is specified, the default "**any**" will be applied.

nbp-zone <*ZoneName*>

Permit or deny requests from a specific zone. Specify the name of the zone. If no zone is specified, the default "**any**" will be applied.

Restrictions

You may only use this command in ARE-Configure mode. For more about this mode, please see Chapter 6, *are Commands*.

When you apply an Appletalk/ARE ACL to an interface, the X-Pedition appends an *implicit deny rule* to that ACL. The implicit deny rule denies ***all*** traffic. If you intend to allow all traffic that doesn't match your specified ACL rules to go through, you must *explicitly* define a rule to permit all traffic.

A single ACL can contain rules for NBP, zone, or cable-range, due to the fact that these three Appletalk/ARE ACL rules act independently of one another. In effect, a single ACL containing all three types of rule will act as if it were three different ACLs.

Examples

To permit all broadcast requests for ACL ar310:

```
xp(are-config)# acl ar310 permit appletalk nbp-brrq
```

The *implicit deny rule* will automatically cause all forward requests to be dropped.

To permit both broadcast and forward requests:

```
xp(are-config)# acl ar310 permit appletalk nbp-brrq nbp-fwdrq
```

To deny recognition of all laser printers on any zone:

```
xp(are-config)# acl ar310 deny appletalk nbp-object ~ nbp-type laserprinter
xp(are-config)# acl ar310 permit nbp-brrq nbp-fwdrq
```

To deny recognition of laser printer "printer 1":

```
xp(are-config)# acl ar310 deny appletalk nbp-object printer1 nbp-type laserprinter
xp(are-config)# acl ar310 permit nbp-brrq nbp-fwdrq
```

acl permit|deny appletalk cable-range

Purpose

Creates an Appletalk/ARE (Advanced Routing Engine) ACL containing cable range rules.

Format

acl <*name*> **permit|deny appletalk cable-range** <*range*>

Mode

ARE-Configure

Description

The **acl permit appletalk cable-range** and **acl deny appletalk cable-range** commands create and define an ACL to allow or block specific types Appletalk/ARE traffic from entering or leaving the X-Pedition. In this case, you use the commands to define cable range rules. Unlike with other ACLs for the X-Pedition, all cable range rules within an ACL must either permit or deny traffic; you may not mix the two commands.

Parameters

<*name*>	Name of this ACL. You may use a string of characters or a number.
<*range*>	Cable range for which to apply rule. Specify values between 1 and 65279, inclusive.

Restrictions

You may only use this command in ARE-Configure mode. For more about this mode, please see Chapter 6, *are Commands*.

All cable range rules within a single ACL must either permit or deny traffic. For example, if you create a rule, “acl appleacl deny appletalk cable-range 1-100”, any additional cable range rules for ACL “appleacl” must also be of the type “deny.”

When you apply an Appletalk/ARE ACL to an interface, the X-Pedition appends an *implicit deny rule* to that ACL. The implicit deny rule denies ***all*** traffic. If you intend to allow all traffic that doesn’t match your specified ACL rules to go through, you must *explicitly* define a rule to permit all traffic. Thusly, if you define an ACL to “deny” cable range access, any cable range not listed within the ACL will be permitted. If you define an ACL to “permit” cable range access, any cable range not listed within the ACL will be blocked.

A single ACL can contain rules for cable-range, NBP, and zone due to the fact that these three Appletalk/ARE ACL rules act independently of one another. In effect, a single ACL containing all three types of rule will act as if it were three different ACLs.

acl permit|deny appletalk zone

Purpose

Creates an Appletalk/ARE (Advanced Routing Engine) ACL containing zone rules.

Format

acl <*name*> **permit|deny appletalk zone** <*ZoneName*>

Mode

ARE-Configure

Description

The **acl permit appletalk zone** and **acl deny appletalk zone** commands create and define an ACL to allow or block specific types Appletalk/ARE traffic from entering or leaving the X-Pedition. In this case, you use the commands to define zone rules. Unlike with other ACLs for the X-Pedition, all zone rules within an ACL must either permit or deny traffic; you may not mix the two commands.

Note: The X-Pedition acts as either a *seed* or a *no-seed* router. If you set as a seed router, the X-Pedition will assign zone information to the networks to which it is attached.

Parameters

<*name*>	Name of this ACL. You may use a string of characters or a number.
<*ZoneName*>	The name of the zone for which to apply rule. Zone names may use a string of characters or numbers. (A *zone* refers a logical grouping of appletalk networks on a LAN or WAN.)

Restrictions

You may use this command only in ARE-Configure mode. For more about this mode, please see Chapter 6, *are Commands*.

All zone rules within a single ACL must either permit or deny traffic. For example, if you create a rule, “acl zacl deny appletalk zone myzone,” any additional zone rules for ACL “zacl” must also be of the type “deny.”

When you apply an Appletalk/ARE ACL to an interface, the X-Pedition appends an *implicit deny rule* to that ACL. The implicit deny rule denies ***all*** traffic. If you intend to allow all traffic that doesn’t match your specified ACL rules to go through, you must *explicitly* define a rule to permit all traffic. Thusly, if you define an ACL to “deny” zone access, any zone not listed within the ACL

will be permitted. If you define an ACL to “permit” zone access, any zone not listed within the ACL will be blocked.

A single ACL can contain rules for zone, NBP, and cable-range, due to the fact that these three Appletalk/ARE ACL rules act independently of one another. In effect, a single ACL containing all three types of rule will act as if it were three different ACLs.

Examples

If you wish to permit only those packets that emanate from zoneA (i.e., all other packets are ignored), enter the following:

```
acl 243 permit appletalk zone zoneA
```

If you wish to deny all packets from zoneB (i.e., all other packets are welcome), enter the following:

```
acl 245 deny appletalk zone zoneB
```

acl permit|deny icmp

Purpose

Create an ICMP ACL.

Format

acl <*name*> **permit|deny icmp** <*SrcAddr/Mask*> <*DstAddr/Mask*> <*tos*>

Mode

Configure

Description

The **acl permit icmp** and **acl deny icmp** commands define an ACL to allow or block ICMP traffic from entering or leaving the X-Pedition. For each parameter describing a flow, you can specify a value or use the keyword **any** to indicate a *wildcard* (“don’t care”) condition. When you specify only some of the parameters, the remaining fields will require the **any** keyword. If you do not specify any value for any field, the X-Pedition applies a wildcard condition to every field, giving the same effect as if you specify the **any** keyword.

Parameters

<*name*>	Name of this ACL. You can use a string of characters or a number.
<*SrcAddr/Mask*>	The source address and the filtering mask of this flow. If the source address is a network or subnet address, you must supply the filtering mask. Generally, the filtering mask is the network mask of this network or subnet. If the source address is that of a host then no mask is required. By default, if a mask is not supplied, the source address is treated as that of a host. You can specify the mask using the traditional IP address format (“255.255.0.0”) or the CIDR format (“/16”).
<*DstAddr/Mask*>	The destination address and the filtering mask of this flow. The same requirements and restrictions for <*SrcAddr/Mask*> apply to <*DstAddr/Mask*>.
<*tos*>	IP TOS (Type of Service) value. You can specify a TOS value from 0 – 255.
log	This optional parameter allows you to enable ACL logging for this specific ACL rule. If logging is turned on when you apply the ACL to an interface (e.g., with the *acl apply interface* on page 3), you will override the rule-based logging values. In order to enable rule-based logging, the log option must remain off.

Restrictions

When you apply an ACL to an interface, the X-Pedition appends an *implicit deny rule* to that ACL. The implicit deny rule denies ***all*** traffic. If you intend to allow all traffic that doesn't match your specified ACL rules to go through, you must *explicitly* define a rule to permit all traffic.

You may not apply ACLs to interface EN0 of the control module.

Examples

To deny ICMP traffic from the subnet 10.24.5.0 (with a 24 bit netmask) to any destination:

```
xp(config)# acl 310 deny icmp 10.24.5.0/24 any
```

To create an ACL to permit ICMP traffic from the host 10.12.28.44 to subnet 10.43.21.0:

```
xp(config)# acl 312 permit icmp 10.12.28.44 10.43.21.0/24
```

acl permit|deny igmp

Purpose

Create an IGMP ACL.

Format

acl <*name*> **permit|deny igmp** <*SrcAddr/Mask*> <*DstAddr/Mask*> <*tos*>

Mode

Configure

Description

The **acl permit igmp** and **acl deny igmp** commands define an ACL to allow or block IGMP traffic from entering or leaving the X-Pedition. For each parameter describing a flow, you can specify a value or use the keyword **any** to indicate a *wildcard* ("don't care") condition. When you specify only some of the parameters, the remaining fields will require the **any** keyword. If you do not specify any value for any field, the X-Pedition applies a wildcard condition to every field, giving the same effect as if you specify the **any** keyword.

<*name*>	Name of this ACL. You can use a string of characters or a number.
<*SrcAddr/Mask*>	The source address and the filtering mask of this flow. If the source address is a network or subnet address, you must supply the filtering mask. Generally, the filtering mask is the network mask of this network or subnet. If the source address is that of a host then no mask is required. By default, if a mask is not supplied, the source address is treated as that of a host. You can specify the mask using the traditional IP address format ("255.255.0.0") or the CIDR format ("/16").
<*DstAddr/Mask*>	The destination address and the filtering mask of this flow. The same requirements and restrictions for <*SrcAddr/Mask*> apply to <*DstAddr/Mask*>.
<*tos*>	IP TOS (Type of Service) value. You can specify a TOS value from 0 – 255.
log	This optional parameter allows you to enable ACL logging for this specific ACL rule.

Restrictions

When you apply an ACL to an interface, the X-Pedition appends an *implicit deny rule* to that ACL. The implicit deny rule denies ***all*** traffic. If you intend to allow all traffic that doesn't match your specified ACL rules to go through, you must *explicitly* define a rule to permit all traffic.

Examples

To create an ACL to deny IGMP traffic from the subnet 10.1.5.0 (with a 24 bit netmask) to any destination:

```
xp(config)# acl 410 deny igmp 10.1.5.0/24 any
```

To create an ACL to permit IGMP traffic from the host 10.33.34.44 to subnet 10.11.21.0:

```
xp(config)# acl 714 permit igmp 10.33.34.44 10.11.21.0/24
```

acl permit|deny ip

Purpose

Create an IP ACL.

Format

acl <*name***> permit|deny ip** <*SrcAddr/Mask*> <*DstAddr/Mask*> <*SrcPort*> <*DstPort*> <*tos*> <*tos-mask*> **any** [**accounting 5-minutes|15-minutes|hourly**] [**log**]

Mode

Configure

Description

The **acl permit ip** and **acl deny ip** commands define an Access Control List to allow or block IP traffic from entering or leaving the router. Unlike the more specific variants of the acl commands for **tcp** and **udp**, the IP version of the command includes IP-based protocols such as **tcp**, **udp**, **icmp** and **igmp**. For each parameter describing a flow, you can specify a value or use the keyword **any** to indicate a *wildcard* ("don't care") condition. When you specify only some of the parameters, the remaining fields will require the **any** keyword. If you do not specify any value for any field, the X-Pedition applies a wildcard condition to every field, giving the same effect as if you specify the **any** keyword.

The two exceptions to this rule are the optional parameters <*tos*> (type of service) and **accounting**. <*tos*> is a value from 0 to 255. The **accounting** keyword is only valid for the **permit** command, and can be placed anywhere on the command line.

Parameters

<*name*>	Name of this ACL. You can use a string of characters or a number. The string must be less than 100 characters.
<*SrcAddr/Mask*>	The source address and the filtering mask of this flow. If the source address is a network or subnet address, you must supply the filtering mask. Generally, the filtering mask is the network mask of this network or subnet. If the source address is that of a host then no mask is required. By default, if a mask is not supplied, the source address is treated as that of a host. You can specify the mask using the traditional IP address format ("255.255.0.0") or the CIDR format ("/16").
<*DstAddr/Mask*>	The destination address and the filtering mask of this flow. The same requirements and restrictions for <*SrcAddr/Mask*> apply to <*DstAddr/Mask*>.

<*SrcPort*>	For TCP or UDP, the number of the source TCP or UDP port. This field applies only to TCP or UDP traffic. If the incoming packet is ICMP or another non-TCP or non-UDP packet and you specified a source or destination port, the X-Pedition does not check the port value. The X-Pedition checks only the source and destination IP addresses in the packet. You can specify a range of port numbers using operator symbols; for example, 10-20 (between 10 and 20 inclusive), >1024 (greater than 1024), <1024 (less than 1024), !=1024 (not equal to 1024). The port numbers of some popular services are already defined as keywords. For example, for Telnet, you can enter the port number 23 as well as the keyword **telnet**.
<*DstPort*>	For TCP or UDP, the number of the destination TCP or UDP port. This field applies only to incoming TCP or UDP traffic. The same requirements and restrictions for <*SrcPort*> apply to <*DstPort*>.
<*tos*>	IP TOS (Type of Service) value. You can specify a TOS value from 0 – 255.
<*tos-mask*>	Mask value used for the TOS byte. You can specify a mask value from 0– 255. Default is **30**. Specify **any** for any TOS value.
log	This optional parameter allows you to enable ACL logging for this specific ACL rule.

Restrictions

When you apply an ACL to an interface, the X-Pedition appends an *implicit deny rule* to that ACL. The implicit deny rule denies ***all*** traffic. If you intend to allow all traffic that doesn't match your specified ACL rules to go through, you must *explicitly* define a rule to permit all traffic.

Examples

To create an ACL to permit IP traffic from the subnet 10.1.0.0 (with a 16 bit netmask) to any destination:

```
xp(config)# acl 100 permit ip 10.1.0.0/16 any
```

The following command creates an ACL to deny any incoming TCP or UDP traffic coming from a privileged port (less than 1024). If the incoming traffic is not TCP or UDP, then the X-Pedition check only the source and destination addresses, not the port number. Therefore, this ACL will deny all non-TCP and non-UDP traffic.

```
xp(config)# acl 120 deny ip any any 1-1024 any
```

To create an ACL to permit Telnet traffic (port 23) from the host 10.23.4.8 to the subnet 10.2.3.0:

```
xp(config)# acl 130 permit ip 10.23.4.8 10.2.3.0/24
```

The following command creates an ACL to permit all IP traffic. Since none of the ACL fields are specified, they are all assumed to be wildcards.

```
xp(config)# acl allip permit ip
```

The above command is equivalent to the following:

```
xp(config)# acl allip permit ip any any any any
```

acl permit|deny ip-protocol

Purpose

Create an ACL for any IP protocol type.

Format

acl <*name***> permit|deny ip-protocol** <*proto-num*> <*SrcAddr/Mask*> <*DstAddr/Mask*> <*tos*> [**log**]

Mode

Configure

Description

The **acl permit ip-protocol** and **acl deny ip-protocol** commands define an Access Control List to allow or block IP traffic from entering or leaving the router for any protocol type. Unlike the more specific variants of the acl commands such as **ip**, **tcp** and **udp**, the **ip-protocol** version of the command allows the user to specify any valid IP protocol type. This command allows the user to specify an IP protocol other than the ones available with other **acl permit|deny** commands. For example, to specify an ACL for IP encapsulation in IP, one can use the IPinIP protocol type, 4, in the ACL. For each parameter describing a flow, you can specify a value or use the keyword **any** to indicate a *wildcard* ("don't care") condition. When you specify only some of the parameters, the remaining fields will require the **any** keyword. If you do not specify any value for any field, the X-Pedition applies a wildcard condition to every field, giving the same effect as if you specify the **any** keyword.

Parameters

<*name*>	Name of this ACL. You can use a string of characters or a number.
<*proto-num*>	IP protocol number of this flow.
<*SrcAddr/Mask*>	The source address and the filtering mask of this flow. If the source address is a network or subnet address, you must supply the filtering mask. Generally, the filtering mask is the network mask of this network or subnet. If the source address is that of a host then no mask is required. By default, if a mask is not supplied, the source address is treated as that of a host. You can specify the mask using the traditional IP address format ("255.255.0.0") or the CIDR format ("/16").
<*DstAddr/Mask*>	The destination address and the filtering mask of this flow. The same requirements and restrictions for <*SrcAddr/Mask*> apply to <*DstAddr/Mask*>.
<*tos*>	IP TOS (Type of Service) value. You can specify a TOS from 0 – 255.

log This optional parameter allows you to enable ACL logging for this specific ACL rule.

Restrictions

When you apply an ACL to an interface, the X-Pedition appends an *implicit deny rule* to that ACL. The implicit deny rule denies ***all*** traffic. If you intend to allow all traffic that doesn't match your specified ACL rules to go through, you must *explicitly* define a rule to permit all traffic.

Examples

To create an ACL to permit VRRP traffic (IP protocol type 112) from the subnet 10.14.0.0 (with a 16 bit netmask) to any destination:

```
xp(config)# acl 100 permit ip-protocol 112 10.14.0.0/16 any
```

The following command has the same function as **acl 120 deny igmp** since the protocol type for IGMP is 2.

```
xp(config)# acl 120 deny ip-protocol 2
```

acl permit|deny ipx

Purpose

Create an IPX ACL.

Format

acl <*name*> **permit|deny ipx** <*SrcAddr*> <*SrcSocket*> <*DstAddr*> <*DstSocket*> <*SrcNetMask*> <*DstNetMask*>

Mode

Configure

Description

The **acl permit ipx** and **acl deny ipx** commands define an ACL to allow or block IPX traffic from entering or leaving the X-Pedition.

Parameters

<*name*>	Name of this ACL. You can use a string of characters or a number.
<*SrcAddr*>	The source IPX address in <*network*>**.**<*node*> format, where <*network*> *is the network address and* <*node*> is the MAC address. The X-Pedition will interpret this number in hexadecimal format. You do not need to use a "0x" prefix. You can use the keyword **any** to specify a wildcard ("don't care") condition. To specify any network, enter **FFFFFFFF**.<*node*>; to specify any node, enter <*network*>.**FF:FF:FF:FF:FF:FF**.
<*SrcSocket*>	Source IPX socket. The X-Pedition will interpret this number in hexadecimal format. You do not need to use a "0x" prefix. You can use the keyword **any** to specify a wildcard ("don't care") condition.
<*DstAddr*>	The destination IPX address in <*network*>**.**<*node*> format. The syntax for the destination address is the same as the syntax for the source address <*SrcAddr*>. The X-Pedition will interpret this number in hexadecimal format. You do not need to use a "0x" prefix. You can use the keyword **any** to specify a wildcard ("don't care") condition.
<*DstSocket*>	Destination IPX socket. The X-Pedition will interpret this number in hexadecimal format. You do not need to use a "0x" prefix. You can use the keyword **any** to specify a wildcard ("don't care") condition.

<*SrcNetmask*> Source network mask. This field specifies a group of networks for which the ACL applies. This mask field is ANDed with the network portion of <*SrcAddr*> and the source network of the incoming packets to determine a hit. The X-Pedition will interpret this number in hexadecimal format. You do not need to use a "0x" prefix.

This is an optional argument and if you omit the argument, the X-Pedition uses the hexadecimal value FFFFFFFF.

<*DstNetmask*> Destination network mask. This field specifies a group of networks for which the ACL applies. This mask field is ANDed with the network portion of <*DstAddr*> and the destination network of the incoming packets to determine a hit. The X-Pedition will interpret this number in hexadecimal format. You do not need to use a "0x" prefix.

This is an optional argument and if you omit the argument, the X-Pedition uses the hexadecimal value FFFFFFFF.

Restrictions

When you apply an ACL to an interface, the X-Pedition appends an *implicit deny rule* to that ACL. The implicit deny rule denies ***all*** traffic. If you intend to allow all traffic that doesn't match your specified ACL rules to go through, you must *explicitly* define a rule to permit all traffic.

Examples

The following command creates an ACL to permit IPX traffic from the host with IPX address AAAAAAAA.01:20:0A:F3:24:6D, any socket, to any other IPX address (network.node), any socket.

```
xp(config)# acl 100 permit ipx AAAAAAAA.01:20:0A:F3:24:6D any any any
```

The following command creates an ACL to deny IPX traffic from the host with IPX address F6D5E4.01:20:0A:F3:24:6D, with socket address 451, to any other IPX address (network.node), any socket.

```
xp(config)# acl 200 deny ipx F6D5E4.01:20:0A:F3:24:6D 451 any any
```

acl permit|deny ipxgns

Purpose

Create an IPX GNS (Get Nearest Server) ACL.

Format

acl <*name*> **permit|deny ipxgns** <*ServerAddr*> <*ServiceType*> <*ServiceName*>

Mode

Configure

Description

The **acl permit ipxgns** and **acl deny ipxgns** commands define an ACL to allow or block replying to GNS requests.

Parameters

<*name*>	Name of this ACL. You can use a string of characters or a number.
<*ServerAddr*>	The SAP server's IPX address in <*network*>**.**<*node*> format, where <*network*> *is the network address and* <*node*> is the MAC address. You can use the keyword **any** to specify a wildcard ("don't care") condition.
<*ServiceType*>	The SAP service type. Express the service type in hexadecimal. You do not need to use a "0x" prefix. You can use the keyword **any** to specify a wildcard ("don't care") condition.
<*ServiceName*>	The SAP service name. This is an optional argument and if you omit the argument, the X-Pedition applies a wildcard condition to the field.

Restrictions

When you apply an ACL to an interface, the X-Pedition appends an *implicit deny rule* to that ACL. The implicit deny rule denies ***all*** traffic. If you intend to allow all traffic that doesn't match your specified ACL rules to go through, you must *explicitly* define a rule to permit all traffic. You can only apply the **acl permit ipxgns** and **acl deny ipxgns** commands to output.

Examples

To create a GNS ACL to permit the X-Pedition to reply with the server “FILESERVER”, whose IPX address is F6D5E4.01:20:0A:F3:24:5D, to get nearest server requests:

```
xp(config)# acl 100 permit ipxgns F6D5E4.01:20:0A:F3:24:5D 0004 FILESERVER
```

To create a GNS ACL to prevent the X-Pedition from replying with the server “ARCHIVESERVER”, whose IPX address is F6D5E4.01:20:0A:F3:24:5C, to a get nearest server request:

```
xp(config)# acl 200 deny ipxgns F6D5E4.01:20:0A:F3:24:5C 0009 ARCHIVESERVER
```

acl permit|deny ipxrip

Purpose

Create an IPX RIP (Route Information Protocol) ACL.

Format

acl <*name*> **permit|deny ipxrip** <*FromNetwork*> <*ToNetwork*>

Mode

Configure

Description

The **acl permit ipxrip** and **acl deny ipxrip** commands define an ACL which allows or blocks IPX RIP traffic from entering or leaving the X-Pedition.

Parameters

<*name*>	Name of this ACL. You can use a string of characters or a number.
<*FromNetwork*>	The "from" IPX network address. You can use the **any** keyword to specify a wildcard condition. If you use **any**, the X-Pedition uses the value 0 for <*FromNetwork*> *and FFFFFFFE for* <*ToNetwork*>.
<*ToNetwork*>	The "to" IPX network address. This is an optional parameter. If you omit this parameter, the value that the X-Pedition assumes depends on whether you specified **any** for <*FromNetwork*>. –If you omit the <*ToNetwork*> value and you used the value **any** for <*FromNetwork*>, the X-Pedition sets the <*ToNetwork*> to FFFFFFFE. –If you omit the <*ToNetwork*> value but do not use the value **any** for <*FromNetwork*>, the X-Pedition sets <*ToNetwork*> to the same value you specified for <*FromNetwork*>.

Restrictions

Please note that the rules within an ACL must belong to the same protocol family.

When you apply an ACL to an interface, the X-Pedition appends an *implicit deny rule* to that ACL. The implicit deny rule denies ***all*** traffic. If you intend to allow all traffic that doesn't match your specified ACL rules to go through, you must *explicitly* define a rule to permit all traffic.

Examples

To create an ACL to permit IPX RIP traffic from networks AA000001 to AAFFFFFF:

```
xp(config)# acl 100 permit ipxrip AA000001 AAFFFFFF
```

acl permit|deny ipxsap

Purpose

Create an IPX SAP (Service Advertisement Protocol) ACL.

Format

acl <*name*> **permit|deny ipxsap** <*ServerAddr*> <*ServiceType*> <*ServiceName*>

Mode

Configure

Description

The **acl permit ipxsap** and **acl deny ipxsap** commands define an ACL to allow or block IPX SAP traffic from entering or leaving the X-Pedition.

Parameters

<*name*>	Name of this ACL. You can use a string of characters or a number.
<*ServerAddr*>	The SAP server's IPX address in <*network*>**.**<*node*> format, where <*network*> *is the network address and* <*node*> is the MAC address. You can use the keyword **any** to specify a wildcard ("don't care") condition. To specify any network, enter **FFFFFFFF**.<*node*>; to specify any node, enter <*network*>.**FF:FF:FF:FF:FF:FF**.
<*ServiceType*>	The SAP service type. Express the service type in hexadecimal. You do not need to use a "0x" prefix. You can use the keyword **any** to specify a wildcard ("don't care") condition.
<*ServiceName*>	The SAP service name. This is an optional argument and if you omit the argument, the X-Pedition applies a wildcard condition to the field.

Restrictions

Please note that the rules within an ACL must belong to the same protocol family.

When you apply an ACL to an interface, the X-Pedition appends an *implicit deny rule* to that ACL. The implicit deny rule denies ***all*** traffic. If you intend to allow all traffic that doesn't match your specified ACL rules to go through, you must *explicitly* define a rule to permit all traffic.

Examples

To create a SAP ACL to permit SAP information related to the server “FILESERVER” whose IPX address is F6D5E4.01:20:0A:F3:24:5D:

```
xp(config)# acl 100 permit ipxsap F6D5E4.01:20:0A:F3:24:5D 0004 FILESERVER
```

To create a SAP ACL to deny SAP information related to the server “ARCHIVESERVER” whose IPX address is F6D5E4.01:20:0A:F3:24:5C:

```
xp(config)# acl 200 deny ipxsap F6D5E4.01:20:0A:F3:24:5C 0009 ARCHIVESERVER
```

acl permit|deny ipxtype20

Purpose

Create an IPX type 20 ACL.

Format

acl <*name*> **permit|deny ipxtype20**

Mode

Configure

Description

The **acl permit ipxtype20** and **acl deny ipxtype20** commands define an ACL to allow or block IPX type 20 packets from entering or leaving the X-Pedition.

Parameters

<*name*>	Name of this ACL. You may use a string of characters or a number.

Restrictions

Please note that the rules within an ACL must belong to the same protocol family.

When you apply an ACL to an interface, the X-Pedition appends an *implicit deny rule* to that ACL. The implicit deny rule denies ***all*** traffic. If you intend to allow all traffic that doesn't match your specified ACL rules to go through, you must *explicitly* define a rule to permit all traffic.

Examples

To create an ACL to deny IPX type 20 packets:

```
xp(config)# acl 100 deny ipxtype20
```

acl permit|deny tcp

Purpose

Create a TCP ACL.

Format

acl <*name*> **permit|deny tcp** <*SrcAddr/Mask*> <*DstAddr/Mask*> <*SrcPort*> <*DstPort*> <*tos*> <*tos-mask*> [**accounting 5-minutes|15-minutes|hourly**] [**established**]**[log]**

Mode

Configure

Description

The **acl permit tcp** and **acl deny tcp** commands define an ACL to allow or block TCP traffic from entering or leaving the X-Pedition. For each parameter describing a flow, you can specify a value or use the keyword **any** to indicate a *wildcard* ("don't care") condition. When you specify only some of the parameters, the remaining fields will require the **any** keyword. If you do not specify any value for any field, the X-Pedition applies a wildcard condition to every field, giving the same effect as if you specify the **any** keyword.

The two exceptions to this rule are the optional parameters <*tos*> (type of service) and **accounting**. <*tos*> is a value from 0 to 255. The **accounting** keyword is only valid for the **permit** command, and can be placed anywhere on the command line.

Parameters

<*name*>	Is the name of this ACL. You can use a string of characters or a number.
<*SrcAddr/Mask*>	Is the source address and the filtering mask of this flow. If the source address is a network or subnet address, you must supply the filtering mask. Generally, the filtering mask is the network mask of this network or subnet. If the source address is that of a host then no mask is required. By default, if a mask is not supplied, the source address is treated as that of a host. You can specify the mask using the traditional IP address format ("255.255.0.0") or the CIDR format ("/16").
<*DstAddr/Mask*>	Is the destination address and the filtering mask of this flow. The same requirements and restrictions for <*SrcAddr/Mask*> apply to <*DstAddr/Mask*>.
<*SrcPort*>	For TCP or UDP, is the number of the source TCP or UDP port. *This field applies only to incoming TCP or UDP traffic.* You can specify a range of port numbers using operator symbols; for example, 10-20 (between 10 and 20 inclusive), >1024 (greater than 1024), <1024 (less than 1024), !=1024 (not

	equal to 1024). The port numbers of some popular services are already defined as keywords. For example, for Telnet, you can enter the port number 23 as well as the keyword **telnet**.
<*DstPort*>	For TCP or UDP, is the number of the destination TCP or UDP port. *This field applies only to incoming TCP or UDP traffic.* The same requirements and restrictions for <*SrcPort*> apply to <*DstPort*>.
<*tos*>	Is the IP TOS (Type of Service) value. You can specify a TOS value from 0 – 255.
<*tos-mask*>	Mask value used for the TOS byte. You can specify a mask value from 0– 255. Default is **30**. Specify **any** for any TOS value.
established	Allows TCP responses from external hosts, provided the connection was established internally.
log	This optional parameter allows you to enable ACL logging for this specific ACL rule.

Restrictions

When you apply an ACL to an interface, the X-Pedition appends an *implicit deny rule* to that ACL. The implicit deny rule denies ***all*** traffic. If you intend to allow all traffic that doesn't match your specified ACL rules to go through, you must *explicitly* define a rule to permit all traffic.

Examples

To create an ACL to permit TCP traffic from the subnet 10.21.33.0 (with a 24 bit netmask) to any destination:

```
xp(config)# acl 100 permit tcp 10.21.33.0/255.255.255.0 any
```

To create an ACL to deny any incoming HTTP traffic:

```
xp(config)# acl noweb deny tcp any any http any
```

To create an ACL to permit FTP traffic (both command and data ports) from subnet 10.31.34.0 to 10.31.60.0:

```
xp(config)# acl ftp100 permit tcp 10.31.34.0/24 10.31.60.0/24 20-21 any
```

acl permit|deny udp

Purpose

Create a UDP ACL.

Format

acl <*name*> **permit|deny udp** <*SrcAddr/Mask*> <*DstAddr/Mask*> <*SrcPort*> <*DstPort*> <*tos*> <*tos-mask*> [**accounting 5-minutes|15-minutes|hourly**] **[log]**

Mode

Configure

Description

The **acl permit udp** and **acl deny udp** commands define an ACL to allow or block UDP traffic from entering or leaving the X-Pedition. For each parameter describing a flow, you can specify a value or use the keyword **any** to indicate a *wildcard* ("don't care") condition. When you specify only some of the parameters, the remaining fields will require the **any** keyword. If you do not specify any value for any field, the X-Pedition applies a wildcard condition to every field, giving the same effect as if you specify the **any** keyword.

The two exceptions to this rule are the optional parameters <*tos*> (type of service) and **accounting**. <*tos*> is a value from 0 to 255. The **accounting** keyword is only valid for the **permit** command, and can be placed anywhere on the command line.

Parameters

<*name*>	Name of this ACL. You can use a string of characters or a number.
<*SrcAddr/Mask*>	The source address and the filtering mask of this flow. If the source address is a network or subnet address, you must supply the filtering mask. Generally, the filtering mask is the network mask of this network or subnet. If the source address is that of a host then no mask is required. By default, if a mask is not supplied, the source address is treated as that of a host. You can specify the mask using the traditional IP address format ("255.255.0.0") or the CIDR format ("/16").
<*DstAddr/Mask*>	The destination address and the filtering mask of this flow. The same requirements and restrictions for <*SrcAddr/Mask*> apply to <*DstAddr/Mask*>.
<*SrcPort*>	For TCP or UDP, the number of the source TCP or UDP port. *This field applies only to incoming TCP or UDP traffic.* You can specify a range of port numbers using operator symbols; for example, 10-20 (between 10 and 20 inclusive), >1024 (greater than 1024), <1024 (less than 1024), !=1024 (not equal to 1024).

	The port numbers of some popular services are already defined as keywords. For example, for Telnet, you can enter the port number 23 as well as the keyword **telnet**.
<*DstPort*>	For TCP or UDP, the number of the destination TCP or UDP port. *This field applies only to incoming TCP or UDP traffic.* The same requirements and restrictions for <*SrcPort*> apply to <*DstPort*>.
<*tos*>	IP TOS (Type of Service) value. You can specify a TOS value from 0 – 255.
<*tos-mask*>	Mask value used for the TOS byte. You can specify a mask value from 0– 255. Default is **30**. Specify **any** for any TOS value.
log	This optional parameter allows you to enable ACL logging for this specific ACL rule.

Restrictions

When you apply an ACL to an interface, the X-Pedition appends an *implicit deny rule* to that ACL. The implicit deny rule denies ***all*** traffic. If you intend to allow all traffic that doesn't match your specified ACL rules to go through, you must *explicitly* define a rule to permit all traffic.

Examples

Here are some examples of ACL commands for permitting and denying UDP traffic flows.

```
xp(config)# acl 100 permit udp 10.1.3.0/24 any
```

Creates an ACL to permit UDP traffic from the subnet 10.1.3.0 (with a 24 bit netmask) to any destination.

```
xp(config)# acl notftp deny udp any any tftp any
```

Creates an ACL to deny any incoming TFTP traffic.

```
xp(config)# acl udpnfs permit udp 10.12.0.0/16 10.7.0.0/16 any nfs
```

Creates an ACL to permit UDP based NFS traffic from subnet 10.12.0.0 to subnet 10.7.0.0.

acl-policy enable external

Purpose

Allow an external server to create and delete ACLs.

Format

acl-policy enable external| policy-routing-external

Mode

Configure

Description

The **acl-policy enable** command allows ACLs to be configured by an external agent, such as the Policy Manager. If this command is in the active configuration, an external server can create, modify, and delete ACLs on the X-Pedition. If this command is not in the active configuration, then ACLs can only be created, modified, and deleted using the CLI.

Parameters

external Enables ACLs to be configured by an external agent such as the Policy Manager.

policy-routing-external

Enables policy routing to be configured by an external agent such as the Policy Manager.

Restrictions

The only action allowed by the **acl-policy enable external** command is to allow an external server to create, modify, and delete ACLs. Once entered, this command must be negated in order to prohibit an external server from creating, altering, or deleting ACLs. An external server can only modify ACLs that it created, or ACLs that were created using the CLI with the “external” flag. It cannot modify an ACL that was created using the CLI with the “local” flag.

acl show

Purpose

Displays one or more ACLs.

Format

acl show [aclname <*string*>|**all] | [interface** <*string*>|**all-ip] | [service] | [port** <*port list*>|**all-ports] | [all]**

Mode

Enable

Description

The **acl show** command allows you to display the ACLs currently configured. Using the parameters associated with this command allows you to sort and display the ACLs by the name, interface, port, or service type.

You may also use the **acl show** command to display rules for Appletalk/ARE ACLs. Under this command, the Appletalk ACL is displayed in three separate sections. Each section provides information on configured zone, cable-range, and NBP rules, respectively. An *implicit deny* or *permit* rule is appended to the end of the ACL in all three sections. The third section, displaying Name Binding Protocol (NBP) rules, contains six fields:

1. **Forward**: Displays all permit and deny traffic.
2. **Count**: The number of times the ACL has been used to permit or deny traffic.
3. **Object**, **Type**, and **Zone**: The requested object, type, and zone names to be filtered. These parameters display as **object: type@zone**.
4. **Packet Type**: The type of NBP packet currently being filtered (broadcast request or forward request).

Parameters

aclname Use this parameter to display ACLs by name.

- <*string*> The name of the ACL.
- **all** Specify **all** to display all ACLs.

interface Use this parameter to display ACLs attached to a specific interface.

<string>	The name of the interface.
all-ip	Specify **all** to display ACLs attached to all IP interfaces.
service	Use this parameter to display ACLs applied to services.
port	Use this parameter to display ACLs applied to a specified port(s).
<port list>	The list of port(s) or SmartTRUNK(s).
all-ports	Specify **all** to display ACLs applied to all ports.
all	Use this parameter to display all ACLs.

Restrictions

You may sort Appletalk/ARE ACLs by **aclname** and **interface** only.

Chapter 2

acl-edit Commands

The **acl-edit** command activates the ACL Editor mode. The ACL Editor provides a user-friendly interface for maintaining and manipulating rules in an ACL. Using the editor, you can add, delete or re-order ACL rules. In addition, if the modified ACL is currently applied to an interface, the ACL is automatically "re-applied" to the interface and takes effect immediately. To edit an ACL, you enter the **acl-edit** command in Configure mode. The command must also specify the name of the ACL you want to edit. Only one ACL can be edited at one time.

Note: You may also use the ACL Editor to maintain and manipulate Appletalk/ARE (Advanced Routing Engine) ACL rules. In order to do this, however, you must be in **ARE-Configure** mode. For more information on this mode, please see Chapter 6, *are Commands*.

Command Summary

Table 2 lists the commands available with the ACL Editor. The sections following the table describe the command syntax.

Table 2. acl-edit commands

acl-edit <*aclname*>
acl permit|deny
delete <*rule#*>
exit
move <*rule#*> **after** <*rule#*>
save
show

acl-edit

Purpose

Enter ACL Editor to edit the specified ACL.

Format

acl-edit <*aclname*>

Mode

Configure

Description

The **acl-edit** command enters the ACL Editor to edit an ACL specified by the user. Once inside the ACL editor, the user can then add, delete or re-order ACL rules for that ACL. If the ACL happens to be applied to an interface, changes made to that ACL will automatically take effect when the changes are committed to the running system.

Parameters

<*aclname*> Name of the ACL to edit.

Restrictions

Inside the ACL Editor, you can only add rules for the ACL you specified in the acl-edit command. You cannot add rules for other ACLs. Basically, each ACL editing session works only on one ACL at a time. For example, if you start with *acl-edit 110*, you cannot add rules for ACL *121*.

Example

To edit ACL 111:

```
xp(config)# acl-edit 111
xp(acl-edit)> ?
 acl             - Configure L3 Access Control List
 delete          - Delete an ACL rule
 exit            - Exit current mode
 help            - Describe online help facility
 move            - Move an ACL rule
 save            - Save changes made to this ACL
 show            - Show contents of this ACL
xp(acl-edit)>
```

acl permit| deny

Purpose

Create an ACL rule to permit or deny traffic.

Format

acl <*name*> **permit|deny**

Mode

ACL Editor

Description

The **acl permit| deny** commands are equivalent to the same commands in the Configuration mode. You can use these commands to create rules for the ACL that you are editing. Just like the acl commands in Configuration mode, new rules are appended to the end of the rules. You can use the move command to re-order the rules.

Restrictions

You can only add rules for the ACL you specified in the acl-edit command. You cannot add rules for other ACLs. For example, if you start with *acl-edit 110*, you cannot add rules for ACL *121*.

Example

To add a new rule (deny all UDP traffic) to ACL 111:

```
xp(config)# acl-edit 111
xp(acl-edit)> show
 1*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2000-2002 any
 2*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2003-2005 any
xp(acl-edit)> acl 111 deny udp
xp(acl-edit)> show
 1*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2000-2002 any
 2*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2003-2005 any
 3*: acl 111 deny udp
xp(acl-edit)>
```

delete

Purpose

Deletes a rule from an ACL.

Format

delete <*rule#*>

Mode

ACL Editor

Description

The **delete** commands allows the administrator to delete a specific rule from an ACL. When in the ACL Editor, each rule is displayed with its rule number. One can delete a specific rule from an ACL by specifying its rule number with the delete command.

Parameters

<*rule#*> Number of the ACL rule to delete.

Restrictions

None

Example

To delete ACL rule number 2 from the ACL:

```
xp(config)# acl-edit 111
xp(acl-edit)> show
 1*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2000-2002 any
 2*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2003-2005 any
 3*: acl 111 deny udp
xp(acl-edit)> delete 2
xp(acl-edit)> show
 1*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2000-2002 any
 2*: acl 111 deny udp
xp(acl-edit)>
```

exit

Purpose

Exit ACL Editor.

Format

exit

Mode

ACL Editor

Description

The **exit** command allows the user to exit the ACL Editor. Before exiting, if changes are made to this ACL, the system will prompt the user to see if the changes should be committed to the running system or discarded. If the user commits the changes then changes made to this ACL will take effect immediately. If the ACL is applied to an interface, the ACL is automatically re-applied to the interface. Packets going through this interface will be matched against the new rules in this ACL. If the user chooses not to commit the changes, the changes will be discarded. The next time the user edits this ACL, changes from the previous edit session will be lost.

Parameters

None

Restrictions

None

Example

To exit the ACL editor:

```
xp(config)# acl-edit 111
xp(acl-edit)> show
 1*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2000-2002 any
 2*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2003-2005 any
 3*: acl 111 deny udp
xp(acl-edit)> delete 3
xp(acl-edit)> show
 1*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2000-2002 any
 2*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2003-2005 any
xp(acl-edit)> exit
Do you want to commit your ACL changes (yes: commit, no: discard) [yes]? no
xp(config)#
```

move

Purpose

Re-order ACL rules by moving a rule to another position.

Format

move <*src-rule#*> **after** <*dst-rule#*>

Mode

ACL Editor

Description

The **move** command provides the user with the ability to re-order rules within an ACL. When new rules are entered in the ACL Editor, they are appended to the end of the rules. One can move these rules to the desired location by using the move command. The move command can also be used on existing ACL rules created in Configuration mode instead of the ACL Editor.

Parameters

<*src-rule#*> Rule number of the rule you want to move.

<*dst-rule#*> Rule number of the rule after which you want the source rule to move to.

Restrictions

None.

Examples

To move rule #2 to the end of the list:

```
xp(config)# acl-edit 111
xp(acl-edit)> show
 1*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2000-2002 any
 2*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2003-2005 any
 3*: acl 111 deny udp
xp(acl-edit)> move 2 after 3
xp(acl-edit)> show
 1*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2000-2002 any
 2*: acl 111 deny udp
 3*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2003-2005 any
xp(acl-edit)>
```

save

Purpose

Save any changes made by the ACL Editor.

Format

save

Mode

ACL Editor

Description

The **save** command saves any non-committed changes made by the ACL Editor. If changes are made to this ACL, the changes will be saved and will take effect immediately. If the ACL is applied to an interface, the ACL is automatically re-applied to the interface. Packets going through this interface will be matched against the new rules in this ACL. The **save** command also contains an implicit exit command. Regardless of whether changes were made by the ACL Editor or not, upon completion of the **save** command, the user exits the ACL Editor and returns to Configuration mode. Consequently, one should issue the **save** command after all the changes are made.

Parameters

None

Restrictions

None

Examples

To save and commit the changes made by the ACL Editor.

```
xp(config)# acl-edit 111
xp(acl-edit)> show
 1*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2000-2002 any
 2*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2003-2005 any
 3*: acl 111 deny udp
xp(acl-edit)> delete 2
xp(acl-edit)> show
 1*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2000-2002 any
 2*: acl 111 deny udp
xp(acl-edit)> save
2003-04-29 14:38:33 %ACL-A-MODIFIED, ACL (111) modified.
xp(config)#
```

show

Purpose

Displays the contents of the ACL in the current editing session.

Format

show

Mode

ACL Editor

Description

The **show** command displays the contents of the ACL currently being edited.

Parameters

None

Restrictions

None

Examples

To display the contents of the ACL currently being edited:

```
xp(config)# acl-edit 111
xp(acl-edit)> show
 1*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2000-2002 any
 2*: acl 111 permit tcp 10.1.15.0/24 10.1.11.0/24 2003-2005 any
 3*: acl 111 deny udp
xp(acl-edit)>
```

Chapter 3

aging Commands

The **aging** commands control aging of learned MAC address entries in the X-Pedition's Layer-2 lookup tables or Layer-3 and Layer-4 flows. Using the **aging** commands, you can show Layer-2 or Layer-3 and Layer-4 aging information, set or disable Layer-2 aging on specific ports, set or disable aging of Layer-3 and Layer-4 flows, or set or disable NAT or LSNAT flows.

Note: Interfaces configured with PVCs do not support LSNAT.

Command Summary

Table 3 lists the **l2** and **l3** aging commands. The sections following the table describe the command syntax.

Table 3. aging commands

aging l2 disable <*port-list*>**|all-ports**
aging l2 set aging-timeout <*seconds*> **port** <*port-list*>**|all-ports**
aging l2 show status
aging l3 set timeout <*seconds*>**|disable**
aging l3 set nat-flow-timeout <*minutes*>**|disable**
aging l3 show status

aging l2 disable

Purpose

Disable aging of MAC addresses.

Format

aging l2 disable <*port-list*>|**all-ports**

Mode

Configure

Description

By default, the X-Pedition ages learned MAC addresses in the Layer-2 lookup tables. Each port has its own Layer-2 lookup table. When a learned entry ages out, the X-Pedition removes the aged out entry. You can disable this behavior by disabling aging on all ports or on specific ports.

Parameters

<*port-list*>|**all-ports**
: The port(s) on which you want to disable aging. You can specify a single port or a comma-separated list of ports. If you use the **all-ports** keyword, aging is disabled on all ports.

Restrictions

Unknown.

Examples

To disable aging on slot 1, port 3:

```
xp(config)# aging l2 disable et.1.3
```

To disable aging on slot 4, port 2, and slots 1 through 3, ports 4, 6, 7, and 8:

```
xp(config)# aging l2 disable et.4.2,et.(1-3).(4,6-8)
```

To disable aging on all ports:

```
xp(config)# aging l2 disable all-ports
```

aging l2 set aging-timeout

Purpose

Set the aging time for learned MAC entries.

Format

aging l2 set <*port-list*>|**all-ports aging-timeout** <*seconds*>

Mode

Configure

Description

The **aging l2 set aging-timeout** command sets the aging time for learned MAC entries. When the aging time expires for a MAC address, the X-Pedition removes the MAC address from the specified port(s). The aging time is specified in seconds.

Parameters

<*port-list*>|**all-ports**
: The port(s) on which you want to set the aging time. You can specify a single port or a comma-separated list of ports. If you use the **all-ports** keyword, the aging time is set on all ports.

<*seconds*>
: The number of seconds the X-Pedition allows a learned MAC address to remain in the L2 lookup table (for the specified port). You can specify from 15 to 1000000 seconds. The default is 300 seconds.

Restrictions

None.

Example

To set the aging time to 15 seconds on all ports:

```
xp(config)# aging l2 set all-ports aging-timeout 15
```

aging l2 show status

Purpose

Show the L2 aging status for X-Pedition ports.

Format

aging l2 show status

Mode

User

Description

The **aging l2 show status** command shows whether L2 aging is enabled or disabled on X-Pedition ports. For ports on which L2 aging is enabled, this command also shows the aging time.

Parameters

None.

Restrictions

None.

aging l3 set timeout

Purpose

Set the aging time for a Layer-3 or Layer-4 flow.

Format

aging l3 set timeout <*seconds*>|**disable**

Mode

Configure

Description

The **aging l3 set timeout** command sets the aging time for a Layer-3 or Layer-4 flow. The aging time is specified in seconds.

Parameters

<*seconds*> The number of seconds the X-Pedition allows for a Layer-3 or Layer-4 flow. You can specify a value from 4 to 3600 seconds. For example, in an ISP environment (where thousands of flows are possible), you could change this value to 180-300 (3-5 minutes) to help in keeping with longer-term flows. The default is 30 seconds.

disable Disables Layer-3 and Layer-4 aging.

Restrictions

None.

Example

To set the Layer-3 or Layer-4 flow aging time to 300 seconds (5 minutes):

```
xp(config)# aging l3 set timeout 60
```

aging l3 set nat-flow-timeout

Purpose

Set the aging time for NAT and LSNAT flows.

Format

aging l3 set nat-flow-timeout <*minutes*>|**disable**

Mode

Configure

Description

The **aging l3 set nat-flow-timeout** command sets the aging time for Network Address Translation (NAT) and Load Sharing NAT flows. The aging time is specified in minutes.

Parameters

<*minutes*> The number of minutes the X-Pedition allows for NAT and LSNAT flows. You can specify from 2 to 1440 minutes. The default is 2 minutes.

disable Disables NAT and LSNAT flow aging.

Restrictions

Interfaces configured with PVCs do not support LSNAT.

Example

To set the NAT aging time to 5 minutes:

```
xp(config)# aging l3 set nat-flow-timeout 5
```

aging l3 show status

Purpose

Show the L3 aging status for X-Pedition ports.

Format

aging l3 show status

Mode

User

Description

The **aging l3 show status** command shows whether Layer-3 or Layer-4 aging is enabled or disabled on X-Pedition ports. For ports on which Layer-3 or Layer-4 aging is enabled, this command also shows the aging time.

Parameters

None.

Restrictions

None.

Example

To show whether Layer-3 or Layer-4 aging is enabled and display the aging time for enabled ports:

```
xp# aging l3 show status
L3 Aging: Timeout 30 seconds
```

Chapter 4

appletalk Commands

The **appletalk** commands allow the user to manipulate the AppleTalk Protocol for an Advanced Routing Engine (ARE) module. Before using any of the commands in this chapter, you must first execute the command **are enable protocol appletalk module** <*module-number*> from the Configuration mode. For more information on the **are enable protocol appletalk** command, please see Chapter 6, *are Commands*.

Command Summary

Table 4 lists the **appletalk** commands. The sections following the table describe the command syntax.

Table 4. appletalk commands

appletalk aarp clear address <*Net.Node*>**|all**
appletalk aarp show address <*Net.Node*>**|all**
appletalk ping <*address*> **[packets** <*num*>**] [size** <*packet-size*>**] [wait** <*seconds*>**]**
appletalk qos internal-queue-priority <*priority*>
appletalk show aarp-globals
appletalk show interfaces <*InterfaceName*>**|all**
appletalk show routes interface <*InterfaceName*>**|all**
appletalk show rtmp-jitter|-status|-update-interval|-valid-interval
appletalk show zip-query-interval
appletalk show zone interface <*InterfaceName*>**|all**
appletalk show zone network <*range*>**|all**

appletalk aarp clear

Purpose

Removes the specified AppleTalk Address Resolution Protocol (AARP) entries.

Format

appletalk aarp clear address <*Net.Node*>|**all**

Mode

Enable

Description

The **appletalk aarp clear** command allows the user to remove specific AppleTalk AARP entries from the AppleTalk AARP tables. This command will not remove permanent AARP entries (such as those created with the **appletalk add aarp** command).

Parameters

address <*Net.Node*>	Specifies AppleTalk AARP entry to remove. Specifying **all** will remove all AppleTalk AARP entries.

Restrictions

None.

Example

To remove AppleTalk AARP entry 1.2:

```
xp# appletalk aarp clear address 1.2
```

appletalk aarp show

Purpose

Displays the specified AppleTalk Address Resolution Protocol (AARP) entries.

Format

appletalk aarp show address <*Net.Node*>|**all**

Mode

Enable

Parameters

address <*Net.Node*>	Specifies AppleTalk AARP entry to display. Specifying **all** will display all AppleTalk AARP entries.

Restrictions

None.

appletalk ping

Purpose

Tests connection for the specified AppleTalk address.

Format

appletalk ping <*address*> **[packets** <*num*>**] [size** <*packet-size*>**] [wait** <*seconds*>**]**

Mode

Enable

Description

The **appletalk ping** command allows the user to test the connection between the router and a specific AppleTalk address.

Parameters

<*address*>	Specifies AppleTalk address you want to ping.
packets <*num*>	Specifies total number of packets to send. The default is 1.
size <*packet-size*>	Specifies the size of each packet. This number must lie between 0 and 585. The default size is 20.
wait <*seconds*>	Specifies the number of seconds to wait for all ping responses to arrive. The default is 1.

Restrictions

None.

Examples

To ping AppleTalk address 1.2:

```
xp# appletalk ping 1.2
```

To ping the same AppleTalk address 5 times with packets 100 bytes in length, with a wait time of 3 seconds before displaying the ping response:

```
xp# appletalk ping 1.2 packets 5 size 100 wait 3
```

appletalk qos internal-queue-priority

Purpose

This command allows users to prioritize AppleTalk traffic.

Format

appletalk qos internal-queue-priority <*priority*>

Mode

ARE Configure

Description

Sets the internal queue priority for all forwarded AppleTalk traffic.

Parameter

<*priority*> Possible priorities are **low-priority**, **med-priority**, **high-priority**, and **control-priority**. The default priority level for AppleTalk traffic is low.

Restrictions

None.

Example

To set the priority for all forwarded AppleTalk traffic to medium, enter the following:

```
xp(config)# appletalk qos internal-queue-priority med-priority
```

appletalk show aarp-globals

Purpose

Displays all AppleTalk AARP settings.

Format

appletalk show aarp-globals

Mode

Enable

Parameters

None.

Restrictions

None.

appletalk show interfaces

Purpose

Displays AppleTalk interfaces defined on the system.

Format

appletalk show interfaces <*InterfaceName*>|**all**

Mode

Enable

Parameters

<*InterfaceName*>|**all**
Specifies an interface to display. Entering **all** will display all interfaces on the system.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

None.

appletalk show routes

Purpose

Displays AppleTalk routing table for system interface(s).

Format

appletalk show routes interface <*InterfaceName*>**|all**

Mode

Enable

Description

The **appletalk show routes** command shows the user the AppleTalk routing table for all interfaces, or a specified interface. If you choose the **all** parameter, the entire routing table will be shown. If you choose to display a specific interface, the commands will show the entire routing table *minus* any routes filtered for that interface.

Parameters

interface <*InterfaceName*>**|all**

Specifies an interface for which to display routing table. If a specific interface is entered, the routing table will not include routes which are filtered for that interface.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

None.

Example

To display the routing table for interface “app3”:

```
xp# appletalk show routes interface app3
```

appletalk show rtmp

Purpose

Displays various Routing Table Maintenance Protocol (RTMP) statistics.

Format

appletalk show rtmp-jitter|-status|-update-interval|-valid-interval

Mode

Enable

Description

The **appletalk show rtmp** command shows the user statistics for the RTMP, including functions defined by the **appletalk rtmp** command.

Parameters

jitter	Displays RTMP jitter statistics.
status	Displays the RTMP status for each interface. This status shows whether RTMP and split-horizon are enabled or disabled.
update-interval	Displays, in seconds, the currently set interval between the sending of RTMP updates.
valid-interval	Displays, in seconds, the currently set interval during which an RTMP route is considered valid.

Restrictions

None.

appletalk show zip-query-interval

Purpose

Displays the currently set number of seconds between Zone Information Protocol (ZIP) queries.

Format

appletalk show zip-query-interval

Mode

Enable

Description

The **appletalk show zip-query-interval** command shows the user the currently set interval between ZIP queries, as defined by the **appletalk zip query-interval** command.

Parameters

None.

Restrictions

None.

appletalk show zone interface

Purpose

Displays all zones for specified interface(s).

Format

appletalk show zone interface <*InterfaceName*>|**all**

Mode

Enable

Description

The **appletalk show zone interface** command shows the user all zones for specified interfaces on the router. Zones derived from other routers on the network will not be displayed.

Parameters

<*InterfaceName*>|**all**
Specifies the interface for which you would like to see zone statistics. Specifying **all** will display zones for all interfaces on the router.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

None.

Examples

To display zones for interface “app7”:

```
xp# appletalk show zone interface app7
```

To display zones for all interfaces:

```
xp# appletalk show zones interface all
```

appletalk show zone network

Purpose

Displays all zones for specified cable range(s).

Format

appletalk show zone network <*range*>**|all**

Mode

Enable

Description

The **appletalk show zone network** command shows the user all zones for specified cable ranges on the router. If the keyword **all** is specified, command will display all known zones on the entire network.

Parameters

<*range*>**|all**	Specifies the cable range for which you would like to see zone statistics. Valid numbers include 1 to 65279. Specifying **all** will display zones from the entire network.

Restrictions

None.

Examples

To display zones for range 10-100:

```
xp# appletalk show zone network 10-100
```

To display zones from the entire network:

```
xp# appletalk show zones network all
```

Chapter 5

appletalk Configuration Commands

The **appletalk** configuration commands allow the user to configure the AppleTalk Protocol for an Advanced Routing Engine (ARE) module. Before using any of the commands in this chapter, you must first execute the command **are enable protocol appletalk module** <*module-number*> from the Configuration mode.Then you must enter the ARE-Configuration mode. These commands can be used *only* from the ARE-Configuration mode. For more information on the **are enable protocol appletalk** command and ARE-Configuration mode, please see Chapter 6, *are Commands*.

Command Summary

Table 5 lists the **appletalk** configuration commands. The sections following the table describe the command syntax.

Table 5. appletalk configuration commands

appletalk add aarp exit-port <*port*> **address** <*Net.Node*> **macaddr** <*MACaddr*>
appletalk add route interface <*InterfaceName*> **cable-range** <*range*> **gateway** <*Net.Node*> **distance** <*hops*>
appletalk aarp interval <*seconds*>
appletalk aarp timeout <*seconds*>
appletalk checksun disable
appletalk rtmp jitter <*percent*>
appletalk rtmp update-disable interface <*InterfaceName*>**|all**
appletalk rtmp update-interval <*seconds*>

Table 5. appletalk configuration commands (Continued)

appletalk rtmp valid-interval <*seconds*>
appletalk split-horizon disable
appletalk zip query-interval <*seconds*>

appletalk add aarp

Purpose

Creates a permanent AppleTalk Address Resolution Protocol (AARP) entry.

Format

appletalk add aarp exit port <*port*> **address** <*Net.Node*> **macaddr** <*MACaddr*>

Mode

ARE-Configure

Description

The **appletalk add aarp** command allows the user to create a permanent AppleTalk AARP entry.

Parameters

exit-port <*port*>
Specifies port for which to send any packet destined for the following address.

address <*Net.Node*>
Specifies AppleTalk address to associate with the following MAC address.

macaddr <*MACaddr*>
Specifies MAC address to associate with the previous AppleTalk address. MAC address should be entered in the following format: xx:xx:xx:xx:xx:xx.

Restrictions

You must be in ARE-Configure mode before using this command.

appletalk add route

Purpose

Adds a static route to the routing table.

Format

appletalk add route interface <*InterfaceName*> **cable-range** <*range*> **gateway** <*Net.Node*> **distance** <*hops*>

Mode

ARE-Configure

Description

The **appletalk add route** command allows the user to add a static route to the routing table.

Parameters

interface <*InterfaceName*>
Specifies interface through which packets will be routed.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

cable-range <*range*>
Specifies range of network numbers which can be reached through this route. Valid numbers include 1 to 65279.

gateway <*Net.Node*>
Specifies the address of the next router a packet destined for the cable range will encounter.

distance <*hops*>
Specifies how many routers a packet must encounter before reaching its final destination.

Restrictions

You must be in ARE-Configure mode before using this command.

Example

To add a route on interface “if1”:

```
xp(are-config)# appletalk add route interface if1 cable-range 3-4 gateway 5.6 distance 2
```

This command determines that an AppleTalk packet destined for network “3-4” will exit through this interface. The AppleTalk address for the next router to be encountered is “5.6,” and there are “2” hops (routers) between this router and the destination.

appletalk aarp interval

Purpose

Sets the interval between AppleTalk AARP requests.

Format

appletalk aarp interval <*seconds*>

Mode

ARE-Configure

Description

The **appletalk aarp interval** command creates an interval between AARP requests while the system is attempting to determine a hardware address.

Parameters

<*seconds*> Specifies number of seconds at which you want to set the interval.

Restrictions

You must be in ARE-Configure mode before using this command.

appletalk aarp timeout

Purpose

Determines the age-out time of the AppleTalk AARP table.

Format

appletalk aarp timeout <*seconds*>

Mode

ARE-Configure

Parameters

<*seconds*> Specifies number of seconds at which you want to set the age-out time.

Restrictions

You must be in ARE-Configure mode before using this command.

appletalk checksum disable

Purpose

Disables checksum calculation for out-going packets.

Format

appletalk checksum disable

Mode

ARE-Configure

Parameters

None.

Restrictions

You must be in ARE-Configure mode before using this command.

appletalk rtmp jitter

Purpose

Staggers Routing Table Maintenance Protocol (RTMP) routing updates.

Format

appletalk rtmp jitter <*percent*>

Mode

ARE-Configure

Description

The **appletalk rtmp jitter** command staggers routing updates by altering the RTMP update interval in order to avoid syncing with other routers on the same link.

Parameters

<*percent*>	Specifies the percentage to alter the RTMP update interval. For example, if the current RTMP update interval is 10, and you set the "jitter" to 10%, the update interval will be altered to occur between 9 and 11 seconds.

Restrictions

You must be in ARE-Configure mode before using this command.

Example

To alter the RTMP update interval by 15%:

```
xp(are-config)# appletalk rtmp jitter 15
```

appletalk rtmp update-disable

Purpose

Disables RTMP updates.

Format

appletalk rtmp update-disable interface <*InterfaceName*>**|all**

Mode

ARE-Configure

Description

The **appletalk rtmp update-disable** command prevents RTMP updates from being sent out on the indicated interface(s).

Parameters

interface <*InterfaceName*>**|all**

Specifies interface for which you want to disable RTMP updates. You may specify **all** to disable updates on all available interfaces.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

You must be in ARE-Configure mode before using this command.

Example

To disable RTMP updates on interface “app5”:

```
xp(are-config)# appletalk rtmp update-disable interface app5
```

appletalk rtmp update-interval

Purpose

Determines the number of seconds between RTMP updates.

Format

appletalk rtmp update-interval <*seconds*>

Mode

ARE-Configure

Description

The **appletalk rtmp update-interval** command sets the number of seconds between RTMP updates on an interface.

Parameters

<*seconds*>	Specifies number of seconds at which to set RTMP update interval. The default is 10.

Restrictions

You must be in ARE-Configure mode before using this command.

appletalk rtmp valid-interval

Purpose

Determines number of seconds a route is considered valid.

Format

appletalk rtmp valid-interval <*seconds*>

Mode

ARE-Configure

Description

The **appletalk rtmp valid-interval** command sets the number of seconds for which a route is considered valid. A route is considered invalid after the valid-interval expires twice. A route is deleted after the valid-interval expires three times. The valid timer is reset every time an RTMP packet is received which validates the route.

Parameters

<*seconds*>	Specifies number of seconds for which you want to set the valid-interval timer. The default is 20.

Restrictions

You must be in ARE-Configure mode before using this command.

appletalk split-horizon disable

Purpose

Disables RTMP from using split-horizon methodology.

Format

appletalk split-horizon disable

Mode

ARE-Configure

Description

The **appletalk split-horizon disable** command prevents the Route Maintenance Protocol (RTMP) from using the split-horizon methodology. This methodology is enabled by default.

Parameters

None.

Restrictions

You must be in ARE-Configure mode before using this command.

appletalk zip query-interval

Purpose

Determines interval between Zone Information Protocol (ZIP) queries.

Format

appletalk zip query-interval <*seconds*>

Mode

ARE-Configure

Description

The **appletalk zip query-interval** command sets the interval between zip queries. These queries occur when the router discovers a routing table entry without an associated zone. It sends a zip query to collect appropriate zones for the entry.

Parameters

<*seconds*>	Specifies number of seconds for which you want to set the query interval timer. The default is 10.

Restrictions

You must be in ARE-Configure mode before using this command.

Chapter 6

are Commands

The **are** commands allow you to manipulate the Advanced Routing Engine (ARE) module, the full-featured AppleTalk Phase II router available for the X-Pedition. These commands apply to the ARE module globally, and will function regardless of protocol designation.

Command Summary

Table 6 lists the **are** commands. The sections following the table describe the command syntax.

Table 6. are commands

are enable protocol appletalk module <*module-number*>
are-config <*module-number*>
system are-promimage upgrade <*module-number*> <*tftp-server*> <*filename*>

are enable protocol appletalk

Purpose

Enables AppleTalk protocol on an ARE module.

Format

are enable protocol appletalk module <*module-number*>

Mode

Configure

Description

The **are enable protocol appletalk** command allows the user to enable AppleTalk protocol on a specified ARE module. This command is required before the user attempts to configure any protocol-specific commands on a module.

Parameters

module <*module-number*> Specifies ARE module for which to enable the protocol.

Restrictions

None.

Example

The following examples demonstrate how to enable AppleTalk protocol on module 5:

```
xp(config)# show
Running system configuration:
    !
    ! Last modified from Console on 2001-12-06 12:21:43
    !
 1 : vlan create blue appletalk id 100
 2 : vlan create green appletalk id 200
 3 : vlan create red appletalk id 300
 4 : vlan add ports et.1.1 to blue
 5 : vlan add ports et.1.2 to green
 6 : vlan add ports et.1.3 to red
    !
 7 : are enable protocol appletalk module 5
    !
 8 : interface create appletalk Apple20000 vlan blue noseed
 9 : interface create appletalk Apple21000 vlan green cable-range 21000-21010 zone Teachers address
21001.1
10 : interface create appletalk Apple22000 vlan red cable-range 22000-22010 zone Admins address
22001.1
11 : interface add appletalk Apple21000 zone Biology
12 : interface add appletalk Apple21000 zone Journalism
```

```
xp(config)# show

1 : vlan create blue appletalk id 100
2 : vlan create black appletalk id 400
3 : vlan create yellow appletalk id 500
4 : vlan add ports et.1.1 to blue
5 : vlan add ports et.1.2 to black
6 : vlan add ports et.1.3 to yellow
!
7 : are enable protocol appletalk module 5
!
8 : interface create appletalk Apple20000 vlan blue cable-range 20000-20010 zone Students address
20000.1
9 : interface create appletalk Apple30000 vlan black cable-range 30000-30010 zone Geology address
30000.1
10 :interface create appletalk Apple40000 vlan yellow cable-range 40000-40010 zone English address
40000.1
11 : interface add appletalk Apple30000 zone Chemistry
12 : interface add appletalk Apple30000 zone Physics
```

are-config

Purpose

Places CLI session in ARE-Configure mode.

Format

are config <*module-number*>

Mode

Configure

Description

The **are configure** command places the CLI session in ARE-Configure mode. All configuration settings for a specific ARE module or modules must be made from this mode.

Note: When you negate an interface configured with an ARP command, the X-Pedition automatically reassigns the command to a non-existing interface in the same configuration.

Parameters

<*module-number*>	Specifies ARE module for which to enter ARE-Configure mode. If no module is specified, interface will return a list of all active modules available for configuration.

Restrictions

User must be in Configure mode.

Example

To enter ARE-Configure mode on module 5:

```
xp(config)# are config 5
```

system are-promimage upgrade

Purpose

Upgrades boot PROM image on a specified ARE module.

Format

system are-promimage upgrade <*module-number*> <*tftp-server*> <*filename*>

Mode

Enable

Description

The **system are-promimage upgrade** command allows you to upgrade a specific ARE module's boot PROM image with the image located on the tftp server.

Note: In order to take advantage of this upgrade, you must reboot the ARE module.

Parameters

<*module-number*>	Specifies ARE module for which to upgrade Boot PROM image.
<tftp-server>	Specifies tftp server on which the image is located.
<filename>	Specifies image file name.

Restrictions

None.

Example

To upgrade the PROM for ARE module 5 with the file "prom_image_file":

```
xp(config)# system are-promimage upgrade 5 tftp://host1/public/prom_image_file
```

Chapter 7

arp Commands

The **arp** commands enable you to add, display, and clear ARP entries on the X-Pedition.

Command Summary

Table 7 lists the arp commands. The sections following the table describe the command syntax.

Table 7. arp commands

arp add <*host*> **mac-addr** <*MAC-addr*> **[vlan | exit-port** <*port*>**}] keep-time** <*seconds*>
arp clear <*host*>**|all [interface** <*string*>**| unresolved | all] [port <***port*>**]**
arp set drop-unresolved disabled| enabled
arp set interface <*name*>**|all keep-time** <*number*>
arp set max-unresolved <*num*>
arp set unresolve-threshold <*num*>
apr set unresolve-timer <*num*>
arp show <*IPaddr*>**|all [undecoded] [unresolved] [interface** <*string*>**| all] [port <***port*>**]**

arp add

Purpose

Add an ARP entry.

Format

arp add <*host*> **mac-addr** <*MAC-addr*> **[vlan | exit-port** <*port*>**}] keep-time** <*seconds*>

Mode

Enable and Configure

Description

The **arp add** command lets you manually add ARP entries to the ARP table. Typically, the X-Pedition creates ARP entries dynamically. Using the **arp add** command, you can create an ARP entry to last a specific amount of time or as a permanent ARP entry. This command exists in both Enable and Configure mode with a slight variation. The **keep-time** option is valid only in Enable mode and allows you to create an ARP entry that will last for a specific amount of time. The Configure mode version of the **arp add** command does not use the **keep-time** option and the ARP entries created will be permanent and will not have an expiration time.

If you specify an **exit port**, packets destined for the IP address will always transmit out the given exit port. If you specify the **vlan** option, the ARP entry will be associated to a VLAN rather than a specific exit port and traffic destined for the given IP address will always flood out of the entire VLAN/interface that provides a route to it. If you specify neither option, packets will transmit on all ports of the interface until the host receives an ARP request. The X-Pedition will then update the exit port with the port on which the ARP request was received, so that subsequent packets will transmit on only one port.

Parameters

<*host*>	Hostname or IP address of this ARP entry.
mac-addr <*MAC-addr*>	MAC address of the host.
vlan	Traffic to this host should be flooded out the VLAN/interface it belongs to.
exit-port <*port*>	The port for which you are adding the entry. Specify the port to which the host is connected.
keep-time <*seconds*>	The number of seconds this ARP entry should remain in the ARP table. A value of 0 means this is a permanent ARP entry.

Note: This option is valid only for the Enable mode **arp add** command.

Restrictions

- If you enter the **arp add** command while in the Configure mode, you can add only permanent ARP entries.
- The X-Pedition clears all expired and unresolved ARP entries once every 5 minutes. Therefore, expired ARPs may be kept up to 5 minutes longer than the keep-time.

Examples

To create an ARP entry for the IP address 10.8.1.2 at port et.4.7 for 15 seconds:

```
xp# arp add 10.8.1.2 mac-addr 08:00:20:a2:f3:49 exit-port et.4.7 keep-time 15
```

To create a permanent ARP entry for the host *nfs2* at port et.3.1:

```
xp(config)# arp add nfs2 mac-addr 080020:13a09f exit-port et.3.1
```

To create a permanent ARP entry for IP address 10.8.1.25 that will always flood the traffic out the subnet:

```
xp(config)# arp add 10.8.1.25 mac-addr 080020:a2f360 vlan
```

arp clear

Purpose

Remove an ARP entry from the ARP table.

Format

arp clear <*host*>**|all [interface** <*string*>**| all] [port <***port*>**] unresolved**

Mode

Enable

Description

The **arp clear** command lets you manually remove entries from the ARP table. The command can remove both dynamic and permanent entries.

Parameters

<*host*> Hostname or IP address of the ARP entry to remove.

all Remove all ARP entries, thus clearing the entire ARP table.

interface Specify this optional parameter to clear only entries in the ARP table that corresponds to a specific interface.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

<*string*> Specifies the interface name.

all Specifies all interfaces.

port Specify this optional parameter to clear only entries in the ARP table that corresponds to a specific exit port.

<*port*> Specifies the exit port.

unresolved Specify this optional parameter to clear only currently unresolved entries.

Examples

To remove the ARP entry for the host 10.8.1.2 from the ARP table:.

```
xp# arp clear 10.8.1.2
```

To clear the entire ARP table.

```
xp# arp clear all
```

If the Startup configuration file contains **arp add** commands, the Control Module re-adds the ARP entries even if you have cleared them using the **arp clear** command. To permanently remove an ARP entry, use the **negate** command or **no** command to remove the entry. Here is an example of the **no** command:

```
xp# no arp add nfs2 macaddr 080020:13a09f exit-port et.3.1
```

This command removes the ARP entry for “nfs2”.

arp set drop-unresolved

Purpose

To specify how unresolved traffic is handled.

Format

arp set drop-unresolved disabled| enabled

Mode

Configure.

Description

The **arp set drop-unresolved** command lets you specify how to deal with traffic that cannot be resolved by the Address Resolution Protocol.

When the X-Pedition receives an IP packet with an unknown nexthop MAC address, the router will attempt to resolve it by broadcasting an ARP request on the destination subnet. If the host replies to the ARP, the router will forward the packet to the host—however, if the router does not receive a reply, it will send one ARP request for each of the next four data packets it receives. If the ARP has not been resolved when the sixth packet arrives, the X-Pedition will (by default) drop the sixth and all subsequent packets *in software* for 20 seconds and transmit ICMP destination unreachable messages back to the sender(s) for each new packet received. The X-Pedition will then remove the unresolved ARP entry and the resolution process will resume.

When the **arp set drop-unresolved** command is enabled, any unresolved ARP that has not started sending ICMP destination unreachable messages will have its sixth and all subsequent packets dropped *in hardware* through the addition of a Layer-3 drop flow—*no ICMP message will be sent*.

Furthermore, the X-Pedition will cycle periodically through the list of all unresolved ARPs and re-send ARP requests in an attempt to resolve their nexthop MAC addresses. To configure the frequency of the resolution attempts and the maximum number of ARP requests sent with each attempt, use the **arp set unresolve-timer** and **arp set unresolve-threshold** commands. The behavior you configure will last until the Layer-3 drop ages out or until the X-Pedition clears all expired and unresolved ARP entries (every 5 minutes). The router will then re-start the resolution process.

Parameters

disabled Specifies that all unresolved ARP traffic will be handled by *software*, and that ICMP destination unreachable *messages will be sent* if the IP address cannot be resolved. This is the default behavior.

enabled Specifies that all unresolved ARP traffic will be dropped by the *hardware* with ICMP destination unreachable *messages suppressed*. The X-Pedition will then attempt to resolve the ARP periodically, according to the unresolve-threshold and the unresolve-timer.

Restrictions

None.

Examples

To drop IP packets with unresolved nexthop MAC addresses in *hardware* and *suppress* ICMP destination unreachable messages after ARP fails to resolve the IP address:

```
xp# arp set drop-unresolved enabled
```

arp set interface

Purpose

Set the lifetime (in seconds) of un-accessed ARP entries.

Format

arp set interface <*name*>**| all keep-time** <*number*>

Mode

Configure

Description

The **arp set interface** command lets users specify the amount of time (in seconds) to keep un-accessed ARP entries. ARP entries not accessed during the defined keep-time value are deleted the next time the system checks for un-accessed entries (by default, once every 5 minutes). As a result, an ARP entry may not be deleted immediately after the keep-time passes.

Parameters

interface <*name*>**| all** Name of the interface(s) for which you will define the lifespan.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

keep-time <*number*> Number of seconds determining lifespan of ARP interfaces. The default value is 1200 seconds (20 minutes).

Restrictions

The X-Pedition clears all expired and unresolved ARP entries once every 5 minutes. Therefore, expired ARPs may be kept up to 5 minutes longer than the keep-time.

arp set max-unresolved

Purpose

This command allows users to limit the number of unresolved ARP entries created by the X-Pedition.

Format

arp set max-unresolved <*num*>

Mode

Configure.

Description

The **arp set max-unresolved** command lets you specify the maximum number of unresolved ARP entries the X-Pedition may create. When the number of unresolved ARP entries exceeds this limit, the older ARP entries are removed to make room for the new ones.

Please note that the X-Pedition may stop sending ICMP host unreachable messages and transmit more ARP requests if the number of unresolved ARPs exceeds the specified limit and the entries are constantly removed and relearned. However, if the maximum is set too high, the router may exhaust available system memory and suffer degraded performance. Under normal network conditions, the number of unresolved ARPs is only a small fraction of the total number of ARP entries created, but network events such as a route change or an STP topology change can temporarily increase the number of unresolved ARP entries. Since the frequency of these events varies from network to network, there is no global solution. Users are encouraged to experiment on their own—if you are unsure of where to set the max-threshold, Enterasys recommends leaving the number at 1000 (the default).

Parameters

<*num*> The maximum number of unresolved ARP entries an X-Pedition can keep. Requires a number greater than or equal to 500. The default is 1000.

Restrictions

None.

Examples

To limit the number of unresolved ARP entries to under 500:

```
xp# arp set max-unresolved 500
```

arp set unresolve-threshold

Purpose

This command allows users to limit the number of unresolved ARPs the X-Pedition will periodically attempt to resolve if **arp set drop-unresolved** is enabled.

Format

arp set unresolve-threshold <*num*>

Mode

Configure.

Description

The **arp set unresolve-threshold** command lets you specify the maximum number of ARP requests sent in each periodic resolution attempt. When **arp set drop-unresolved** is enabled, the X-Pedition will periodically cycle through the list of all unresolved ARPs and send ARP requests in an attempt to resolve their nexthop MAC addresses. This command controls the number of ARP entries the router attempts to resolve. Also see *arp set drop-unresolved* on page 108.

Parameters

<*num*>	The maximum number of ARP requests sent in each periodic resolution attempt. Requires a number greater than or equal to 1. The default is 50.

Restrictions

This command has no effect unless **arp set drop-unresolved enabled** command is configured.

Examples

To increase the maximum number of ARP requests sent during each attempt to 100:

```
xp# arp set unresolve-threshold 100
```

arp set unresolve-timer

Purpose

Allows users to specify the frequency of the periodic resolution attempts when the **arp set drop-unresolved** command is enabled.

Format

arp set unresolve-timer <*num*>

Mode

Configure.

Description

The **arp set unresolve-timer** command lets you specify the frequency of the periodic resolution attempts. When the **arp set drop-unresolved** command is enabled, the X-Pedition will periodically cycle through the list of all unresolved ARPs and re-send ARP requests in an attempt to resolve their nexthop MAC addresses. This command controls how often to make these resolution attempts. Refer to *arp set drop-unresolved* on page 108 for details.

Parameters

<*num*>	The interval (in seconds) between each periodic resolution attempt. Requires a number greater than or equal to 10. The default is 10.

Restrictions

This command has no effect unless the **arp set drop-unresolved enabled** command is configured.

Examples

To increase the interval between each subsequent resolution attempt to 30 seconds:

```
xp# arp set unresolve-timer 30
```

arp show

Purpose

Display the ARP table.

Format

arp show <*IPaddr*>**|all [undecoded] [unresolved] [interface** <*string*>**| all] [port <***port*>**]**

Mode

Enable

Description

The **arp show** command displays the entire ARP table.

Parameters

<*IPaddr*> Shows the ARP entry for the specified IP address.

all Shows all entries in the ARP table.

undecoded Specify this optional parameter to show MAC addresses in hexadecimal format.

unresolved Specify this optional parameter to show only MAC addresses in the ARP table that have yet to be mapped to an network layer address.

interface Specify this optional parameter to show only addresses in the ARP table that is associated with the specific interface.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

<*string*> Specifies the interface name.

all Specifies all interfaces.

port Specify this optional parameter to show only addresses in the ARP table that corresponds to a specific exit port.

<*port*> Specifies the exit port.

Chapter 8

atm Commands

Command Summary

Table 8 lists the **atm** commands. The sections following Table 8 describe the command syntax for each command.

Note: Interfaces configured with PVCs do not support LSNAT or VRRP.

Table 8. atm commands

atm apply service <*string*> **port** <*port list*>
atm create vcl port <*port list*> **[vbr]**
atm define service <*string*> **[srv-cat ubr| cbr| rt-vbr| nrt-vbr] [pcr] | [pcr-kbits] [scr] | [scr-kbits] [mbs] [encaps llc-mux| vc-mux] [oam on| off] [oam-f5-type current-segment | end-to-end]**
atm set peer-addr port <*port*> **ip-address** <*ipaddr*>**|ipx-address** <*netaddr*>.<*macaddr*>
atm set port <*port list*> **cell-mapping direct| plcp**
atm set port <*port list*> **pdh-cell-scramble on| off**
atm set port <*port list*> **vpi-bits** <*num*>
atm set vcl port <*port*> **forced-bridged**
atm show [vpl port <*port list*>**| all [summary]] | [vcl port** <*port list*>**| all [summary]] [service** <*string*>**| all] | [port-settings** <*port list*>**| all-ports] | [stats port** <*port list*>**]**

atm apply service

Purpose

Apply a service profile.

Format

atm apply service <*string*> **port** <*port list*>

Mode

Configure

Description

The **atm apply service** command applies a service profile to a virtual channel (VC), virtual path (VP), and/or atm port. Service profiles define certain preset values for traffic and QoS parameters. Each service profile has its own unique set of traffic and QoS guarantees in handling transmission of ATM cells.

An important concept when applying service profile definitions is the concept of inheritance. Since a service profile definition can be applied to a virtual channel, virtual path, or on a port; the actual connection can inherit the service profile definition from any one of the three. The virtual channel will inherit the service profile definition that is directly applied on it. If no service profile was applied to the virtual channel, the connection will inherit the service profile applied to the virtual path. If no service profile definition was applied to the virtual path, then the connection will inherit the service profile applied to the ATM port. If no service profile was applied to the port, then the default service profile UBR is applied.

The following service classes are supported: CBR (constant bit rate), rt-VBR (real-time variable bit rate), nrt-VBR (non real-time variable bit rate), and UBR (unspecified bit rate). ABR (available bit rate) is not currently supported.

Parameters

<*string*> Is the character string of a previously-defined service. You define a service using the **atm define service** command (see page 122).

<*port list*> Is the port name, in the format: **`media.slot.port.vpi.vci`**

media	Is the media type. This is **at** for an ATM port.
slot	Is the slot number where the module is installed.
port	Is the number of the port through which data is passing.
vpi	Is the Virtual Path Identifier. This parameter is optional.
vci	Is the Virtual Channel Identifier. This parameter is optional.

Examples

To apply the pre-defined service profile ‘CBR1’ to virtual channel at.5.1.1.100:

```
xp(config)# atm apply service CBR1 port at.5.1.1.100
```

To apply the pre-defined service profile ‘CBR1’ to virtual path at.5.1.1:

```
xp(config)# atm apply service CBR1 port at.5.1.1
```

To apply the pre-defined service profile ‘CBR1’ to port at.5.1:

```
xp(config)# atm apply service CBR1 port at.5.1
```

atm create vcl port

Purpose

Create a virtual channel.

Format

atm create vcl port <*port list*> **[vbr]**

Mode

Configure

Description

The **atm create vcl** command creates a virtual channel on an ATM port. Virtual channels are point to point cell-switched connections used for ATM cell traffic. Virtual channels are defined by specifying a VCI (Virtual Channel Identifier) and VPI (Virtual Path Identifier) pair.

The range of available VCI and VPI are set by the **atm set port vpi-bits** command later in this chapter.

Note: Be careful when specifying VCI numbers 0 through 31. Those VPI/VCI pairs are used by some protocols for signaling purposes.

Parameters

<*port list*> Is the port name, in the format: **`media.slot.port.vpi.vci`**

media Is the media type. This is **at** for an ATM port.

slot Is the slot number where the module is installed.

port Is the number of the port through which data is passing.

vpi Is the Virtual Path Identifier.

vci Specifies the Virtual Channel Identifier. This number identifies a particular virtual channel. The combination of VPI and VCI is known as the VPI/VCI pair, and identifies the virtual channel.

Note: Do not specify VCI numbers 0 through 31. Some protocols use these VPI/VCI pairs for signaling purposes.

vbr Opens the VC with a default VBR service. All VCs to which you will apply a VBR service must be created with this option for traffic shaping to behave properly. If you specify the 'vbr' option, you may apply only VBR services to the VC. If you do not specify the 'vbr' option, you may apply only UBR and CBR services to the VC.

Note: Traffic on ATM Virtual Circuits configured with a Variable Bit Rate (nrt-vbr or rt-vbr) traffic descriptor will not obey the configured traffic descriptor's parameters. ATM policing mechanisms will drop nonconforming ATM cells.

Restrictions

None.

Examples

To create a virtual channel on slot 5, port 1, VPI 1, and VCI 100:

```
xp(config)# atm create vcl port at.5.1.1.100
```

To create many virtual channels simultaneously:

```
xp(config)# atm create vcl port at.5.1.(1,3-5,7).(100,555-600,700)
```

The following commands create an ATM virtual channel on an ATM port and associate the port with an IPX interface. This allows IPX routing between two IPX interfaces. As with any IPX interface, IPX routing using RIP (the default) will begin when you configure an IPX interface.

```
xp(config)# atm create vcl port at.3.1.1.100
xp(config)# interface create ipx finance address 01234567 peer-address 01234567.00:00:1d:a9:8c:a1
port at.3.1.1.100
xp(config)# interface create ipx marketing address 01234569 port et.1.1
```

atm define service

Purpose

Define a service profile.

Format

atm define service <*string*> **[srv-cat ubr| cbr| rt-vbr| nrt-vbr] [pcr] | [pcr-kbits] [scr] | [scr-kbits] [mbs] [encaps llc-mux| vc-mux] [oam on| off] [oam-f5-type current-segment | end-to-end]**

Mode

Configure

Description

The **atm define** command defines a set of traffic parameters. You can then apply this set of traffic parameters to a virtual channel. Quality of Service (QoS) parameters define the delays, dependability, and peak limits for a virtual channel. Class of Service defines the bandwidth guarantees. When a virtual channel is established, a service profile definition created by this command can then be applied to the connection.

Parameters

<*string*> Is a character string. The maximum length is 32 bytes.

srv-cat Is the service category (UBR is the default):

cbr Constant Bit Rate. This service category provides a guaranteed constant bandwidth specified by the Peak Cell Rate (PCR). This service requires only the PCR value. The Sustainable Cell Rate (SCR) and Maximum Burst Size (MBS) values are ignored. This service category is intended for applications that require constant cell rate guarantees such as uncompressed voice or video transmission.

ubr Unspecified Bit Rate. This service category is strictly best effort and runs at the available bandwidth. Users may limit the bandwidth by specifying a PCR value. The SCR and MBS are ignored. This service class is intended for applications that do not require specific traffic guarantees. UBR is the **default**.

nrt-vbr Non Real-Time Variable Bit Rate. This service category provides a guaranteed constant bandwidth (specified by the SCR), but also provides for peak bandwidth requirements (specified by the PCR). This service category

requires the PCR, SCR, and MBS options and is intended for applications that can accommodate bursty traffic with no need for real-time guarantees.

rt-vbr Real-Time Variable Bit Rate. This service category provides a guaranteed constant bandwidth (specified by the SCR), but also provides for peak bandwidth requirements (specified by the PCR). This service category requires the PCR, SCR, and MBS options and is intended for applications that can accommodate bursty real-time traffic such as compressed voice or video.

pcr Peak Cell Rate. This rate specifies the maximum cell transmission rate, expressed in cells/sec. The **default** is 353207 cells/sec for ATM OC-3. This parameter is valid for CBR, rtVBR, nrtVBR, and UBR service categories. This parameter is optional for UBR.

pcr-kbits Is the Peak Cell Rate, and specifies the maximum cell transmission rate, expressed in kbits/sec. The **default** is 149759 kbits/sec (353207 cells/sec) for ATM OC-3. This is the same as PCR, but is expressed in kbits/sec, and therefore may be a more convenient form. However, since the natural unit for ATM is cells/sec, there may be a difference in the actual rate because the kbit/sec value may not be an integral number of cells. This parameter is valid for CBR, rtVBR, nrtVBR, and UBR service categories.

scr Sustainable Cell Rate. This rate specifies the average cell rate, expressed in cells/sec. The **default** is 0 cells/sec. This parameter is valid only for rtVBR and nrtVBR service categories.

scr-kbits Sustainable Cell Rate expressed in kbits/sec. The **default** is 0 kbits/sec. This is the same as SCR, but is expressed in kbits/sec, and therefore may be a more convenient form. However, since the natural unit for ATM is cells/sec, there may be a difference in the actual rate because the kbit/sec value may not be an integral number of cells. This parameter is valid only for rtVBR and nrtVBR service categories.

mbs Is the Maximum Burst Size in cells. **MBS** specifies how many cells (2 to 255) can be transmitted at the Peak Cell Rate. The **default** is 0 cells. This parameter is valid only for rtVBR and nrtVBR service categories.

encaps Is the encapsulation scheme to transport multi protocol data over the AAL5 layer. Either **llc-mux** (logical link control based on multiplexing) or **vc-mux** (virtual channel-based multiplexing). The default is **llc-mux**.

oam OAM (Operation, Administration, and Management) loopback cells are used to provide loopback capabilities and confirm whether a VC connection is up. Only F5 OAM segments and end-to-end are supported, which provides loopback capabilities on a VC connection level. This parameter turns OAM ON or OFF on the PVC. The default is **off**. OAM OFF means that the X-Pedition responds to F5 OAM requests, but will not generate F5 OAM responses.

oam-f5-type

Used to specify the path of the OAM cells. Select either current-segment or end-to-end. The **default** is current-segment.

Restrictions

scr can not exceed **pcr**. No parameters may exceed the link rate for the type of **phy**.

Examples

To define a 10Mbps service:

```
xp(config)# atm define service CBR-example srv-cat cbr pcr_kbits 10000
```

atm set peer-addr

Purpose

Maps peer address to virtual channels.

Format

atm set peer-addr port <*port*> **ip-address** <*ipaddr*>**|ipx-address** <*netaddr*>.<*macaddr*>

Mode

Configure

Description

The **atm set peer-addr** command allows you to map a peer address for an ATM port to a specific virtual channel. This allows you to associate a specific virtual channel and its interface to a specific peer address.

Parameters

port <*port*> Specifies a single port, including virtual channel, in the format: **media.slot.port.vpi.vci**.

- **media** Is the media type. This is always **at** for an ATM port.
- **slot** Is the slot number where the module is installed.
- **port** Is the number of the port through which data is passing.
- **vpi** Is the Virtual Path Identifier.
- **vci** Is the Virtual Channel Identifier.

ip-address <*ipaddr*>
Specifies an IP address for the peer. Specify a unicast IP address and netmask value in the following format: **a.b.c.d**/e. This IP address will be mapped to the VC.

ipx-address <*netaddr*>.<*macaddr*>
Specifies an IPX address for the peer. Specify an IPX network and node address in the following format: **a1b2c3d4.aa:bb:cc:dd:ee:ff**. If a <*macaddr*> is not specified, then a wildcard address is used. This IPX address will be mapped to the VC.

Restrictions

None.

Example

To map the peer address 10.0.0.100/24 to the virtual channel at.4.1.0.100:

```
xp (config)# atm set peer-addr ports at.4.1.0.100 ip-address 10.0.0.100/24
```

atm set port cell-mapping

Purpose

Sets the format used to map ATM cells.

Format

atm set port <*port list*> **cell-mapping direct| plcp**

Mode

Configure

Description

The **atm set port cell-mapping** command specifies the format for mapping ATM cells into PDH (plesiochronous digital hierarchy) T3 and E3 frames. The ATM cells that each frame carries does not fit exactly into the PDH frame, therefore mapping of the data is necessary to ensure efficient transmission.

Parameters

<*port list*>	Specifies the ATM port(s). Specify **all-ports** to select all ports.
direct	Specifies ATM direct mapping. Default.
plcp	Specifies physical layer convergence protocol mapping.

Restrictions

Cell mapping is valid only for T3 and E3 PHY interfaces.

Example

To set cell-mapping to plcp for ATM port at.9.1:

```
xp(config)# atm set port at.9.1 cell-mapping plcp
```

atm set port pdh-cell-scramble

Purpose

Enables cell scrambling for ATM ports.

Format

atm set port <*port list*> **pdh-cell-scramble on| off**

Mode

Configure

Description

The **atm set port pdh-cell-scramble** command allows you to enable payload scrambling for PDH (plesiochronous digital hierarchy) PHY interfaces for the ATM line card, such as T1, T3, E1, and E3. Scrambling a payload is important in optimizing the transmission density of the data stream. Since all transmission use the same source clock for timing, scrambling the payload using a random number generator converts the data stream to a more random sequence. This ensures optimal transmission density of the data stream.

Parameters

port <*port list*> Specifies the port, in the format: **`media.slot.port`**. Specify **all-ports** to enable cell scrambling on all ports.

- **media** Specifies the media type. This is **at** for ATM ports**.**
- **slot** Specifies the slot number where the module is installed.
- **port** Specifies the port number.

on Enables cell scrambling.

off Disables cell scrambling.

Restrictions

This command is valid only for PDH PHY interfaces. SONET frames are scrambled using the SONET commands.

Example

To enable cell scrambling for ATM port at.9.1:

```
xp(config)# atm set port pdh-cell-scramble on
```

atm set port vpi-bits

Purpose

Sets the bit allocation for VPI on an ATM port.

Format

atm set port <*port list*> **vpi-bits** <*num*>

Mode

Configure

Description

The **atm set port vpi-bits** command allows you to set the number of bits allocated for VPI on an ATM port. There are 12 bits available for each VPI/VCI pair. The number of bits allocated define the amount of VPI and VCI values available. The following equations define the number of virtual paths and virtual channels:

# of virtual paths = 2^n; where *n* is the number of bits allocated for VPI and *n* is a value from 1 to 4

# of virtual channels = 2^{12-n}; where *n* is the number of bits allocated for VCI

Since there are only 12 bits available for each VPI/VCI pair, the more bits you allocate for VPI, the less bits remain for VCI. This is a shared number of bits. With the bit allocation command, you set the number of bits allocated for VPI. In turn, this sets the remaining number of bits as the number of bits allocated for VCI. The maximum value for *n* is 4.

Note: Be careful when specifying VCI numbers 0 through 31. Those VPI/VCI pairs are used by some protocols for signaling purposes.

Note: The maximum value for *n* is 4.

Parameters

port <*port list*> This parameter identifies the ATM port. Specify this parameter in the format: **`media.slot.port`**. Specify **all-ports** to set bit allocation on all ports.

- **media** Specifies the media type. This is **at** for ATM ports**.**
- **slot** Specifies the slot number where the module is installed.
- **port** Specifies the port number.

vpi-bits <*num*>

This parameter sets the number of bits for VPI. Specify any number between 1 and 4 (the **default** is 1).

Restrictions

None.

Example

To allocate 3 bits for VPI on port at.9.1:

```
xp(config)# atm set port at.9.1 vpi-bits 3
```

atm set vcl

Purpose

Sets the VCL operation mode.

Format

atm set vcl port <*port*> **forced-bridged**

Mode

Configure

Description

The **atm set vcl** command enables forced bridging on a per-VC basis. Forced-bridging forces the VC to encapsulate all ingress/egress traffic into a Layer-2 frame. This formats all traffic on a VC as bridged traffic, better suited for inter operability with other routers.

Parameters

<*port*> Specifies a single port, including the virtual channel, in the format: **media.slot.port.vpi.vci**

- **media** The media type. This is always **at** for an ATM port.
- **slot** The slot number where the module is installed.
- **port** The number of the port through which data is passing.
- **vpi** The Virtual Path Identifier.
- **vci** The Virtual Channel Identifier.

forced-bridged Enables encapsulation of all traffic as Layer-2 bridged traffic. This parameter can be used for inter-operability between the Enterasys X-Pedition and other vendor products.

Restrictions

None.

Example

To encapsulate all traffic as bridged traffic on at.4.1.0.100:

```
xp (config)# atm set vcl port at.4.1.0.100 forced-bridged
```

atm show

Purpose

Display information specific to an ATM port.

Format

atm show [vpl port <*port list*>**| all [summary]] | [vcl port** <*port list*>**| all [summary]] [service** <*string*>**| all] | [port-settings** <*port list*>**| all-ports] | [stats port** <*port list*>**]**

Mode

Enable

Parameters

vpl port <*port list*>**| all [summary]**

Shows VPL configurations on a port.

Specify **at.slot.port** to display all VPL configurations on the port.

Specify **at.slot.port.vpl** to display only the specified VPL configuration on the port.

Specify **all** to display verbose VPL configurations on all ports.

Specify **summary** to display summarized VPL configuration in tabular form.

vcl port <*port list*>**| all [summary]**

Shows VCL configurations on a port.

Specify **at.slot.port** to display all VCLs configurations on the port.

Specify **at.slot.port.vpl** to display all VCL configurations for a specified VPL.

Specify **at.slot.port.vpl.vcl** to display only the specified VCL configuration on the port.

Specify **all** to display verbose VCL configurations on all ports.

Specify **summary** to display summarized VCL configuration in tabular form.

service <*string*>**| all**

Shows the profile for a defined service. Specify **all** to show all ATM service profiles.

port-settings <*port list*>**| all-ports**

Shows the characteristics of an ATM port that were set by the **port set** command.

Specify the port using the following format: **at.slot.port**. Specify **all-ports** to show characteristics of all ATM ports.

stats port <*port list*>

Specify **at.slot.port.vpl** to display traffic statistics for all VCLs within a specified VPL.

Specify **at.slot.port.vpl.vcl** to display traffic statistics for the specified VCL only.

Restrictions

None.

Examples

To display information about the VPL configurations on ATM port 1:

```
xp(atm-show)# vpl port at.9.1

VPL Table Contents for Slot 9, Port 1:
   Virtual Path Identifier:      1
   Administrative Status:       Up
   Operational Status:          Up
   Last State Change:           1581
   Service Definition:          default-OC3
      Service Class:            UBR
      Peak Bit Rate:             Best Effort
      Sustained Bit Rate:       0 Kbits/sec (0 cps)
      Maximum Burst Size:        0 cells
      Encapsulation Type:       LLC Multiplexing
      F5-OAM:                    Responses Only
      F5-OAM-Type:              Current Segment
```

- Virtual Path Identifier — Identifies a particular VP.
- Administrative Status — Shows whether the VP is a viable network element.
  Up indicates a viable network element.
  Down indicates a non-viable network element.
- Operational Status — Shows whether the VP is passing traffic.
  Up indicates traffic.
  Down indicates no traffic.
- Last State Change — Shows the last time the VP went up or down. Time is in seconds relative to system boot-up.
- Service Definition — Shows the name of the defined service and its traffic parameters

To display information about all the defined service profiles for UBR:

```
xp# atm show service all

default-OC3
   Service Class:            UBR
   Peak Bit Rate:             Best Effort
   Sustained Bit Rate:      0 Kbits/sec (0 cps)
   Maximum Burst Size:    0 cells
   Encapsulation Type:    LLC Multiplexing
   F5-OAM:                    Responses Only
   F5-OAM-Type:           Current Segment
```

- Service Class — Shows the type of service class.
  UBR indicates Unspecified Bit Rate
  CBR indicates Constant Bit Rate
  RT-VBR indicates Real-time Variable Bit Rate
  NRT-VBR indicates Non Real-time Variable Bit Rate
- Peak Bit Rate — Shows the maximum bit transmission rate.
- Sustained Bit Rate — Shows the average bit transmission rate (in Kilobits per second).
- Maximum Burst Size — Shows how many cells can be transmitted at the Peak Bit Rate.
- Encapsulation Type — Shows the encapsulation scheme to transport multi protocol data over the AAL5 layer.
  LLC Multiplexing indicates logical link control encapsulation (**default**).
  VC Multiplexing indicates VC-based multiplexing encapsulation.
- F5-OAM — Shows how OAM (Operation, Administration, and Management) loopback cells provide loopback capabilities and confirm whether a VC connection is up. F5 OAM segments and end-to-end are supported, which provides loopback capabilities on a VC connection level.
  Responses Only indicates that the port will respond but doesn't generate OAM cells.
  Requests & Responses indicates that the port will respond and generate OAM cells.
- F5-OAM-Type — Shows F5-OAM-Type setting.

To display port-setting information about ATM port 1:

```
xp(atm-show)# port-settings at.9.1
Port information for Slot 9, Port 1:
   Port Type:                T3 ATM coaxial cable
   Xmt Clock Source:         Local
   Scramble Mode:            Payload
   Line Coding:              B3ZS
   Cell Mapping:             Direct
   Framing                   Cbit-Parity
   VC Mode:                  1 bit of VPI, 11 bits of VCI
   Service Definition:       default-OC3
      Service Class:         UBR
      Peak Bit Rate:         Best Effort
      Sustained Bit Rate:    0 Kbits/sec (0 cps)
      Maximum Burst Size:    0 cells
      Encapsulation Type:    LLC Multiplexing
      F5-OAM:                Requests & Responses
      F5-OAM-Type:           Current Segment
```

- Port Type — Shows the type of PHY interface for the port.
- Xmt Clock Source — Shows the timing source for the port.
  Local indicates the on board clock oscillator as the timing source.
  Loop indicates the receiver input as the timing source.
- Scramble Mode — Shows the scramble/descramble mode for the port.
  None indicates no scrambling.
  Payload indicates scrambling of the payload only.
  Frame indicates scrambling of the stream only.
  Both indicates scrambling of payload and stream.
- Line Coding — Shows the particular DS1/T1 and DS3/T3 coding convention.
- Cell Mapping — Shows the format used to map ATM cells.
  Direct indicates direct cell mapping.
  Plcp indicates physical layer convergence protocol mapping.
- Framing — Shows the type of framing scheme.
  cbit-parity is used for T3 framing.
  m23 is used for T3 framing.
  esf indicates extended super frame and is used for T1 framing.
  g832 is used for E3 framing.
  g751 is used for E3 framing.
- VC Mode — Shows the bit allocation for VPI and VCI.
- Service Definition — Shows the name of the defined service on the port and its traffic parameters.

Chapter 9

bgp Commands

The **bgp** commands let you display and set parameters for the Border Gateway Protocol (BGP).

Notes:

- BGP management traps are not supported in this release.
- The X-Pedition does not currently follow “Breaking Ties (Phase2),” Section 9.1.2.1 (p. 37-38) of RFC 1771. Instead, the router follows “Breaking Ties (Phase2),” Section 9.1.2.2 (p. 49-50) of Draft-ietf-ier-bgp-4-17.

Command Summary

Table 9 lists the **bgp** commands. The sections following the table describe the command syntax.

Table 9. bgp commands

bgp add network <*ipaddr-mask*>**|all group** <*number-or-string*>
bgp add peer-host <*ipaddr*> **group** <*number-or-string*>
bgp clear peer-host <*ipaddr*>
bgp create peer-group <*number-or-string*>
bgp set DampenFlap <*option*>
bgp set default-metric <*num*>
bgp set cluster-id <*ipaddr*>
bgp set multipath off
bgp set peer-group <*number-or-string*>

Table 9. bgp commands (Continued)

bgp set peer-host <*ipaddr*>
bgp set preference <*num*>
bgp show aspaths <*aspath*>|**all** [**to-terminal**|**to-file**]
bgp show cidr-only <*ip-addr-mask*>|**default**|**all** [**to-terminal**|**to-file**]
bgp show community community-id <*number*> **autonomous-system** <*number*>| **well-known-community [no-export**|**no-advertise**|**no-export-subconfed]**| **reserved-community** <*number*>**]** [**to-terminal**|**to-file]**
bgp show peer-as <*number*> **[to-terminal**|**to-file]**
bgp show peer-group-type external|**internal**|**routing [to-terminal**|**to-file]**
bgp show peer-host <*ipaddr*> **received-routes**|**all-received-routes**|**advertised-routes [to-terminal**|**to-file]**
bgp show regexp to-terminal|**to-file**
bgp show routes <*ip-addr-mask*>|**default**|**all [to-terminal**|**to-file]**
bgp show summary [to-terminal|**to-file]**
bgp show sync-tree
bgp start|**stop**
bgp trace packets [detail| **send**| **receive**| **group** <*number-or-string*>| **peer-host** <*ipaddr*>**]**| **open [detail**| **send**| **receive**| **group** <*number-or-string*>| **peer-host** <*ipaddr*>**]**| **update [detail**| **send**| **receive**| **group** <*number-or-string*>| **peer-host** <*ipaddr*>**]**| **keep-alive [detail**| **send**| **receive**| **group** <*number-or-string*>| **peer-host** <*ipaddr*>**]**| **aspath [group** <*number-or-string*>| **peer-host** <*ipaddr*>**]**| **local-options [all**| **general**| **state**| **normal**| **policy**| **task**| **timer**| **route**| **group** <*number-or-string*>| **peer-host** <*ipaddr*>**]**

bgp add network

Purpose

Adds a network to a BGP peer group.

Format

bgp add network <*ip-addr-mask*>|**all group** <*number-or-string*>

Mode

Configure

Description

The **bgp add network** command lets you add a BGP peer network, thus allowing peer connections from any addresses in the specified range of network and mask pairs.

Parameters

network <*ip-addr-mask*>|**all**
Specifies a network from which peer connections are allowed. Specify an IP address and Mask value. Example: 1.2.3.4/255.255.0.0 or 1.2.3.4/16. Specify **all** to add all networks.

group <*number-or-string*>
Specifies the group ID associated with this network range.

Restrictions

None.

bgp add peer-host

Purpose

Add a BGP peer by adding a peer host.

Format

bgp add peer-host <*ipaddr*> **group** <*number-or-string*>

Mode

Configure

Description

The **bgp add peer-host** command adds a peer-host to a BGP group.

Parameters

peer-host <*ipaddr*>
Specifies the peer host's IP address.

group <*number-or-string*>
Specifies the group ID of the group to which the peer host belongs.

Restrictions

When adding multiple peer hosts to a peer group, you may not connect more than one peer group to the same AS.

bgp clear peer-host

Purpose

Disconnect and re-establish a peer connection.

Format

bgp clear peer-host <*ipaddr*>

Mode

Enable.

Description

The **bgp clear peer-host** command sends a notification packet to the selected peer host, causing the peer session to close and re-establish.

Parameters

peer-host <*ipaddr*>
Specifies the peer host's IP address.

Restrictions

None.

bgp create peer-group

Purpose

Create a BGP group based on a type and autonomous system number. You may create any number of groups, but each group must have a unique combination of type and autonomous system.

Format

bgp create peer-group <*number-or-string*> **type external|internal|routing**
[autonomous-system <*number*>**]**
[proto any|rip|ospf|static]
[interface <*interface-name-or-ipaddr*> |**all]**

Mode

Configure

Description

The **bgp create peer-group** command creates an entity into which peers are added. Peers are added to this group with the commands **bgp add peer-host** or **bgp add network**. Once a group is created, group-wide parameters may be applied with the **bgp set peer-group** command.

Parameters

peer-group <*number-or-string*>
Is a group ID, which can be a number or a character string.

type Specifies the type of BGP group you are adding. Specify one of the following:

external Use for external BGP peers. Full policy checking is applied to all incoming and outgoing advertisements. All peers in this group must be directly reachable through a local interface (i.e., they must be L2 adjacent).

internal Use for IBGP peers only, where no IGP is used. This group expects all peers to be Layer-2 adjacent so that next hops received in updates can be used directly for forwarding.

routing An IBGP type that uses the routes of an interior protocol to resolve forwarding addresses (this implementation comes closest to the IBGP implementation of other router vendors). This type will determine the immediate next hops for routes by using the next hop received with a route from a peer as a forwarding address, and using this to look up an immediate next hop in an IGP's routes. Such groups support distant peers, but need to be informed of the IGP whose routes they are using to determine immediate next hops.

autonomous-system
Specifies the autonomous system of the peer group. Specify a number from 1 – 65534.

proto Used for group-type routing only. Specifies the interior protocol to use to resolve BGP next hops. Specify one of the following:

- **any** Use any IGP to resolve BGP next hops.
- **rip** Use RIP to resolve BGP next hops.
- **ospf** Use OSPF to resolve BGP next hops.
- **static** Use static to resolve BGP next hops.

interface <*interface-name-or-IPaddr*> |**all**
Interfaces whose routes are carried via the IGP for which third-party next hops may be used instead. Use only with parameter type ROUTING. Specify the interface or use **all** to use all interfaces.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

None.

bgp set cluster-id

Purpose

Specifies the route reflection cluster ID for BGP.

Format

bgp set cluster-id <*ipaddr*>

Mode

Configure

Description

The **bgp set cluster-id** command specifies the route reflection cluster ID for BGP. The cluster ID defaults to the same as the router-id. If a router is to be a route reflector, then a single cluster ID should be selected and configured on all route reflectors in the cluster. If there is only one route reflector in the cluster, the cluster ID setting may be omitted, as the default will suffice.

Parameters

cluster-id <*ipaddr*>
Is the cluster ID.

Restrictions

The only constraints on the choice of cluster ID are (a) IDs of clusters within an AS must be unique within that AS, and (b) the cluster ID must not be 0.0.0.0. Choosing the cluster ID to be the router ID of one router in the cluster will always fulfill these criteria.

bgp set DampenFlap

Purpose

Configures parameters for Weighted Route Dampening.

Format

bgp set dampenflap [state enable|disable]|[suppress-above <*num*>**]|**
[reuse-below <*num*>**]|[max-flap** <*num*>**]|[unreach-decay** <*num*>**]|**
[reach-decay <*num*>**]|[keep-history** <*num*>**]**

Mode

Configure

Description

The **bgp set dampenflap** command configures the state of Weighted Route Dampening.

Parameters

state enable|disable
Causes the Route Instability History to be maintained (**enable** option) or not (**disable** option).

suppress-above <*num*>
Is the value of the instability metric at which route suppression will take place. A route will not be installed in the FIB or announced even if it is reachable during the period that it is suppressed. The default is 3.0.

reuse-below <*num*>
Is the value of the instability metric at which a suppressed route will become *un*suppressed, if it is reachable but currently suppressed. The value must be less than that for the suppress-above option. The default is 2.0.

max-flap <*num*>
Is the upper limit of the instability metric. This value must be greater than the larger of 1 and that for suppress-above. The default is 16.0.

unreach-decay <*num*>
Specifies the time in seconds for the instability metric value to reach one-half of its current value when the route is *unreachable*. This half-life value determines the rate at which the metric value is decayed. The default is 900.

reach-decay <*num*>
Specifies the time in seconds for the instability metric value to reach one half of its current value when the route is *reachable*. This half-life value determines the rate at which the

metric value is decayed. A smaller half-life value will make a suppressed route reusable sooner than a larger value. The default is 300.

keep-history <*num*>
Specifies the period in seconds over which the route flapping history is to maintained for a given route. The size of the configuration arrays is directly affected by this value. The default is 1800.

Restrictions

None.

bgp set default-metric

Purpose

Set the metric used when advertising routes through BGP.

Format

bgp set default-metric <*num*>

Mode

Configure

Description

The **bgp set default-metric** command lets you set the default metric BGP uses when it advertises routes. If this command is not specified, no metric is propagated. This metric may be overridden by a metric specified on the neighbor or group statements or in an export policy.

Parameters

<*num*> Specifies the default cost. Specify a number from 0 - 65535.

Restrictions

None.

bgp set multipath

Purpose

Disables multipath route calculation for BGP routes.

Format

bgp set multipath off

Mode

Configure

Description

The **bgp set multipath** command disables multipath route calculation for BGP routes. No multipath forwarding occurs as a result of this command.

Parameters

off

Disables multipath route calculation for BGP routes.

Restrictions

If you negate this command from the active configuration file, the X-Pedition will not automatically recreate multipath routes. To recreate multipath routes, stop and restart bgp.

bgp set peer-group

Purpose

Set parameters for the specified BGP Peer Group.

Format

bgp set peer-group <*number-or-string*>
[
med|reflector-client|no-client-reflect|[metric-out <*num*>**]|[set-pref** <*num*>**]**
[local-pref <*num*>**] [local-as** <*num*>**]|ignore-first-as-hop|**
[generate-default enabled|disabled]|[gateway <*ipaddr*>**]|next-hop-self|**
[preference <*num*>**]|[preference2** <*num*>**]|[local-address** <*ipaddr*>**]|**
[hold-time <*num*>**]|[version 2|3|4]|passive|[send-buffer** <*num*>**]|**
[recv-buffer <*num*>**]|[in-delay** <*num*>**]|[out-delay** <*num*>**]|[keep all|none]|**
show-warnings|no-aggregator-id|keep-alives-always|v3-asloop-okay|no-v4-asloop|
[as-count <*num*>**]|log-up-down|[ttl** <*num*>**]|**
[optional-attributes-list <*number-or-string*>**]**
]

Mode

Configure

Description

The **bgp set peer-group** command sets parameters for the specified BGP group.

Parameters

group <*number-or-string*>
Specifies the group.

med
Forces med to be used for route selection process. By default, any metric (Multi_Exit_Disc, or MED) received on a BGP connection is ignored—to use MEDs in route selections, you *must* specify this option. Furthermore, the X-Pedition does not send MEDs on external connections. To send MEDs, use the **metric** option of the **ip-router policy create bgp-export-destination** command or select the **metric-out** parameter of the **bgp set** commands.

Note: Before the router can process and select the correct route based on the MED values received from other BGP peers, users must set the selection process in the active configuration of the router where the peer is defined. To set the selection process, enter one (or both) of the following commands in the configuration, depending on the type of BGP peer configured (i.e., peer group, peer host, or both):

bgp set peer-group <*group Name*> **med**
bgp set peer-host <*IP Address*> **med**

reflector-client |no-client-reflect
The **reflector-client** option specifies that GateD will act as a route reflector for this group. All routes received from any group member will be sent to all other internal neighbors, and all routes received from any other internal neighbors will be sent to the reflector clients. Since the route reflector forwards routes in this way, the reflector-client group need not be fully meshed. Use only for *internal* and *routing* groups.

If the **no-client-reflect** option is specified, routes received from reflector clients will only be sent to internal neighbors which are not in the same group as the sending reflector client. In this case the reflector-client group should be fully meshed. In all cases, routes received from normal internal peers will be sent to all reflector clients.

Note: It is necessary to export routes from the local AS into the local AS when acting as a route reflector. The reflector-client option specifies that GateD will act as a route reflector for this group. All routes received from any group member will be sent to all other internal neighbors, and all routes received from any other internal neighbors will be sent to the reflector clients. Since the route reflector forwards routes in this way, the reflector-client group need not be fully meshed.

metric-out <*num*>
Specifies the primary metric used on all routes sent to the specified peer group. Specify a number from 0 - 65535.

set-pref <*num*>
Unless you set this parameter, GateD will ignore the LOCAL-PREF value received in update packets sent from this peer group. Even then it will be used only in group types of internal or routing BGP configurations (i.e., IBGP). When set, the RIB will use LOCAL_PREF to calculate the preference of routes received in those updates. The preference (Prf1 in the RIB table) is calculated as (254 - LOCAL_PREF + set-pref). This ensures that you can never set Prf1 lower than the set-pref value by a received LOCAL_PREF. The global protocol preference (Prf1) determined by the RIB will be used to set LOCAL_PREF on transmitted updates.

local-pref <*num*>
This parameter allows you to set the BGP LOCAL_PREF attribute sent to this peer group in update packets. By default, LOCAL_PREF = 100. Use this parameter with a group type of *internal* or *routing* only. If you use the SET_PREF parameter, LOCAL_PREF is ignored—the LOCAL_PREF attribute will be determined from the RIB preference (Prf1) value instead.

local-as <*num*>
Identifies the autonomous system which the router is representing to this group of peers. The default is the one configured by the **set autonomous_system** command. Specify a number from 1 - 65534.

ignore-first-as-hop
Some routers, known as Route Servers, are capable of propagating routes without appending their own AS to the AS path. By default, GateD will drop such routes. Specifying ignore-first-as-hop here or on either the **create peer-group** or **set peer-host** CLI commands disables this feature. This option should only be used if it is positively known that the peer is a route server and not a normal router.

generate-default enabled|disabled
Specifies whether the router should generate a default route when BGP receives a valid update from its peer. If this option is not specified, then the generation of default route is enabled.

gateway <*ipaddr*>
If a network is not shared with a peer, this option specifies a router on an attached network to be used as the next hop router for routes received from this neighbor. This field is used for EBGP Multihop. **The IP address must be a host address on a locally attached network**.

next-hop-self
This option causes the next hop in route advertisements set to this peer or group of peers to be set to our own router's address even if it would normally be possible to send a third-party next hop. Use of this option may cause efficient routes to be followed, but it may be needed in some cases to deal with broken bridged interconnect media (in cases where the routers on the shared medium do not really have full connectivity to each other) or broken political situations. Use only for EXTERNAL groups.

preference <*num*>
Specifies the preference used for routes learned from these peers. Specify a number from 0-255.

preference2 <*num*>
In case of a preference tie, this option (the second preference), may be used to break the tie. The default value is 0. Specify a number from 0-255.

local-address <*ipaddr*>
Specifies the address to be used on the local end of the TCP connection with the peer or with the peer's gateway when the gateway option is used. A session with an external peer will only be opened when an interface with the appropriate local address (through which the peer or gateway address is directly reachable). In either case incoming connections will only be recognized as matching a configured peer if they are addressed to the configured local address. Use for *internal* and *routing* groups only. **It should be one of the interface addresses.**

hold-time <*num*>
Specifies the hold time value (in seconds) to use when negotiating the connection with this peer. If BGP does not receive a keepalive, update, or notification message from a peer within the period specified in the Hold Time field of the BGP Open message, the BGP connection will be closed. The value must be either 0 (no keepalives will be sent) or a value from 6 to 65,535.

Note: Every time a user changes the hold time for a BGP session (whether for a peer group or peer host), the X-Pedition will close and re-open the connection when the user saves the change to the active configuration.

version 2|3|4
Specifies the version of the BGP protocol to use with this peer. If not specified, only the specified version will be offered. Specify 2, 3, or 4.

passive
Specifies that active OPENs to this peer should not be attempted. BGP would wait for the peer to issue an OPEN. By default, all explicitly configured peers are active, they periodically send OPEN messages until the peer responds. Note that if it is applied to both sides of a peering session, it will prevent the session from ever being established.

send-buffer <*num*>
Controls the amount of send buffer acquired from the memory subsystem. The maximum supported is 65535 bytes. By default, BGP acquires the maximum supported. Specify a number from 4096 – 65535.

recv-buffer <*num*>
Controls the amount of receive buffer acquired from the memory subsystem. The maximum supported is 65535 bytes. By default, BGP acquires the maximum supported. Specify a number from 4096 – 65535.

in-delay <*num*>
Used to dampen route fluctuations. In delay specifies the amount of time in seconds a route learned from a BGP peer must be stable before it is accepted into the routing database. Specify a number equal to or greater than 0. The default value is 0, meaning that this feature is disabled.

out-delay <*num*>
Used to dampen route fluctuations. Out delay is the amount of time in seconds a route must be present in the routing table before it is exported to BGP. Specify a number equal to or greater than 0. The default value is 0, meaning that this feature is disabled.

keep all|none
Used to retain routes learned from a peer even if the routes' AS paths contain one of our exported AS numbers.

show-warnings
This option causes GateD to issue warning messages when receiving questionable BGP updates such as duplicate routes and/or deletions of non-existing routes. Normally these events are silently ignored.

no-aggregator-id
This option causes GateD to specify the router ID in the aggregator attribute as zero (instead of its router ID) in order to prevent different routers in an AS from creating aggregate routes with different AS paths.

keep-alives-always
This option causes GateD to always send keepalives, even when an update could have correctly substituted for one. This allows inter operability with routers that do not completely obey the protocol specifications on this point.

v3-asloop-okay
By default GateD will not advertise routes whose AS path is looped (i.e. with an AS appearing more than once in the path) to version 3 external peers. Setting this flag removes this constraint. Ignored when set on internal groups or peers.

no-v4-asloop
Prevents routes with looped AS paths from being advertised to version 4 external peers. This can be useful to avoid advertising such routes to peer which would incorrectly forward the routes on to version 3 neighbors.

as-count <*num*>
This option determines how many times the X-Pedition will insert its own AS number when we send the AS path to an external neighbor.

Specify a number between 1 and 25. The default is 1. Higher values typically are used to bias upstream neighbors' route selection. (All else being equal, most routers will prefer to use routes with shorter AS Paths. Using **ascount**, the AS Path the X-Pedition sends can be artificially lengthened.)

Note that ascount supersedes the **no-v4-asloop** option–regardless of whether **no-v4-asloop** is set, we will still send multiple copies of our own AS if the **as-count** option is set to something greater than one. Also, note that if the value of ascount is changed and GateD is reconfigured, routes will not be sent to reflect the new setting. If this is desired, it will be necessary to restart the peer session.

log-up-down
This option causes a message to be logged via the Syslog mechanism whenever a BGP peer enters or leaves the ESTABLISHED state.

ttl <*num*>
By default, BGP sets the IP TTL for local peers to ONE and the TTL for non-local peers to 255. This option is provided when attempting to communicate with improperly functioning routers that ignore packets sent with a TTL of ONE. Specify a number between 1 and 255.

optional-attributes-list <*number-or-string*>
Specifies the ID of the optional-attributes-list to be associated with this peer-group.

Restrictions

None.

bgp set peer-host

Purpose

Set parameters for a BGP peer host.

Format

bgp set peer-host <*ipaddr*> **[group** <*number-or-string*>|**[metric-out** <*num*>**]|**
[set-pref <*num*>**][local-as** <*num*>**]|ignore-first-as-hop|**
[generate-default enabled|disabled]|[gateway <*ipaddr*>**]|next-hop-self|**
[preference <*num*>**][preference2** <*num*>**][local-address** <*ipaddr*>**]|**
[hold-time <*num*>**]|[version 2|3|4]|passive|[send-buffer** <*num*>**]|**
[recv-buffer <*num*>**]|[in-delay** <*num*>**]|[out-delay** <*num*>**]|[keep all|none]|**
show-warnings|no-aggregator-id|keep-alives-always|v3-asloop-okay|
no-v4-asloop|[as-count <*num*>**]|[ttl** <*num*>**]|**
[optional-attributes-list <*number-or-string*>**]]**

Mode

Configure

Description

The **bgp set peer-host** command lets you set various parameters for the specified BGP peer host.

Parameters

group <*number-or-string*>
Specifies the group ID

metric-out <*num*>
Specifies the primary metric used on all routes sent to the specified peer. The metric hierarchy is as follows, starting from the most preferred: 1)The metric specified by export policy. 2) Peer-level metricout. 3) Group-level metricout 4) Default metric. For *internal* and *routing* peers use the **group** command to set the metric-out. Specify a number from 0 - 65535.

set-pref <*num*>
Allows BGP's LOCAL_PREF attribute to be used to set the GateD preference on reception, and allows the GateD preference to set the LOCAL_PREF on transmission. The set-pref metric works as a lower limit, below which the imported LOCAL_PREF may not set the GateD preference. For *internal* and *routing* peers, use the **group** command to set the metric-out. Specify a number from 0 - 255. **This parameter applies to *internal* and *routing* peers only.**

local-as <*num*>
Identifies the autonomous system the router is representing to this peer. The default is the one

configured using the **ip-router global set autonomous-system** command. Specify a number from 1 - 65534.

ignore-first-as-hop
Some routers, known as Route Servers, are capable of propagating routes without appending their own AS to the AS path. By default, GateD will drop such routes. Specifying ignore-first-as-hop here or on either the **bgp create peer-group** or **bgp set peer-host** CLI commands disables this feature. This option should only be used if it is positively known that the peer is a route server and not a normal router.

generate-default enabled|disabled
Specifies whether the router should generate a default route when BGP receives a valid update from its peer. If this option is not specified, then the generation of default route is enabled.

gateway <*IPaddr*>
if a network is not shared with a peer, this option specifies a router on an attached network to be used as the next hop router for routes received from this neighbor. This is used for **EBGP multihop**. **The IP address must be a host address on a locally attached network.**

next-hop-self
This option causes the next hop in route advertisements set to this peer to be set to our own router's address, even if it would normally be possible to send a third-party next hop. Use of this option may cause inefficient routes to be followed, but it may be needed in some cases to deal with broken bridged interconnect media (in cases where the routers in the shared medium do not really have full connectivity to each other) or broken political situations. **Use only for external peer hosts.**

preference <*num*>
Specifies the preference used for routes learned from this peer. This can differ from the default BGP preference set in the bgp set preference statement, so that GateD can prefer routes from one peer, or group of peer, over others. This preference may be explicitly overridden by import policy. Specify a number from 0 - 255.

preference2 <*num*>
In case of preference tie, this option (the second preference), may be used to break the tie. The default value is 0. Specify a number from 0 - 255.

local-address <*IPaddr*>
Specifies the address to be used on the local end of the TCP connection with the peer or with the peer's gateway when the gateway option is used. A session with an external peer will be opened only when an interface with the appropriate local address (through which the peer or gateway address is directly reachable) is operating. For other types of peers, a peer session will be maintained when any interface with the specified local address is operating. In either case incoming connections will only be recognized as matching a configured peer if they are addressed to the configured local address. For *internal* and *routing* peers, use the **group** command to set the local-address—**the address should be one of the interface addresses.**

hold-time <*num*>
Specifies the hold time value to use when negotiating the connection with this peer, in seconds. If BGP does not receive a keepalive, update, or notification message from a peer within the

period specified in the Hold Time field of the BGP Open message, then the BGP connection will be closed. The value must be either 0 (no keepalives will be sent) or at least 6.

Note: Setting the hold-time to 0 is not negotiated. If the remote peer attempts to negotiate any value other than 0, the higher value will be used. If you want to prevent keepalives from being exchanged, configure both peers to use a hold-time of 0.

version 2|3|4
Specifies the version of the BGP protocol to use with this peer. If not specified, only the specified version will be offered. Specify 2, 3, or 4.

passive
Specifies that active OPENs to this peer should not be attempted. BGP would wait for the peer to issue an OPEN. By default, all explicitly configured peers are active, they periodically send OPEN messages until the peer responds. Note that if it is applied to both sides of a peering session, it will prevent the session from ever being established.

send-buffer <*num*>
Controls the amount of send buffer acquired from the memory subsystem. The maximum supported is 65535 bytes. By default, BGP acquires the maximum supported. Specify a number from 4096 - 65535.

recv-buffer <*num*>
Controls the amount of receive buffer acquired from the memory subsystem. The maximum supported is 65535 bytes. By default, BGP acquires the maximum supported. Specify a number from 4096 – 65535.

in-delay <*num*>
Used to dampen route fluctuations. In delay specifies the amount of time in seconds a route learned from a BGP peer must be stable before it is accepted into the routing database. The default value is 0, meaning that this feature is disabled. Specify a number equal to or greater than 0.

out-delay <*num*>
Used to dampen route fluctuations. Out delay is the amount of time in seconds a route must be present in the routing table before it is exported to BGP. The default value is 0, meaning that this feature is disabled. Specify a number equal to or greater than 0.

keep all|none
Used to retain routes learned from a peer even if the routes' AS paths contain one of our exported AS numbers.

show-warnings
This option causes GateD to issue warning messages when receiving questionable BGP updates such as duplicate routes and/or deletions of non-existing routes. Normally these events are silently ignored.

no-aggregator-id
This option causes GateD to specify the router ID in the aggregator attribute as zero (instead of its router ID) in order to prevent different routers in an AS from creating aggregate routes with different AS paths.

keep-alives-always
This option causes GateD to always send keepalives, even when an update could have

correctly substituted for one. This allows inter operability with routers that do not completely obey the protocol specifications on this point.

v3-asloop-okay
By default GateD will not advertise routes whose AS path is looped (i.e. with an AS appearing more than once in the path) to version 3 external peers. Setting this flag removes this constraint. Ignored when set on internal groups or peers.

no-v4-asloop
Prevents routes with looped AS paths from being advertised to version 4 external peers. This can be useful to avoid advertising such routes to peer which would incorrectly forward the routes on to version 3 neighbors.

as-count <*num*>
This option determines how many times (1-25). we will insert our own AS number when we send the AS path to an external neighbor. Specify a number equal to or greater than 0. The default is 1. Higher values are typically used to bias upstream neighbors' route selection.

Note: **As-count** supersedes the **no-v4-asloop** option. If you set the **as-count** option to something greater than one, the X-Pedition will still send multiple copies of its own AS.

Also, note that if you change the value of **as-count** and reconfigure GateD, routes will not be sent to reflect the new setting. If you want to send new routes containing this information, you must restart the peer session. Use for external peer-hosts only.

log-up-down
Causes a message to be logged via the Syslog mechanism whenever a BGP peer enters or leaves the ESTABLISHED state.

ttl <*num*>
By default, BGP sets the IP TTL for local peers to ONE and the TTL for non-local peers to 255. This option is provided when attempting to communicate with improperly functioning routers that ignore packets sent with a TTL of ONE. Specify a number from 1-255.

optional-attributes-list <*num-or-string*>
Specifies the ID of the optional-attributes-list to be associated with this peer.

Restrictions

When adding multiple peer hosts to a peer group, you may not connect more than one peer group to the same AS.

bgp set preference

Purpose

Set BGP preference.

Format

bgp set preference <*num*>

Mode

Configure

Description

The **bgp set preference** command lets you set the BGP preference for the X-Pedition.

Parameters

<*num*> Specifies the preference of routes learned from BGP. Specify a number from 0 -255. The default preference is 170.

Restrictions

None.

bgp show aspaths

Purpose

Displays BGP AS path information

Format

bgp show aspaths <*aspath*>|**all** [**to-terminal**|**to-file]**

Mode

Enable

Description

The **bgp show aspaths** command displays information about a specified AS path or all AS paths. The AS path is listed along with the number of routes that use it.

Parameters

<*aspath*>	Displays information about the specified AS path.
all	Displays information about all AS paths.
to-terminal	Causes output to be displayed on the terminal. This is the default.
to-file	Causes output to be saved in the file **/gatedtrc/gated.dmp**.

Restrictions

None.

Example

To display information about all AS paths:

```
xp# bgp show aspaths all
Hash  Ref   Path
0     5     IGP (Id 1)
2     1     (64900) 64901 64902 IGP (Id 3)
7     4     (64900) 64901 IGP (Id 2)
```

bgp show cidr-only

Purpose

Display routes in the BGP routing table with CIDR network masks

Format

bgp show cidr-only <*ip-addr-mask*>**|all** [**to-terminal**|**to-file]**

Mode

Enable

Description

The **bgp show cidr-only** command displays the same type of route information as the bgp show routes command. The difference is that the **bgp show cidr-only** command limits the display to CIDR routes only.

Parameters

<*ip-addr-mask*>	Displays information about the specified CIDR route.
all	Displays information about all CIDR routes.
to-terminal	Causes output to be displayed on the terminal.
to-file	Causes output to be saved in the file **/gatedtrc/gated.dmp**.

Restrictions

None.

Example

To display information all CIDR routes in the X-Pedition's BGP route table:

```
xp# bgp show cidr-only all
Proto     Route/Mask NextHop        ASPath
BGP       12.2.19/25 207.135.89.65   (64800) 64753 64752 64751 6379 3561 11277 IGP (Id 13805)
BGP       12.5.172/22 207.135.89.65   (64800) 64753 64752 64751 6379 3561 1 IGP (Id 173)
BGP       12.5.252/23 207.135.89.65   (64800) 64753 64752 64751 6379 5646 1 7018 6301 IGP (Id 926)
BGP       12.6.42/23 207.135.89.65   (64800) 64753 64752 64751 6379 5646 1 7018 11090 IGP (Id 979)
BGP       12.6.134/23 207.135.89.65   (64800) 64753 64752 64751 6379 5646 1 701 7314 10562 IGP (Id 388)
BGP       12.7.214/23 207.135.89.65   (64800) 64753 64752 64751 6379 5646 7018 4129 IGP (Id 31004)
```

bgp show community

Purpose

Displays routes that belong to a specified community.

Format

bgp show community community-id <*number*> **autonomous-system** <*number*>**| well-known-community [no-export|no-advertise|no-export-subconfed]| reserved-community** <*number*>**]** [**to-terminal|to-file]**

Mode

Enable

Description

The **bgp show community** command displays routes that belong to a specified community in a specified autonomous system.

Parameters

community-id <*number*>
Is the community identifier portion of a community split. This is combined with the autonomous-system value entered to create a value for the community attribute.

autonomous-system <*number*> Is an autonomous system number.

well-known-community
Is one of the well-known communities. Specify one of the following:

no-export
Is a special community that indicates the routes associated with this attribute must not be advertised outside a BGP confederation boundary. Since the X-Pedition's implementation does not support confederations, this boundary is an AS boundary.

no-advertise
is a special community indicating that the routes associated with this attribute must not be advertised to other BGP peers.

no-export-subconfed
Is a special community indicating the routes associated with this attribute must not be advertised to external BGP peers. (This includes peers in other members' autonomous systems inside a BGP confederation.)

reserved-community <*number*>
This option specifies one of the reserved communities that is not well-known. A reserved

community is one that is in one of the following ranges (0x0000000 - 0x0000FFFF) or (0xFFFF0000 - 0xFFFFFFFF).

to-terminal
Causes output to be displayed on the terminal. This is the default.

to-file
Causes output to be saved in the file **/gatedtrc/gated.dmp**.

Restrictions

None.

Example

To display routes that belong to community 160 in AS 64900:

```
xp# bgp show community community-id 160 autonomous-system 64900
BGP table: Local router ID is 192.68.11.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop        Metric LocPrf Path
*> 192.68.20/24      172.16.20.2                  64901 i
*> 192.68.222/24     172.16.20.2                  64901 64902 i
```

bgp show peer-as

Purpose

Displays information about TCP and BGP connections to an autonomous system.

Format

bgp show peer-as <*number*> **[to-terminal|to-file]**

Mode

Enable

Description

The **bgp show peer-as** command displays information about routers in a specified autonomous system that are peered with the X-Pedition.

Parameters

peer-as <*number*>	Is the AS number of a peer autonomous system.
to-terminal	Causes output to be displayed on the terminal. This is the default.
to-file	Causes output to be saved in the file **/gatedtrc/gated.dmp**.

Restrictions

None.

Example

To display information about TCP and BGP connections to autonomous system 64901:

```
xp# bgp show peer-as 64901
group type External AS 64901 local 64900 flags <>
 peer 172.16.20.2 version 4 lcladdr (null) gateway (null)
  flags 0x20
  state 0x6 <Established>
  options 0x0 <>
  metric_out -1  preference 170  preference2 0
  recv buffer size 0  send buffer size 0
  messages in 10039 (updates 5, not updates 10034) 190863 octets
  messages out 10037 (updates 1, not updates 10036) 190743 octets
```

bgp show peer-group-type

Purpose

Displays status information about BGP peers by group.

Format

bgp show peer-group-type external|internal|routing [to-terminal|to-file]

Mode

Enable

Description

The **bgp show peer-group-type** command displays status information about BGP peers according to their group.

Parameters

external	Displays status information about external peers.
internal	Displays status information about internal peers.
routing	Displays status information about routing peers.
to-terminal	Causes output to be displayed on the terminal. This is the default.
to-file	Causes output to be saved in the file **/gatedtrc/gated.dmp**.

Restrictions

None.

Example

To display status information about external peers:

```
xp# bgp show peer-group-type external
Group    Neighbor        V    AS MsgRcvd MsgSent State
external 172.16.20.2     4 64901   10045   10044 Established
BGP summary, 1 peers in group type “external”
```

bgp show peer-host

Purpose

Displays status information about BGP peer hosts.

Format

bgp show peer-host <*ipaddr*> **received-routes|all-received-routes|advertised-routes [to-terminal|to-file]**

Mode

Enable

Description

The **bgp show peer-host** command displays information related to a specified BGP peer host. Three types of information can be displayed: routes received and accepted from a BGP peer host, all BGP routes (both accepted and rejected) from a peer host, and all routes the X-Pedition has advertised to a peer host.

Parameters

<*ipaddr*>	Is the IP address of a BGP peer host
received-routes	Displays all valid BGP routes received and accepted from the specified peer host.
all-received-routes	Displays all BGP routes (both accepted and rejected) from the specified peer host.
advertised-routes	Displays all routes the **X-Pedition** has advertised to the specified peer host.
to-terminal	Causes output to be displayed on the terminal. This is the default.
to-file	Causes output to be saved in the file **/gatedtrc/gated.dmp**.

Restrictions

When adding multiple peer hosts to a peer group, you may not connect more than one peer group to the same AS.

Examples

To display all valid BGP routes received and accepted from peer host 172.16.20.2:

```
xp# bgp show peer-host 172.16.20.2 received-routes
BGP table : Local router ID is 192.68.11.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop       Metric LocPrf Path
*> 172.16.70/24      172.16.20.2                64901 i
*> 172.16.220/24     172.16.20.2                64901 i
*> 192.68.20/24      172.16.20.2                64901 i
*> 192.68.222/24     172.16.20.2                64901 64902 i
```

To display all BGP routes (both accepted and rejected) from peer host 172.16.20.2:

```
xp# bgp show peer-host 172.16.20.2 all-received-routes
BGP table : Local router ID is 192.68.11.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop       Metric LocPrf Path
   172.16.20/24     172.16.20.2                64901 i
*> 172.16.70/24      172.16.20.2                64901 i
*> 172.16.220/24     172.16.20.2                64901 i
*> 192.68.20/24      172.16.20.2                64901 i
*> 192.68.222/24     172.16.20.2                64901 64902 i
```

Displays all routes the X-Pedition has advertised to peer host 172.16.20.2:

```
xp# bgp show peer-host 172.16.20.2 advertised-routes
BGP table : Local router ID is 192.68.11.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop       Metric LocPrf Path
*> 172.16.20/24      172.16.20.1                i
*> 192.68.11/24      192.68.11.1                i
```

bgp show regexp

Purpose

Displays the BGP routes matching the AS path regular expression.

Format

bgp show regexp <*string*> **to-terminal|to-file**

Mode

Enable

Description

The **bgp show regexp** command searches through all BGP routes that contain specified keywords belonging to an AS path. These specified keywords are the AS path regular expression upon which the search is executed. The character string can be a combination of AS numbers or names.

Some BGP character string shorthand conventions:

.	Matches any AS number
*****	Zero or more repetitions
+	One or more repetitions
?	Zero or one repetition
|	Alternation
( )	Parentheses group sub expressions

Parameters

<*string*>	A character string that specifies the regular expression. Specify an As.
to-terminal	Causes output to be displayed on the terminal. This is the default.
to-file	Causes output to be saved in the file **/gatedtrc/gated.dmp**.

Restrictions

None.

Example

To display the BGP routes starting with “64751”:

```
xp# bgp show regexp “64751 .*” to-terminal
Network          Next Hop        Metric LocPrf Path
*> 193.226.64/22 134.141.178.33              64751 6379 1 1239 11331 8338 i
```

bgp show routes

Purpose

Displays entries in the BGP routing table.

Format

bgp show routes <*ip-addr-mask*>**|all [to-terminal|to-file]**

Mode

Enable

Description

The **bgp show routes** command displays the IP address/netmask, next hop, and AS path for each BGP route.

Parameters

<*ip-addr-mask*>	Displays information about the specified route.
all	Displays information about all routes.
to-terminal	Causes output to be displayed on the terminal. This is the default.
to-file	Causes output to be saved in the file **/gatedtrc/gated.dmp**.

Restrictions

None.

Example

To display the BGP routing table:

```
xp# bgp show routes all
Proto      Route/Mask NextHop        ASPath
BGP      172.16.70/24 172.16.20.2    (64900) 64901 IGP (Id 2)
BGP     172.16.220/24 172.16.20.2    (64900) 64901 IGP (Id 2)
BGP      192.68.20/24 172.16.20.2    (64900) 64901 IGP (Id 2)
BGP     192.68.222/24 172.16.20.2    (64900) 64901 64902 IGP (Id 3)
```

bgp show summary

Purpose

Displays the status of all BGP connections.

Format

bgp show summary [to-terminal|to-file]

Mode

Enable

Description

The **bgp show summary** command displays the status of all BGP peers of the X-Pedition.

Parameters

to-terminal Causes output to be displayed on the terminal. This is the default.

to-file Causes output to be saved in the file **/gatedtrc/gated.dmp**.

Restrictions

None.

Example

To display the status of all BGP connections:

```
xp# bgp show summary
Neighbor        V    AS MsgRcvd MsgSent      Up/Down State
172.16.20.2     4 64901   10033   10031    6d23h8m1s Established
BGP summary, 1 groups, 1 peers
```

bgp show sync-tree

Purpose

Displays the BGP synchronization tree.

Format

bgp show sync-tree

Mode

Enable

Description

The **bgp show sync-tree** command displays the BGP synchronization tree. The synchronization tree is used by IBGP peers to resolve the next hop (forwarding address). It gives information about routes that are orphaned because the next hop could not be resolved.

Parameters

None.

Restrictions

None.

Examples

The following example shows the next hops for some of the routes that are not resolved (by showing orphaned routes):

```
xp# bgp show sync tree
Task BGP_Sync_64805:
      IGP Protocol: Any      BGP Group: group type Routing AS 64805

      Sync Tree (* == active, + == active with alternate, - ==
inactive with alternate:
      Orphaned routes
            Forwarding address 172.23.1.18
                  3/255 peer 172.23.1.26 preference 170
                  128.36/255.255 peer 172.23.1.26 preference 170
                  128.152/255.255 peer 172.23.1.26 preference 170
                  129.200/255.255 peer 172.23.1.26 preference 170
                  129.253/255.255 peer 172.23.1.26 preference 170
                  130.44/255.255 peer 172.23.1.26 preference 170
                  130.50/255.255 peer 172.23.1.26 preference 170
                  130.132/255.255 peer 172.23.1.26 preference 170
                  134.54/255.255 peer 172.23.1.26 preference 170
                  134.120/255.255 peer 172.23.1.26 preference 170
                  134.173/255.255 peer 172.23.1.26 preference 170
                  134.217/255.255 peer 172.23.1.26 preference 170
                  134.244/255.255 peer 172.23.1.26 preference 170
                  136.1/255.255 peer 172.23.1.26 preference 170
                  137.49/255.255 peer 172.23.1.26 preference 170
                  137.159/255.255 peer 172.23.1.26 preference 170
                  138.239/255.255 peer 172.23.1.26 preference 170
```

The following example shows the next hop for all the routes that are resolved.:

```
xp# bgp show sync-tree
Task BGP_Sync_64805:
      IGP Protocol: Any      BGP Group: group type Routing AS 64805

      Sync Tree (* == active, + == active with alternate, - ==
inactive with alternate:
      Node 3/8388608 route 3/255 metric -1 next hops 172.23.1.6 172.23.1.22
      Node 4/8388608 route 4/255 metric -1 next hops 172.23.1.6 172.23.1.22
      Node 6/8388608 route 6/255 metric -1 next hops 172.23.1.6 172.23.1.22
      Node 9.2/32768 route 9.2/255.255 metric -1 next hops 172.23.1.6 172.23.1.22
      Node 9.20/16384 route 9.20/255.255.128 metric -1 next hops 172.23.1.6 172.23.1.22
      Node 10.12.1/2 route 10.12.1/255.255.255.252 metric 0 interface
      Node 10.12.1.4/2 route 10.12.1.4/255.255.255.252 metric 2 next hop 172.23.1.22
      Node 10.200.12/128 route 10.200.12/255.255.255 metric -1 next hops 172.23.1.6 172.23.1.22
      Node 10.203.12/128 route 10.203.12/255.255.255 metric -1 next hops 172.23.1.6 172.23.1.22
      Node 10.204.12/128 route 10.204.12/255.255.255 metric -1 next hops 172.23.1.6 172.23.1.22
      Node 12/8388608 route 12/255 metric -1 next hops 172.23.1.6 172.23.1.22
      Node 12.2.19/64 route 12.2.19/255.255.255.128 metric -1 next hops 172.23.1.6 172.23.1.22
      Node 12.2.97/128 route 12.2.97/255.255.255 metric -1 next hops 172.23.1.6 172.23.1.22
      Node 12.3.123/128 route 12.3.123/255.255.255 metric -1 next hops 172.23.1.6 172.23.1.22
      Node 12.4.5/128 route 12.4.5/255.255.255 metric -1 next hops 172.23.1.6 172.23.1.22
      Node 12.4.164/128 route 12.4.164/255.255.255 metric -1 next hops 172.23.1.6 172.23.1.22
      Node 12.5.164/128 route 12.5.164/255.255.255 metric -1 next hops 172.23.1.6 172.23.1.22
      Node 12.5.172/512 route 12.5.172/255.255.252 metric -1 next hops 172.23.1.6 172.23.1.22
      Node 12.5.252/256 route 12.5.252/255.255.254 metric -1 next hops 172.23.1.6 172.23.1.22
```

bgp start|stop

Purpose

Start or stop Border Gateway Protocol (BGP).

Format

bgp start|stop

Mode

Configure

Description

The **bgp** start command starts BGP on the X-Pedition.

Parameters

start Starts BGP.

stop Stops BGP.

Restrictions

None.

bgp trace

Purpose

Set BGP trace options.

Format

bgp trace packets [detail| send| receive| group <*number-or-string*>**| peer-host** <*ipaddr*>**]|**
open [detail| send| receive| group <*number-or-string*>**| peer-host** <*ipaddr*>**]|**
update [detail| send| receive| group <*number-or-string*>**| peer-host** <*ipaddr*>**]|**
keep-alive [detail| send| receive| group <*number-or-string*>**| peer-host** <*ipaddr*>**]|**
aspath [group <*number-or-string*>**| peer-host** <*ipaddr*>**]|**
local-options [all| general| state| normal| policy| task| timer| route| group <*number-or-string*>**|**
peer-host <*ipaddr*>**]**

Mode

Configure

Description

The **bgp trace** command lets you set BGP trace options for the X-Pedition.

Parameters

packets Traces all BGP packets.

- **detail** Shows detailed information about the specified packets.
- **send** Shows the specified packets sent by the router.
- **receive** Shows the specified packets received by the router.
- **group** <*number-or-string*>
  The ID of the group for which to enable tracing.
- **peer-host** <*ipaddr*>
  The peer-host ip address for which to enable tracing. The peer-host has to be qualified by the group to which it belongs.

open Traces BGP OPEN packets, which are used to establish a peer relationship.

- **detail** Shows detailed information about the specified packets.
- **send** Shows the specified packets sent by the router.
- **receive** Shows the specified packets received by the router.

group <*number-or-string*>
The ID of the group for which to enable tracing.

peer-host <*ipaddr*>
The peer-host ip address for which to enable tracing. The peer-host has to be qualified by the group to which it belongs.

update Traces BGP update packets, which are used to pass network reachability information.

detail Shows detailed information about the specified packets.

send Shows the specified packets sent by the router.

receive Shows the specified packets received by the router.

group <*number-or-string*>
The ID of the group for which to enable tracing.

peer-host <*ipaddr*>
The peer-host ip address for which to enable tracing. The peer-host has to be qualified by the group to which it belongs.

keep-alive Traces BGP KEEPALIVE packets, which are used to verify reachability.

detail Shows detailed information about the specified packets.

send Shows the specified packets sent by the router.

receive Shows the specified packets received by the router.

group <*number-or-string*>
The ID of the group for which to enable tracing.

peer-host <*ipaddr*>
The peer-host ip address for which to enable tracing. The peer-host has to be qualified by the group to which it belongs.

aspath Traces aspath related events.

group <*number-or-string*>
The ID of the group for which to enable tracing.

peer-host <*ipaddr*>
The peer-host ip address for which to enable tracing. The peer-host has to be qualified by the group to which it belongs.

local-options Sets trace options for this protocol only. You can specify the following:

all Traces all additions, changes, and deletions to the GateD routing table.

general Activates normal and route tracing.

state Traces state machine transitions in the protocol

normal Traces normal protocol occurrences. (Abnormal protocol occurrences are always traced.)

policy Traces the application of protocol and user-specified policies to routes being imported and exported

task Traces system interface and processing associated with this protocol or peer

timer Traces timer usage by this with this protocol or peer

route Traces routing table changes for routes installed by this protocol or peer

Note: If neither the group nor peer-host is specified, tracing is enabled for all groups and peers. If the group is specified and the peer-host is not specified, tracing is enabled for that group. If both the peer-host and group are specified, tracing is enabled for that peer-host in the specified group.

group <*number-or-string*>
The ID of the group for which to enable tracing.

peer-host <*ipaddr*>
The peer-host ip address for which to enable tracing. The peer-host has to be qualified by the group to which it belongs.

Restrictions

None.

Chapter 10

cdp Commands

Command Summary

Table 10 lists the **cdp** commands. The sections following the table describe the command syntax.

Table 10. cdp commands

cdp set global-status auto-enabled |enabled |disabled
cdp set transmit-frequency <*number*>
cdp set authentication-code <*string*>
cdp set port-status port <*port-list*> **|all-ports autoenabled| enabled |disabled**
cdp show neighbors [detail]
cdp show global-info
cdp show stats
cdp show port-status port <*port-list*> **|all-ports**

cdp set global-status

Purpose

Set the global-status of all ports.

Format

cdp set global-status auto-enabled |enabled |disabled

Mode

Configure.

Description

The **cdp set global-status** command allows you to set the global status of all ports to auto-enabled, enabled, or disabled. The default **global-status** is **disabled**, indicating that if a port receives a CDP packet, the X-Pedition will **not** begin transmitting CDP hello packets from that port. This setting applies to all ports on the system. You can set individual ports differently by using *cdp set port-status* on page 183. The port-status takes precedence over the global-status.

Note: Before you can run CDP, you must enable the *task* and the *ports* on which it will run.

Parameters

None.

Restrictions

The port-status takes precedence over the global-status.

Example

Set the global-status to enabled:

```
xp(config)(cdp-set)# global-status enabled
```

cdp set transmit-frequency

Purpose

Set CDP hello packet transmission frequency.

Format

cdp set transmit-frequency <*number*>

Mode

Configure.

Description

The **cdp set transmit-frequency** command specifies the amount of time (in seconds) between successive CDP hello packet transmissions.

Note: The X-Pedition automatically sets the CDP *hold-time* to 3 times the transmission frequency. Hold-time refers to the amount of time to retain a neighbor entry after receiving the last hello packet.

Parameters

<*number*>	Enter a transmit frequency of 5-300 (inclusive).

Restrictions

None.

Example

Set the CDP hello packet transmission frequency to 50 seconds:

```
xp(config)(cdp-set)# transmit-frequency 50
```

cdp set authentication-code

Purpose

Sets authentication code for transmitted CDP packets.

Format

cdp set authentication-code <*string*>

Mode

Configure.

Description

The **cdp set authentication-code** command allows you to set the authentication code for transmitted CDP packets. The authentication code (a string) has a maximum length of 16 bytes.

Parameters

authentication-code <*string*> A character string whose maximum length is 16 bytes.

Restrictions

None.

Example

Assign an authentication code for transmitted packets:

```
xp(config)(cdp-set)# authentication-code enterasys
```

cdp set port-status

Purpose

Set port status for an individual port.

Format

cdp set port-status autoenabled| enabled |disabled port <*port-list*> **|all-ports**

Mode

Configure.

Description

The **cdp set port-status** command allows you to set the status of an individual port to autoenabled, enabled, or disabled. The default status is **disabled**, indicating that if a port receives a CDP packet, the X-Pedition will **not** begin transmitting CDP hello packets from that port. This applies to all ports on the system. The **cdp set port-status** command allows you to override the global status set for the port. (See *cdp set global-status* on page 180.)

Note: Before you can run CDP, you must enable the *task* and the *ports* on which it will run.

Parameters

port <*port-list*>	The port(s) you wish to autoenable, enable, or disable. You may use the keyword **all-ports** to set the status of all ports.

Restrictions

None.

Example

```
xp(config)(cdp-set)# port-status enable port all-ports
```

cdp show neighbors

Purpose

Show CDP neighbors.

Format

cdp show neighbors [detail]

Mode

Enable.

Description

The **cdp show neighbors** command displays the neighbor table which outlines which port the neighbor is seen on, information about the MAC address, IP address, neighbor's port that connects to your port, neighbor type, and capabilities. The **detail** optional will display more verbose information about the neighbor.

Parameters

[detail]	Display more verbose information about the neighbor.

Restrictions

None.

Example

```
xp(cdp-show)# neighbors

Following are the CDP neighbors:

Type: SF7  Network Switch running Secure Fast version 1.7 or lower
     SF8  Network Switch running Secure Fast version 1.8 or greater
     ROU  Router                          BRG  Bridge
     VLM  Cabletron VLAN Manager      NSR  Network Server(NT)
     NWS  Network Workstation(NT)     W95  Windows95
     W98  Windows98                       USR  UNIX Server
     UWS  UNIX Workstation                RWA  Roamabout wireless acc pnt

 Capabilities:
    IG   IGMP enabled on sending port
    RP   Uses RIP for routing                BG   Uses BGP for routing
    OS   Uses OSPF for routing               DV   Supports DVMRP
    1Q   Has 802.1Q support                  GV   Supports GVRP
    GM   Supports GMRP                       IS   Supports IGMP snooping
    SB   Performs source route bridging
    TB   Performs transparent bridging
    L1   Performs Level 1 functionality

Local Port      Neighbor MAC         Neighbor IP       Neighbor Port N   Type    Capabilities
--------------  -------------------  ----------------  ----------------  ------  ------------
et.2.4          00:E0:63:68:5F:F1    10.136.136.104    et.3.1            ROU     OS DV L1
et.7.15         00:01:F4:09:F2:7B    0.0.0.0           et.1.1            ROU     OS DV L1
et.13.24        00:E0:63:A2:38:6F    10.136.136.204    et.1.1            ROU     OS DV L1
gi.12.8         00:E0:63:68:12:F1    10.136.136.202    gi.5.1            ROU     OS DV L1
```

cdp show global-info

Purpose

Show CDP global information.

Format

cdp show global-info

Mode

Enable.

Description

The **cdp show global-info** command displays the current global settings for transmit frequency, holdtime, CDP status (autoenabled, enabled, disabled), and authentication code.

Restrictions

None.

Example

```
xp(cdp-show)# global-info

CDP Global Information:
   Transmit frequency is 60 seconds
   Holdtime is 180 seconds
   CDP status is Auto Enabled
   Authentication code is Default
```

cdp show stats

Purpose

Show CDP traffic.

Format

cdp show stats

Mode

Enable.

Description

The **cdp show stats** command displays the total number of CDP packets sent and received. Also displayed are error statistics for packets received with an unsupported version of CDP, the number of packets that could not be parsed, packet transmission errors, number of memory errors, and number of packets received with invalid authentication code.

Restrictions

None.

Example

```
xp(cdp-show)# stats
CDP statistics:
  Total number of CDP Packets sent                    : 0
  Total number of valid CDP Packets received          : 0
  Packets received with unsupported CDP Version       : 0
  Number of CDP packets that could not be parsed      : 0
  Packet transmission errors                          : 0
  Number of memory errors                             : 0
  Number of packets received with invalid auth code   : 0
```

cdp show port-status

Purpose

Show status of selected ports.

Format

cdp show port-status port <*port-list*> **|all-ports**

Mode

Enable.

Description

The **cdp show port-status** command displays the current status of a specific port(s) including packets sent and received, errors, current link state, and CDP status of port (autoenabled, enabled, or disabled).

Parameters

port <*port-list*>	The port(s) for which you will display the current status. You may enter a series of ports (et.1.3,et.(1,2).(4,6-8)) or enter the keyword **all-ports** to display the status of all ports.

Restrictions

None.

Example

```
xp(cdp-show)# port-status port et.1.3,et.(1,2).(4,6-8)

Port     Port Status      Pkts Sent   Pkts Received      Errors     Link State
----     -----------      ----------  ------------------ ------     ----------
et.1.3   Auto Enabled     0           0                  0          Down
et.1.4   Auto Enabled     0           0                  0          Down
et.1.6   Auto Enabled     0           0                  0          Down
et.1.7   Auto Enabled     0           0                  0          Down
et.1.8   Auto Enabled     0           0                  0          Down
et.2.4   Auto Enabled     0           0                  0          Down
et.2.6   Auto Enabled     0           0                  0          Down
et.2.7   Auto Enabled     0           0                  0          Down
et.2.8   Auto Enabled     0           0                  0          Down
```

Chapter 11

cli Commands

The **cli** commands allow you to change the behavior of the CLI in terms of command completion and command history recall.

Command Summary

Table 11 lists the **cli** commands. The sections following the table describe the command syntax.

Table 11. cli commands

cli set command completion on|off
cli set common
cli set history size <*num*>**|default|maxsize**
cli set terminal rows <*num*> **columns** <*num*>
cli show history
cli show terminal
cli terminal monitor on|off

cli set command completion

Purpose

Turn on or off command completion support.

Format

cli set command completion on|off

Mode

User and Configure

Description

The **cli set command completion** command lets you enable or disable command completion support. This command works in both User and Configure mode. When executed in Configure mode, it turns on or off command completion support for the entire system. When executed in User mode, the command affects only the current login session of the user issuing that command.

Parameters

on Turn on command completion.

off Turn off command completion.

Restrictions

None.

cli set common

Purpose

Switch to the Common CLI mode.

Format

cli set common

Mode

Enable

Description

The **cli set common** command changes the CLI environment from the Native to the Common CLI. When executed in Enable mode, this command configures the system's CLI to use the Common CLI commands and attributes. All users currently logged in and all subsequent users will switch to the Common CLI.

Note: To toggle between Common mode and Native mode, enter the **terminal set ssr** command.

Parameters

None.

Restrictions

None.

cli set history

Purpose

Modify command history recall characteristics.

Format

cli set history size <*num*>|**default**|**maxsize**

Mode

User and Configure

Description

The **cli set history** command lets you to set the size of the command history buffer. Each command stored in this buffer can be recalled without having the user type in the same, complete command again. By setting the size of this history buffer, one tells the router how many of the most recently executed commands should be stored. When the buffer is full, the oldest command is pushed out to make space for the newest command. The **cli set history** command works in both User and Configure mode. When executed in Configure mode, it sets the history size of the entire system. When executed in User mode, the command affects only the current login session of the user issuing that command.

Parameters

size A number specifying how many of the most recently executed commands should be kept. To disable history support, specify a size of 0. The **size** option can also take the following two keywords:

default Sets the history size to the system default.

maxsize Sets the history size to the system maximum.

Restrictions

None.

Examples

To set the history buffer size to 100 commands:

```
xp# cli set history size 100
```

cli set terminal

Purpose

Modify current session's terminal settings.

Format

cli set terminal [columns <*num*>**] [rows** <*num*>**]**

Mode

User

Description

The **cli set terminal** command lets you modify the terminal screen size of the current session. Specifying the number of rows available on your terminal causes the system to automatically pause when screen output fills the entire screen.

Parameters

columns Number of columns for your terminal. Minimum acceptable value is 20.

rows Number of rows for your terminal. The default row size is 25. To prevent output from pausing after one screen full, set the value to 0.

Note: Row and column settings must match the client-side terminal settings—if these settings differ, screen output may not format correctly. This is especially notable when displaying multiple pages of output.

Restrictions

None.

Examples

To set the number of rows to 50 lines:

```
xp# cli set terminal rows 50
```

cli show history

Purpose

Display the command history from the current CLI session.

Format

cli show history

Mode

User

Description

The **cli show history** command shows the commands you have issued during the current CLI session. A number is associated with each command. A command's number is useful for re-entering, modifying, or negating the command.

Note: You also can perform a command history recall by entering **!*** at any command prompt.

Parameters

None.

Restrictions

None.

cli show terminal

Purpose

Display information about the current terminal settings.

Format

cli show terminal

Mode

User

Description

The **cli show terminal** command shows information about the terminal settings. The terminal settings affect the display characteristics of your CLI session.

Parameters

None.

Restrictions

None.

cli terminal monitor

Purpose

Allows the current CLI session to receive or not receive console output.

Format

cli terminal monitor on|off

Mode

Enable

Description

The XP normally sends some system, diagnostic, and tracing messages to the management console only. The **cli terminal monitor** command allows users to configure the current Telnet or SSH CLI session to receive these messages as well.

Note: Enabling the CLI terminal monitor will not display ACL logging messages in Telnet or SSH sessions. This prevents potential session flooding with repeat ACL logging messages.

Parameters

on Turn on receipt of console output.

off Turn off receipt of console output.

Restrictions

None.

Chapter 12

comment Commands

The **comment** commands allow users to add user-defined comment lines to the active configuration file and are often used to log configuration changes and additions. To remove a line of comments, use the **negate** <*numrange*> command. The XP does not require the use of the **save active** command in conjunction with comment commands.

Command Summary

Table 12 lists the **comment** commands. The sections following the table describe the command syntax.

Table 12. comment commands

comment out <*num*>
comment in <*num*>
comment line <*num*> <*string*>
comment move <*num*>

Note: The comment commands take effect *immediately*—they do not require the **save active** command to activate them.

comment out

Purpose

Negates a command in the active configuration file, leaving the command as a comment.

Format

comment out <*num*>

Mode

Configure

Description

The **comment out** command allows you to negate a command or set of commands and leave them as a comment in the active configuration file. This is done by specifying the line number of the command or commands that you wish to negate from active configuration. The command will then be left in the active configuration file as a comment.

Note: When you use the **comment out** command to disable an **snmp set user** command, the XP will delete the passwords for that user. If you attempt to reactivate the command through the **comment in** command, the **snmp set user** command will fail and you will need to re-enter the command and create a new account for the user. To disable users and prevent them from accessing the XP through SNMP, **comment out** the user's corresponding **snmp set user-to-group** command. This will prevent you from having to recreate user accounts.

Note: The comment commands take effect *immediately*—they do not require the **save active** command to activate them.

Parameters

<*num*> Specifies the line number or numbers from the active configuration file that corresponds to the command you wish to negate.

Restrictions

None.

Example

To negate command #10 in the active configuration file:

```
xp# comment out 10
```

comment in

Purpose

Reactivates a command in the active configuration file.

Format

comment in <*num*>

Mode

Configure

Description

The **comment in** command allows you to reactivate a command that you previously negated using the **comment out** command. This is done by specifying the line number for the negated command or commands in the active configuration file.

Note: When you use the **comment out** command to disable an **snmp set user** command, the XP will delete the passwords for that user. If you attempt to reactivate the command through the **comment in** command, the **snmp set user** command will fail and you will need to re-enter the command and create a new account for the user. To disable users and prevent them from accessing the XP through SNMP, **comment out** the user's corresponding **snmp set user-to-group** command. This will prevent you from having to recreate user accounts.

Note: The comment commands take effect *immediately*—they do not require the **save active** command to activate them.

Parameters

<*num*> Specifies the line number or numbers from the active configuration file that corresponds to the command you wish to reactivate.

Restrictions

You may **comment in** only those commands that were previously commented out.

Example

To reactivate line 10 in the active configuration file:

```
xp# comment in 10
```

comment line

Purpose

Adds a comment line to the active configuration file.

Format

comment line <*num*> <*string*>

Mode

Configure

Description

The **comment line** command allows you to add a comment line to the active configuration file. This comment line will be added directly above the command(s) that is currently occupying the line number(s). Comment lines are denoted with a "C" following the line number, and a "!!" directly before the actual line of comments.

Note: The comment commands take effect *immediately*—they do not require the **save active** command to activate them.

Parameters

<*num*> Specifies the line number or numbers to which you wish to add a comment line.

<*string*> Specifies the comment line character string. Enclose the character string in quotation marks.

Restrictions

The character string must be enclosed in quotation marks.

Example

To add the comment "this is an ethernet port" to line 10 only, enter the following:

```
xp# comment line 10 "this is an ethernet port"
10C: !!this is an ethernet port
11   : interface create ip ether1 address-netmask 10.1.1.2/24 port et.4.1
```

A new line containing the comment is inserted at line 10.

To add the comment “ethernet port” to multiple lines (10-12):

```
xp(config)# comment line 10-12 “ethernet port”
10C: !!ethernet port
11   : interface create ip ether1 address-netmask 10.1.1.2/24 port et.4.1
       !
12C : !!ethernet port
13   : interface create ip ether2 address-netmask 20.1.1.2/24 port et.4.2
       !
14C : !!ethernet port
15   : interface create ip ether3 address-netmask 30.1.1.2/24 port et.4.3
```

Note: The **interface create** commands previously occupied lines 10, 11, and 12. With the comments added, these commands now occupy 11, 13, and 15.

comment move

Purpose

Moves a comment line from one line number to another line number in the active configuration file.

Format

comment move <*num*>

Mode

Configure

Description

The **comment move** command allows you to move a comment line or range of comment lines from one line number to another line number within the active configuration file.

Note: The comment commands take effect *immediately*—they do not require the **save active** command to activate them.

Parameters

<*num*> Specify the current line number(s) before the comma, and specify the new line number(s) after the comma.

The format is as follows: <*current line number(s)*, *new line number(s)*>

Restrictions

Do not attempt to move actual commands. When moving a range of comment lines, the line number ranges must be the same size.

Example

To move the comments in lines 1-2 to lines 7-8:

```
xp# comment move 1-2,7-8
```

Chapter 13

configure Command

The **configure** command exits Enable mode and places the CLI session in Configure mode. From Configure mode, users may set and change X-Pedition parameters.

Purpose

Enters the CLI's Configure mode.

Format

configure

Mode

Enable

Description

Configure mode provides the capabilities to configure all features and functions on the X-Pedition. These include router configuration, access control lists and spanning tree. To enter Configure mode, enter the command **config** from Enable mode.

Note: As mentioned previously, up to four Telnet sessions can be run simultaneously on the X-Pedition. All four sessions can be in Configure mode at the same time, so you should consider limiting access to the X-Pedition to authorized users.

The Configure mode command prompt consists of the X-Pedition name followed by **(config)** and a pound sign (#):

```
xp(config)#
```

To exit Configure mode and return to Enable mode, either type **exit** and press Return, or press Ctrl+Z.

Parameters

None.

Restrictions

To enter Configure mode, you must already be in Enable mode.

Chapter 14

copy Command

Format

copy {active| rcp-server| scratchpad| startup| tftp-server| <*[device:]filename*>**|** <*url*>**}**
to {active| backup-CM| ethers| rcp-server| scratchpad| startup| tftp-server| <*[device:]fileame*>**|** <*url*>**}**

Mode

Enable

Description

The **copy** command is used primarily to transfer configuration information—to copy non-configuration files, users should generally employ the **file copy** command. Users can copy configuration information between the X-Pedition and external hosts using protocols such as TFTP or RCP. Within the X-Pedition, users can copy configuration information between the file system, the scratchpad (configuration database), the active (running) configuration, or the Startup configuration. If the X-Pedition is operating in a dual-CM environment, users may copy the startup configuration of the primary Control Module to the secondary Control Module. The **copy** command also allows users to make backup copies of a configuration file.

Note: When copying an external file that contains configuration commands, each command must exist on a single line. If a command breaks over multiple lines, the X-Pedition will read each line as a separate entry and an error will occur.

Multi-line command	Result
system set login-banner “This is the login banner”	**system set login-banner** “This is

Parameters

active Copy information to the active configuration database (the system configuration currently running). Users may copy information into the active configuration from the scratchpad only.

backup-CM
Copy the startup configuration from the Primary control module to the Backup control module. A user may specify the **backup-CM** parameter only as the destination—and only with **startup** as the source. (The **save startup** and **copy** <*source_parameter*> **to startup** commands automatically copy the Primary control module's startup configuration to the Backup Control Module.)

ethers The file located on the PCMCIA card that contains IP/MAC address pairings for reverse ARP queries. Each line within the **ethers** file contains a MAC address and IP address pair. The MAC address and IP address must be in the same format as they would appear on a CLI command (e.g., "00:0d:12:34:56:78 10.136.4.9"). The **ethers** destination parameter is used only with the **tftp-server** or the **tftp** type <*url*> source parameters.

rcp-server Downloads a file from or uploads a file to an RCP server.

scratchpad Copy configuration changes from the scratchpad.

startup Copies the Startup configuration information stored in the control module's NVRAM.

tftp-server Downloads a file from or uploads a file to a TFTP server.

<*[device:]filename*>
Represented in the CLI help as a character string, this parameter specifies the name of a file on the X-Pedition's local file system (NVRAM or PCMCIA Flash Module). The *device:* is optional and may be one of the following:

bootflash: The Control Module's NVRAM.

slot0: The PCMCIA Flash Module in slot 0 (the upper slot).

Note: The X-Pedition supports PCMCIA Flash Modules obtained from Enterasys Networks only. For information regarding the PCMCIA Virtual File systems VFS1 and VFS2, see the *Enterasys X-Pedition User Reference Manual*.

<*url*> Specify a **tftp** or **rcp** URL:

tftp://<*hostname*>/<*path*>

rcp://<*username*>**@**<*hostname*>/<*path*>

Note: Because a URL refers to an absolute path, you must include a backslash within the path name for both the **tftp** and **rcp** URLs, as well as between the hostname and path.

Note: Attention UNIX Users. In order to specify a tftp URL, you must first create the file on the tftp server using the same filename you will to use for the copy command. You

must also make certain that all permissions for the file and any directories within its path are world writable.

Restrictions

- The X-Pedition does not allow some source and destination pair combinations. Typically, users cannot specify the same location type for both source and destination (i.e., a user may not copy from one TFTP server directly to another TFTP server, or copy from scratchpad to scratchpad).
- Users may copy information into the active configuration (the system configuration currently running) from the scratchpad only.
- A user may specify the **backup-CM** parameter only as the destination—and only with **startup** as the source.
- The X-Pedition supports PCMCIA Flash Modules obtained from Enterasys Networks only.
- Because a URL refers to an absolute path, you must include a backslash within the path name for both the **tftp** and **rcp** URLs, as well as between the hostname and path.

The following chart displays valid source and destination combinations:

Table 13. Valid Source and Destination Parameters for the Copy command

Source	Destination
active	**rcp-server** | **startup** | **tftp-server** | <*[device:]filename*> | <*url*>
rcp-server	**scratchpad** | **startup** | <*[device:]filename*>
scratchpad	**active** | **rcp-server** | **startup** | **tftp-server** | <*[device:]filename*> | <*url*>
startup	**backup-CM** | **rcp-server** | **scratchpad** | **tftp-server** | <*[device:]filename*> | <*url*>
tftp-server	**ethers** | **scratchpad** | **startup** | <*[device:]filename*>
<*[device:]filename*>	**rcp-server** | **scratchpad** | **startup** | **tftp-server** | *device:]filename*> | <*url*>
<*url*>	**scratchpad** | **startup** | <*[device:]filename*>

Examples

To copy configuration information from the scratchpad to the active database (causing changes to take immediate effect), enter the following command from the CLI.

```
xp# copy scratchpad to active
```

To copy the file **config.john** to the PCMCIA card, slot0:config.debi:.

```
xp# copy config.john to slot0:config.debi
```

To copy the Startup configuration to a TFTP server for backup purposes, enter the following command. The CLI will then prompt the user for the TFTP server's IP address or hostname and the filename:

```
xp# copy startup to tftp-server
TFTP server? 10.136.11.1
Destination filename? my_startup.cfg
```

To copy a previously saved configuration from a TFTP server to the Startup configuration, enter the following command. Note the use of a URL to specify the TFTP server and the filename.

```
xp# copy tftp://10.1.2.3/backup/config.org to startup
```

To copy the active configuration to a remote server using RCP, enter the following command. Note that this example uses a URL to specify the RCP user name, server, and filename.

```
xp# copy active to rcp://john@server1/config/config.dec25
```

To copy the startup configuration from the Primary control module to the Backup control module:

```
xp# copy startup to backup-CM
```

Chapter 15

dhcp Commands

The **dhcp** commands allow you to configure *scopes* (sets of IP address pools and network parameters) that are to be used by Dynamic Host Configuration Protocol (DHCP) clients and apply them to interfaces on the X-Pedition.

Command Summary

Table 14 lists the **dhcp** commands. The sections following the table describe the command syntax.

Table 14. dhcp commands

dhcp <*scope*> **attach superscope** <*superscope*>
dhcp <*scope*> **define parameters** <*parameter*> <*value*>
dhcp <*scope*> **define pool** <*ip-range*>
dhcp <*scope*> **define static-ip** <*ipaddr*> **mac-address** <*macaddr*> [<*parameter*> <*value*>]
dhcp flush
dhcp global set commit-interval <*hours*>
dhcp global set lease-database <*url*>
dhcp show binding [**active**| **expired**| **static**| <*ipaddr*>**]**
dhcp show num-clients

dhcp attach superscope

Purpose

Creates a group of scopes that share a common interface.

Format

dhcp <*scope*> **attach superscope** <*superscope*>

Mode

Configure

Description

The **dhcp attach superscope** command allows you to create a "superscope," a group of scopes that share a common physical interface. For example, you can define and group together scopes for different subnets that are accessed through a single port or VLAN.

Parameters

<*scope*>	The name of a scope that was previously configured with the **dhcp define** commands.
<*superscope*>	The name of the group to which the specified scope is being attached.

Restrictions

None.

Examples

Consider the following example where the scopes 'client1' and 'client2' exist on the same interface. To group scopes 'client1' and 'client2' into the superscope 'allclients':

```
xp(config)# dhcp client1 attach superscope allclients
xp(config)# dhcp client2 attach superscope allclients
```

dhcp define parameters

Purpose

Define parameters to be used by DHCP clients.

Format

dhcp <*scope*> **define parameters** <*parameter*> <*value*>...

Mode

Configure

Description

The **dhcp define parameters** command allows you to define a set of parameters that are to be used by clients when DHCP is enabled. The client uses these parameters to configure its network environment, for example, the default gateway and DNS domain name. The DHCP server on the X-Pedition supports parameters used by Windows 95/98/NT and MacOS clients.

Parameters

<*scope*>
The name that refers to this set of client parameters.

<*parameter*> <*value*>
You can specify one or more of the following client parameters and values:

address-mask	**(Required) Specifies the address and netmask of the scope's subnet.**

Note: The **address-mask** parameter is *required* and must be defined *before* any other client parameters are specified.

broadcast	Specify the broadcast address.
bootfile	Specify the client's boot filename.
dns-domain	Specify the DNS domain name.
dns-server	Specify the IP address of the DNS server.
gateway	Specify the IP address of the default gateway.
lease-time	Specify how long, in hours, the lease is valid. (A lease is the amount of time that an assigned IP address is valid for a client system.)
lease-time-in-minutes	Specify how long (in minutes) the lease will remain valid.

netbios-name-server	Specify the IP address of the NetBIOS name server or WINS server.
netbios-node-type	Specify the NetBIOS node type of the client.
netbios-scope	Specify the NetBIOS scope of the client.
authoritative	Selecting this option causes the router to send DHCPNAK messages if the IP address specified by a DHCP request is not valid. If you do not select this option, the router will not send a DHCPNAK message.

Restrictions

None.

Examples

The following command configures a group of network parameters for the scope ‘finance’:

```
xp(config)# dhcp finance define parameters address-netmask 10.33.0.0/16 dns-server 10.3.2.1 dns-domain acme.com gateway 10.33.1.1 netbios-node-type b-node lease-time 90 netbios-name-server 10.33.44.55 netbios-scope acme-finance
```

dhcp define pool

Purpose

Define a pool of IP addresses to be used by DHCP clients.

Format

dhcp <*scope*> **define pool** <*ip-range*>

Mode

Configure

Description

The **dhcp define pool** command allows you to define a pool of IP addresses that can be used by DHCP clients. An IP address pool, along with a set of parameters defined with the **dhcp define parameters** command, make up a DHCP "scope".

Parameters

<*scope*>	A name that refers to the specified pool of addresses.
<*ip-range*>	The range of IP addresses to be used by the clients. Use a hyphen (-) to designate the range. If you have more than one pool of IP addresses to specify or if the addresses are not contiguous, specify additional addresses using multiple **dhcp define pool** commands.

Restrictions

None.

Examples

To specify the addresses between 10.1.1.1 to 10.1.1.20 as the pool of IP addresses for the scope 'clients':

```
xp(config)# dhcp clients define pool 10.1.1.1-10.1.1.20
```

To specify two separate pools of IP addresses for the scope 'clients':

```
xp(config)# dhcp clients define pool 10.1.1.1-10.1.1.20
xp(config)# dhcp clients define pool 10.1.1.30-10.1.1.40
```

dhcp define static-ip

Purpose

Define a static IP address for a specific MAC address.

Format

dhcp <*scope*> **define static-ip** <*ipaddr*> **mac-address** <*macaddr*> [<*parameter*> <*value*>...]

Mode

Configure

Description

The **dhcp define static-ip** command allows you to configure a static IP address for a specific MAC address. For example, you can define a static IP address for a printer's MAC address to ensure that the printer always receives the same IP address from the DHCP server. Static IP addresses can be used for BOOTP clients as well as DHCP clients.

If you want a single MAC address to have different static IP addresses, depending upon which subnet or interface the machine is on, you can configure different scopes with different IP addresses that map to the same MAC address.

A client configured for a static IP address inherits the client parameters that are configured for the scope. If you want to configure a specific group of parameters for a static IP address, specify those parameters with the **dhcp define static-ip** command.

Parameters

<*scope*> A name that refers to the specified static IP address.

<*ipaddr*> The static IP address.

<*macaddr*> The MAC address to which the specified static IP address is to be mapped.

<*parameter*> <*value*>
Specifies the client parameters and values for this static IP address. You can specify one or more of the following client parameters and values:

broadcast Specify the broadcast address.

bootfile Specify the client's boot filename.

dns-domain Specify the DNS domain name.

dns-server Specify the IP address of the DNS server.

gateway	Specify the IP address of the default gateway.
lease-time	Specify how long, in minutes, the lease is valid. (A lease is the amount of time that an assigned IP address is valid for a client system.)
netbios-name-server	Specify the IP address of the NetBIOS name server or WINS server.
netbios-node-type	Specify the NetBIOS node type of the client.
netbios-scope	Specify the NetBIOS scope of the client.

Restrictions

None.

Examples

To specify a static IP address 10.1.44.55 to the MAC address 08:00:20:12:34:56 for the scope ‘servers’:

```
xp(config)# dhcp servers define static-ip 10.1.44.55 mac-address 08:00:20:12:34:56
```

To specify a static IP address 10.1.44.55 to the MAC address 08:00:20:12:34:56 for the scope ‘servers’ and give it a specific default gateway address:

```
xp(config)# dhcp servers define static-ip 10.1.44.55 mac-address 08:00:20:12:34:56 gateway 10.1.1.2
```

To define two different scopes (‘public’ and ‘private’) with two different static IP addresses (10.1.44.55 and 10.2.10.23) that map to the MAC address 08:00:20:12:34:56:

```
xp(config)# dhcp public define static-ip 10.1.44.55 mac-address 08:00:20:12:34:56
xp(config)# dhcp private define static-ip 10.2.10.23 mac-address 08:00:20:12:34:56
```

dhcp flush

Purpose

Forces the DHCP server to update its lease database.

Format

dhcp flush

Mode

Enable

Description

The DHCP server normally updates its lease database at the intervals specified with the **dhcp global set commit-interval** command. While the DHCP server is running, you can force the server to immediately update its lease database by using the **dhcp flush** command.

Parameters

None.

Restrictions

None.

dhcp global set commit-interval

Purpose

Configure the intervals at which the DHCP server updates the lease database.

Format

dhcp global set commit-interval <*minutes*>

Mode

Configure

Description

After each client transaction, the DHCP server does not immediately update the information in the lease database. Lease update information is stored in flash memory and flushed to the database at certain intervals. You can use the **dhcp global set commit-interval** command to specify this interval.

Note: Writing to flash memory can be time-consuming if there are many clients on the network.

Parameters

commit-interval <*hours*>
The interval, in hours, that the DHCP server updates the lease database. The default value is 1 hour. You can specify a value between 1-48.

Restrictions

None.

Examples

To configure the DHCP server to update the lease database once every 2 hours:

```
xp(config)# dhcp global set commit-interval 2
```

dhcp global set lease-database

Purpose

Specify a TFTP or RCP server where the lease database is backed up.

Format

dhcp global set lease-database <*url*>

Mode

Configure

Description

By default, the X-Pedition stores the clients' lease information (the lease database) in its flash memory. You can use the **dhcp global set lease-database** command to specify a TFTP or RCP server where the lease database is to be periodically backed up.

Parameters

lease-database <*url*>
The TFTP or RCP server where the lease-database is to be backed up.

Restrictions

None.

Examples

To configure the lease database to be on a TFTP server (10.50.89.88) with the file name 'lease-db':

```
xp(config)# dhcp global set lease-database tftp://10.50.89.88/lease-db
```

To configure the lease database to be on an RCP server (10.50.89.89) with the user name 'john' and the file name 'lease-db':

```
xp(config)# dhcp global set lease-database rcp://john@10.50.89.89/lease-db
```

dhcp show binding

Purpose

Display information from the lease database.

Format

dhcp show binding [active| expired| static| <*ipaddr*>**]**

Mode

Enable

Description

The **dhcp show** command displays information from the lease database. If you do not specify any parameters, the DHCP server displays the entire lease database.

Parameters

active	Displays currently active leases only.
expired	Displays expired leases only.
static	Displays leases with static IP address assignments only.
<*ipaddr*>	IP address. Will display only the binding for the specified IP address.

Restrictions

None.

Examples

To display information from the lease database:

```
xp# dhcp show binding
IP address      Hardware Address     Lease Expiration    Type
------------    ------------------   ------------------  ------
10.20.1.22      00:40:05:41:f1:2d    2003-05-24 17:45:06 dynamic
10.20.1.23      00:00:b4:b1:29:9c    2003-05-24 17:45:04 dynamic
10.20.1.21      00:00:b4:b0:f4:83    2003-05-24 17:45:01 dynamic
10.20.1.20      00:80:c8:e1:20:8a    2003-05-24 09:24:30 dynamic
10.30.7.9       08:00:20:11:22:33                        static
10.30.7.44      08:00:20:44:55:66                        static
```

dhcp show num-clients

Purpose

Display the number of allocated bindings for the DHCP server and the maximum number allowed.

Format

dhcp show num-clients

Mode

Enable

Description

The **dhcp show** command displays the number of allocated bindings for the DHCP server and the maximum number allowed.

Parameters

None.

Restrictions

None.

Examples

To display information:

```
xp# dhcp show num-clients
15 current clients (253 maximum)
```

Chapter 16

diff Command

The **diff** command compares the active configuration with the specified configuration file.

Format

diff <*filename*>**|startup**

Mode

Configure

Description

The **diff configuration** command compares the active configuration with the specified configuration file.

Parameters

<*filename*>	Name of a configuration file.
startup	The Startup configuration file.

Restrictions

None.

Example

To compare the active configuration with the Startup configuration file:

```
xp# diff startup
```

Chapter 17

dvmrp Commands

The **dvmrp** commands let you configure and display information about Distance Vector Multicast Routing Protocol (DVMRP) interfaces.

Notes

- Because DVMRP and PIM-SM run in separate processes on the X-Pedition, current IGMP functionality may be used only with DVMRP. PIM-SM must use a separate group of commands called “PIM IGMP.”
- The X-Pedition does not allow users to enable DVMRP and PIM simultaneously. If a user attempts to enable DVMRP and PIM at the same time, one of the following messages will appear:

  %CLI-E-NODVMRPFAC, This command cannot be used when PIM has been configured
  %CLI-E-NOPIMFAC, This command cannot be used when IGMP or DVMRP has been configured.

  To switch between PIM and DVMRP you must remove the protocol's start command from the startup configuration and restart the router.

Command Summary

Table 15 lists the **dvmrp** commands. The sections following the table describe the command syntax.

Table 15. dvmrp commands

dvmrp accept|noaccept route <*IPaddr/mask*> **[exact] [interface** <*IPaddr*> **[router** <*IPaddr*>**]]**
dvmrp advertise|noadvertise route <*IPaddr/mask*> **[exact] [interface** <*IPaddr*>**]**
dvmrp create tunnel <*name*> **local** <*IPaddr*> **remote** <*IPaddr*>

Table 15. dvmrp commands (Continued)

dvmrp enable no-pruning
dvmrp enable interface <*IPaddr*>|<*interface-name*>|<*tunnel-name*>
dvmrp set interface <*IPaddr*>|<*hostname*> **[metric** <*num*>**] [neighbor-timeout** <*seconds*>**] [prunetime** <*seconds*>**] [rate** <*num*>**] [scope** <*IPaddr/mask*>**] [threshold** <*num*>**] [force-leaf]**
dvmrp set protocol route-report-interval <*number*>
dvmrp show interface [<*IPaddr*>**]**
dvmrp show routes host <*IPaddr*>**|interface** <*IPaddr*>**|net** <*netaddr*>**|router** <*IPaddr*>
dvmrp show rules
dvmrp start

dvmrp accept route

Purpose

Specifies routes to be accepted from DVMRP neighbor routers.

Format

dvmrp accept|noaccept route <*IPaddr/mask*> **[exact] [interface** <*IPaddr*> **[router** <*IPaddr*>**]]**

Mode

Configure

Description

The **dvmrp accept route** command allows you to specify particular routes that can be learned from DVMRP neighbors.

A route is always accepted from a DVMRP neighbor unless you use the **dvmrp noaccept route** to prevent it from being accepted. You can use the **dvmrp accept route** command along with the **dvmrp noaccept route** command to filter the routes accepted from DVMRP neighbor routers.

Parameters

accept
Allows the specified route to be accepted from DVMRP neighbor routers.

noaccept
Prevents the specified route from being accepted from DVMRP neighbor routers.

route <*IPaddr/mask*>
Is the IP address and mask of the route prefix to be accepted.

exact
Causes only routes exactly matching the prefix to be accepted.

interface <*ipAddr*>
Is the IP address of the interface to which you are applying this filter.

router <*IPaddr*>
Is the IP address of a DVMRP neighbor router.

Restrictions

None.

Examples

To cause the X-Pedition to accept only prefix 20.30.40.0/24, and filter out all other routes:

```
xp(config)# dvmrp noaccept route 0/0 interface customer1
xp(config)# dvmrp accept route 20.30.40.0/24 interface customer1
```

If interface customer1 breaks subnet 20.30.40.0/24 into smaller subnets, you can filter out routes from these subnets with the following commands:

```
xp(config)# dvmrp noaccept route 0/0 interface customer1
xp(config)# dvmrp accept route 20.30.40.0/24 interface customer1 exact
```

dvmrp advertise route

Purpose

Specifies routes to be advertised to DVMRP neighbor routers.

Format

dvmrp advertise|noadvertise route <*IPaddr/mask*> **[exact] [interface** <*IPaddr*>**]**

Mode

Configure

Description

The **dvmrp advertise route** command allows you to specify particular routes that can be advertised to DVMRP neighbors. A route is always advertised to a DVMRP neighbor unless you use the **dvmrp noadvertise route** command to prevent it from being advertised. You can use the **dvmrp advertise route** command along with **dvmrp noadvertise route** to filter the routes advertised to DVMRP neighbor routers.

Parameters

advertise
Allows the specified route to be advertised to DVMRP neighbor routers.

noadvertise
Prevents the specified route from being advertised to DVMRP neighbor routers.

route <*IPaddr/mask*>
Is the IP address and mask of the route prefix to be advertised.

exact
Causes only routes exactly matching the prefix to be advertised.

interface <*ipAddr*>
Is the IP address of the interface to which you are applying this filter.

Restrictions

None.

Examples

To prevent route 10.0.0.0/8 from being advertised on interface mbone (all other routes are advertised):

```
xp(config)# dvmrp noadvertise route 10/8 interface mbone
```

To advertise only route 20.20.20.0/24 to its neighbors on interface mbone:

```
xp(config)# dvmrp noadvertise route 0/0 interface mbone
xp(config)# dvmrp advertise route 20.20.20.0/24 interface mbone
```

dvmrp create tunnel

Purpose

Creates a DVMRP tunnel.

Format

dvmrp create tunnel <*name*> **local** <*ipAddr*> **remote** <*ipAddr*>

Mode

Configure

Description

The **dvmrp create tunnel** command creates a tunnel used to pass multicast traffic through a unicast network that resides between DVMRP clouds. As multicast frames exit the DVMRP *source* cloud, they are *encapsulated* in a unicast packet. When frames enter the *destination* cloud, the unicast packets are *un-encapsulated* and returned to the native multicast format. The X-Pedition control module encapsulates and un-encapsulates each packet—not the hardware ASICs (this can degrade overall performance and drop frames from the stream).

If the sum of the traffic crossing a DVMRP tunnel exceeds approximately 8 mbps, packet loss may occur. CPU-bound traffic such as learning flows, ARPs, and routing updates will lower system performance. This does not apply to packets routed by an X-Pedition within a DVMRP cloud.

Parameters

<*name*>	Name of this DVMRP tunnel.
local <*ipAddr*>	IP address of the local end point of this tunnel.
	Note: The local IP address must already be configured on the X-Pedition.
remote <*ipAddr*>	IP address of the remote end point of this tunnel.

Restrictions

Note: Use caution when creating DVMRP tunnels with the **dvmrp create tunnel** command.

- Tunnels use unicast routing principles. Make sure a route exists between the tunnel source and destination (***local*** <*ipAddr*> and ***remote*** <*ipAddr*>) you specify.
- An IP interface has to exist before a tunnel can be created from it.

  Note: A good way to confirm that a tunnel exists is to ping the other end of the tunnel.

- Tunnels cannot be created between two endpoints (i.e., on the same subnet).
- A maximum of eight tunnels are allowed.

Example

To create a DVMRP tunnel called *tun12* between 10.3.4.15 (the local end of the tunnel) and 10.5.3.78 (the remote end of the tunnel):

```
xp(config)# dvmrp create tunnel tun12 local 10.3.4.15 remote 10.5.3.78
```

dvmrp enable no-pruning

Purpose

Disables DVMRP pruning.

Note: Pruning is enabled by default. The current DVMRP specification requires pruning capability. Unless you have a good reason for disabling pruning, Enterasys Networks recommends that you leave it enabled.

Format

dvmrp enable no-pruning

Mode

Configure

Description

Disable DVMRP pruning.

Parameters

None.

Restrictions

None.

dvmrp enable interface

Purpose

Enables DVMRP on an interface.

Format

dvmrp enable interface <*ipAddr/name*>|<*tunnel-name*>

Mode

Configure

Description

The **dvmrp enable interface** command enables DVMRP on the specified interface.

Parameters

<*ipAddr/name*>|<*tunnel-name*>
IP address or tunnel name of the interface on which you are enabling DVMRP.

•If you are enabling DVMRP on an interface that does not have a tunnel, specify its name or IP address.

•If you are enabling DVMRP on an interface that has a tunnel, specify the tunnel name.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

- The Control Module's en0 interface (labeled "10/100 Mgmt") is never used for multicast traffic—it is a management port only.
- DVMRP does not run on multiple IP subnets if created on an interface. Currently, the X-Pedition automatically picks up the first subnet to run DVMRP on it. However any one particular subnet can be picked up by enabling it. But before doing that, no subnet should already be enabled on that interface. The X-Pedition supports a maximum of 64 DVMRP and IGMP interfaces.

  Note: The **igmp enable interface** command has a similar restriction of using only one subnet.

- Because DVMRP and PIM-SM run in separate processes on the X-Pedition, current IGMP functionality may be used only with DVMRP. PIM-SM must use a separate group of commands called “PIM IGMP.”
- The X-Pedition does not allow users to enable DVMRP and PIM simultaneously. If a user attempts to enable DVMRP and PIM at the same time, one of the following messages will appear:

  %CLI-E-NODVMRPFAC, This command cannot be used when PIM has been configured
  %CLI-E-NOPIMFAC, This command cannot be used when IGMP or DVMRP has been configured.

  To switch between PIM and DVMRP you must remove the protocol's start command from the startup configuration and restart the router.

Examples

To enable DVMRP on the IP interface with IP address 10.50.78.2:

```
xp(config)# dvmrp enable interface 10.50.78.2
```

To enable tunnel tun12:

```
xp(config)# dvmrp enable interface tun12
```

dvmrp set interface

Purpose

Configures various DVMRP parameters on an interface.

Format

dvmrp set interface <*IPaddr*>|<*hostname*> **[metric** <*num*>**] [neighbor-timeout** <*seconds*>**] [prunetime** <*seconds*>**] [rate** <*num*>**] [scope** <*IPaddr/mask*>**] [threshold** <*num*>**] [force-leaf]**

Mode

Configure

Description

The **dvmrp set interface** command sets DVMRP parameters on an IP interface.

Parameters

<*ipAddr/name.*
IP address or name of the interface on which you are configuring DVMRP parameters.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

metric <*num*>
The metric (cost) of this interface. Specify a number in the range 1 – 16. The default is 1. Normally you should not change this setting unless the network topology requires it.

neighbor-timeout <*num*>
The number of seconds after which the X-Pedition will consider the neighbor to be down. Specify a number in the range 40 – 400. The default is 35.

Note: If you have old routers, this value should be increased to accommodate them; older routers do not send probes or route updates at 40-second intervals.

prunetime <*seconds*>
The multicast prunetime of this interface. Specify a number in the range 300 – 7200. The default is 7200 seconds (two hours).

rate <*num*>
The multicast rate of this interface in kbps. Specify a number in the range 1 – 10000. The default is 500.

Note: The option applies only to tunnels.

scope <*IPaddr/mask*>
The multicast scope of this interface. The purpose of this option is to disallow the groups specified by a scope from being forwarded across an interface. This option therefore is a filtering mechanism. The threshold and the scope are two common mechanisms for implementing local simple filtering of a multicasting data.

Specify an IP address and network mask. Example: 239.0.0.0/8.

threshold <*num*>
The multicast threshold of this interface. The purpose of this option is to allow forwarding of a packet on a multicast interface only if the packet's threshold is at least the configured value. The threshold and the scope are two common mechanisms for implementing local simple filtering of a multicasting data.

Specify a number in the range 1 – 255. The default is 1.

force-leaf
Per RFC 1075, DVMRP-enabled routers flood out their downstream interfaces for extended periods of time (up to 2.5 minutes) following the expiration of their prune timers. However, the force-leaf option allows users to override this behavior and minimize interface flooding.

Notes:

– Because the force-leaf option violates RFC 1075, Enterasys Technical Support does NOT support this capability and will not address bug fixes or enhancement requests related to it.

– If a client has two DVMRP routers as possible sources to a particular group, using this command will increase the failover time if the forwarding router loses its connection to the client.

Restrictions

None.

Examples

To configure the interface 10.50.89.90 to have a metric of 5 and a threshold of 16:

```
xp(config)# dvmrp set interface 10.50.89.90 metric 5 threshold 16
```

dvmrp set protocol

Format

dvmrp set protocol route-report-interval <*number*>

Mode

Configure

Description

This command allows users to specify the frequency with which to send a DVMRP route report to all neighboring routers.

Parameters

route-report-interval <*number*>
Specify how often (in seconds) to send a route report. By default, this value is 60 seconds.

Restrictions

- The time interval should be a multiple of 5—otherwise, the firmware will change the value internally but not reflect it in the actual configuration.
- The route-report-interval uses normal rounding rules. If a user enters a value of 27 seconds, the actual route-report-interval used will be 25 seconds. If a user enters a value of 28 seconds, the route-report-interval used will be 30 seconds.

Examples

To set the route report interval to every 30 seconds (without rounding the value), enter the following:

```
xp(config)# dvmrp set protocol route-report-interval 30
```

dvmrp show interface

Purpose

Displays DVMRP interfaces.

Format

dvmrp show interface [<*IPaddr*>]

Mode

Enable

Description

The **dvmrp show interface** command displays the state of an interface running DVMRP, along with other neighbor-related information. Neighbors are displayed with their DVMRP version and capability flags and Generation IDs; this information can help in debugging. If rules are in effect for an interface, they are indicated by ExportPol or the ImportPol flags.

Parameters

<*IPaddr*> Displays DVMRP information for the specified interface.

Restrictions

None.

Examples

Here is an example of the **dvmrp show interface** command.

```
xp# dvmrp show interface
Address: 10.50.1.1          Subnet: 10.50.1/24       Met: 1   Thr: 1
Name   : pc                 State: Dn   Igmp  Dvmrp

Address: 207.135.89.10      Subnet: 207.135.89.0/27   Met: 1   Thr: 1
Name   : corp               State: Up   Igmp  Dvmrp Querier  ExportPol
Peer   : 207.135.89.1       Version:   3.255        Flags:0xe  GID: 0x31a

Address: 10.55.89.101       Subnet: 10.55.89/24      Met: 1   Thr: 1
Name   : lab                State: Up   Dvmrp
Peer   : 10.55.89.100       Version:   3.255        Flags:0xe  GID: 0x179

Address: 207.135.89.10      Remote: 207.137.137.1    Met: 1   Thr: 1  Rate: 1000
Name   : mbone              State: Tunnel  Up   Dvmrp ExportPol
Peer   : 207.137.137.1      Version:   3.8          Flags:0xe  GID: 0x6c19d135
```

dvmrp show routes

Purpose

Displays DVMRP unicast routing table.

Format

dvmrp show routes host <*IPaddr*>|**interface** <*IPaddr*>|**net** <*netaddr*>|**router** <*IPaddr*> **subordinates**|**permission**

Mode

Enable

Description

The **dvmrp show routes** command displays the contents of DVMRP unicast routing table.

DVMRP routes show the topology information for the internet multicasting sites. It is independent of IP unicast routing table or protocol. In this table, the information is presented about a address prefix (in form of network-address/network-mask length), the interface and the uplink (parent) router through which this subnet can be reached. This table also shows information about any routers/interfaces which consider this router as their uplink (that is, those routers which depend on this router if traffic were to originate from this subnet). These routers/interfaces are shown as children of the parent router.

Note: The **dvmrp show routes** command can search on the basis of subnet and on the basis of those routes whose parent is a particular interface and/or a particular router.

Note: This command only shows DVMRP routes and not information about current multicast sessions. For information about current multicast sessions, use the **multicast show mroutes** command.

Parameters

host <*IPaddr*>	Displays the route to the specified uplink host address.
interface <*IPaddr*>	Displays the interface address of the specified uplink interface.
net <*netaddr*>	Displays the route to the specified prefix (or subnets falling within the prefix).
router <*IPaddr*>	Displays the route to the specified router.
subordinates	Displays the downstream routers list.
permissions	Indicates whether a route is affected by any rules. Routes marked NoAdv are not advertised.

Restrictions

None.

Examples

To display DVMRP routes offered by the next-hop router 207.137.137.1:

```
xp# dvmrp show routes router 207.137.137.1
DVMRP Routing Table (4232 routes,   8 hold-down-routes)
Net: 128.119.3.16/29        Gateway: 207.137.137.1     Met:  9   Age:  35
Parent: mbone             Children: corp
                                    lab

Net: 128.119.3.8/29         Gateway: 207.137.137.1     Met:  9   Age:  35
Parent: mbone             Children: corp
                                    lab

Net: 209.12.162.16/28       Gateway: 207.137.137.1     Met: 26   Age:  35
Parent: mbone             Children: corp
                                    lab

Net: 208.197.171.112/28     Gateway: 207.137.137.1     Met:  7   Age:  35
Parent: mbone             Children: corp
                                    lab

Net: 208.151.215.240/28     Gateway: 207.137.137.1     Met:  7   Age:  35
Parent: mbone             Children: corp
                                    lab

Net: 208.151.215.192/28     Gateway: 207.137.137.1     Met:  7   Age:  35
Parent: mbone             Children: corp
                                    lab

Net: 208.151.215.96/28      Gateway: 207.137.137.1     Met:  7   Age:  35
Parent: mbone             Children: corp
```

To show non-advertised routes on interface lab:

```
xp# dvmrp show routes interface lab permission
DVMRP Routing Table (4232 routes,   5 hold-down-routes)
Net: 100.100.100/24         Gateway: 10.55.89.100      Met:  2   Age:  25
Parent: lab              Children: corp
                            mbone           leaf NoAdv

Net: 20.20.20/24            Gateway: 10.55.89.100      Met:  2   Age:  25
Parent: lab              Children: corp
                            mbone           leaf NoAdv

Net: 10.55.89/24            Gateway: ----              Met:  1   Age:  --
Parent: lab              Children: corp             leaf NoAdv
                            mbone           leaf NoAdv

Total Routes Printed: 3
```

dvmrp show rules

Purpose

Displays the rules in effect for filtering routes from DVMRP neighbor routers.

Format

dvmrp show rules

Mode

Enable

Description

The **dvmrp show rules** command displays the filtering rules in effect for DVMRP routes. Once you have set rules with the **dvmrp accept** and **dvmrp advertise** commands, you can display the active rules by entering the **dvmrp show rules** command.

Parameters

None.

Restrictions

None.

Example

In this example, the following rules are in effect:

```
dvmrp advertise route 207.135.89.0/24 interface mbone
dvmrp noadvertise route 0/0 interface mbone
dvmrp advertise route 207.135.88.0/24 interface mbone
dvmrp noadvertise route 10/8 interface corp
```

To display information about these rules:

```
# dvmrp show rules
NoAdvertise: 10.0.0.0/8              IF: corp
Advertise  : 207.135.89.0/24          IF: mbone
Advertise  : 207.135.88.0/24          IF: mbone
NoAdvertise: default                IF: mbone
```

These rules would affect the routing table as follows:

```
# dvmrp show route net 10/8 permissions

Net: 10.55.89/24          Gateway: ----          Met:  1   Age:  --
Parent: lab               Children: corp         leaf NoAdv
                                    mbone        leaf NoAdv
```

These rules prevent a directly connected route on this router from being visible to interface corp and mbone. The leaf flag indicates there is no downstream neighbor on the interface.

dvmrp start

Purpose

Starts DVMRP multicast routing.

Format

dvmrp start

Mode

Configure

Description

The **dvmrp start** command starts DVMRP multicast routing on the configured multicast-enabled interfaces and tunnels.

Note: Because DVMRP is a multicast protocol, IGMP will start and stop with DVMRP. However, using PIM as the multicast routing protocol requres that users run the appropriate **pim igmp** commands. Users who wish to run only IGMP on local interfaces must use the **dvmrp start** command.

By default, DVMRP is not enabled and does not interact with any unicast protocol. However if you need to run a tunnel, make sure that the tunnel is accessible through a unicast routing mechanism.

Parameters

None.

Restrictions

The X-Pedition does not allow users to enable DVMRP and PIM simultaneously. If a user attempts to enable DVMRP and PIM at the same time, one of the following messages will appear:

%CLI-E-NODVMRPFAC, This command cannot be used when PIM has been configured
%CLI-E-NOPIMFAC, This command cannot be used when IGMP or DVMRP has been configured.

To switch between PIM and DVMRP you must remove the protocol's start command from the startup configuration and restart the router.

Chapter 18

enable Command

The **enable** command switches the CLI session from User mode to Enable mode.

Format

enable

Mode

User

Description

Enable mode provides more facilities than User mode. You can display critical features within Enable mode including router configuration, access control lists, and SNMP statistics. To enter Enable mode from the User mode, enter the command **enable** (or **en**), then supply the password when prompted. If no password is configured, a warning message advising you to configure a password is displayed. If a password is configured and you do not know your password or pressing Return does not work, see the administrator for the X-Pedition.

The Enable mode command prompt consists of the X-Pedition name followed by the pound sign(#):

```
xp#
```

To exit Enable mode and return to User mode, type **exit** and press Return, or press Ctrl+Z. To proceed from the Enable mode into the Configure mode, use the **configure** command.

Parameters

None.

Restrictions

None.

Chapter 19

erase Command

The **erase** command erases the contents of the scratchpad or Startup configuration files.

Format

erase scratchpad|startup

Mode

Configure

Description

The **erase scratchpad** command erases the contents of the X-Pedition's command scratchpad. The **erase startup** command erases the Startup configuration from the Control Module's NVRAM.

Parameters

scratchpad	Erases the contents of the scratchpad. The scratchpad contains configuration commands that you have issued but have not yet activated.
startup	Erases the contents of the Startup configuration. The Startup configuration is the configuration the X-Pedition uses to configure itself when you reboot it. When you erase the Startup configuration, then reboot immediately, the X-Pedition restarts without any configuration information.

Restrictions

The erase commands do not delete other types of files. To delete a file, use the **file del** command.

Chapter 20

exit Command

The **exit** command exits the current CLI mode to the previous mode. For example, if you are in the Enable mode, **exit** returns you to the User mode. If you are in Configure mode, **exit** returns you to Enable mode. If you are in User mode, **exit** closes your CLI session and logs you off the X-Pedition.

Format

exit

Mode

All modes.

Parameters

None.

Restrictions

None.

Chapter 21

file Commands

The **file** commands enable you to display a directory of the files on a storage device, display the contents of a file on the console, and delete a file.

Command Summary

Table 16 lists the **file** commands. The sections following the table describe the command syntax.

Table 16. file commands

file copy backup-cm| primary-cm {[bootflash:| slot0:] <*src-file-name*>**}** **{[bootflash:| slot0:]** <*dest-file-name*>**}**
file delete backup-cm| primary-cm {[bootflash:| slot0:] <*file-name*>**}**
file dir backup-cm| primary-cm {bootflash:| slot0:} [directory-name]
file rename [device-name] <*original-file-name*> <*new-file-name*>
file type <*file-name*>

file copy

Purpose

Copy a file.

Format

file copy backup-cm| primary-cm {[bootflash:| slot0:] <*src-file-name*>**}**
{[bootflash:| slot0:] <*dest-file-name*>**}**

Mode

User

Description

Copies a file from a specified CM's device to a device on the Primary CM.

Parameters

backup-cm The source file to copy from is on the Backup CM.

primary-cm The source file to copy from is on the Primary CM.

bootflash: The Control Module's NVRAM—the default if device-name not specified.

slot0: The PCMCIA Flash Module in slot 0 (the upper slot).

Note: Device names end with a colon and are not followed by a space.

The X-Pedition supports PCMCIA Flash Modules obtained from Enterasys Networks only. For information regarding the PCMCIA Virtual File systems VFS1 and VFS2, see the *Enterasys X-Pedition User Reference Manual*.

<*src-file-name*>
Name of the file to copy from, located on either the Backup CM or the Primary CM.

<*dest-file-name*>
Name of the file to copy to the Primary CM only.

Restrictions

You can copy a file from either the Backup CM or Primary CM to the Primary CM only. Copying files to the Backup CM is not allowed.

Example

To copy a file named **core** from the slot0 device on the backup CM to the bootflash device on the primary CM and rename it to **core1**:

```
xp# file copy backup-cm slot0:core bootflash:core1
```

To copy a file named **core** from the slot0 device on the primary CM to the bootflash device on the primary CM and rename it to **core1**(if no device is specified, then the default device will be the bootflash):

```
xp# file copy primary-cm slot0:core core1
```

To copy a file named **tempfile** from the bootflash device on the backup CM to the bootflash device on the primary CM and rename it to **newfile** (if no device is specified, then the default device will be the bootflash):

```
xp# file copy backup-cm tempfile newfile
```

file delete

Purpose

Delete a file.

Format

file delete backup-cm| primary-cm {[bootflash:| slot0:] <*file-name*>**}**

Mode

Enable

Description

The **file delete** command deletes the specified file from either the Primary CM or the Backup CM. By default, if a device-name is not specified, it is assumed to be the **bootflash:** device.

Parameters

backup-cm The file to delete is on the Backup CM.

primary-cm The file to delete is on the Primary CM.

bootflash: The Control Module's NVRAM—the default if device-name not specified.

slot0: The PCMCIA Flash Module in slot 0 (the upper slot).

Note: Device names end with a colon and are NOT followed by a space. The X-Pedition supports PCMCIA Flash Modules obtained from Enterasys Networks only. For information regarding the PCMCIA Virtual File systems VFS1 and VFS2, see the *Enterasys X-Pedition User Reference Manual*.

<*file-name*> Name of the file to delete.

Restrictions

None.

Examples

To delete the file bootflash:config.old from the Primary CM:

```
xp# file delete primary-cm bootflash:config.old
```

To delete the file slot0:core.backup from the Backup CM:

```
xp# file delete backup-cm slot0:config.old
```

To delete the file config.save (default bootflash:) on the Backup CM:

```
xp# file delete backup-cm config.save
```

file dir

Purpose

Display contents of a file system.

Format

file dir backup-cm| primary-cm {bootflash:| slot0:} [directory-name]

Mode

User.

Description

Displays a directory of the files on the specified storage device.

Parameters

backup-cm Display the contents of a file system on the Backup control module.

primary-cm Display the contents of a file system on the Primary control module.

device-name Device name of file system to list. You can specify one of the following:

bootflash: The Control Module's NVRAM.
slot0: The PCMCIA Flash Module in slot 0 (the upper slot).

Note: Device names end with a colon and are not followed by a space.

The X-Pedition supports PCMCIA Flash Modules obtained from Enterasys Networks only. For information regarding the PCMCIA Virtual File systems VFS1 and VFS2, see the *Enterasys X-Pedition User Reference Manual*.

directory-name Optional directory to list.

Restrictions

None.

Examples

To display the contents of a file system:

```
xp# file dir backup-cm bootflash:
xp# file dir primary-cm slot0:tmp/
```

file rename

Purpose

Rename a file.

Format

file rename [device-name] <*original-file-name*> <*new-file-name*>

Mode

User

Description

Renames a file from the original file name to a new file name within the same directory.

Parameters

device-name Optional device name of file system where the file is located. You can specify one of the following options:

- **bootflash:** The Control Module's NVRAM (the default if device-name not specified).
- **slot0:** The PCMCIA Flash Module in slot 0 (the upper slot).

Note: Device names end with a colon and are not followed by a space.

The X-Pedition supports PCMCIA Flash Modules obtained from Enterasys Networks only. For information regarding the PCMCIA Virtual File systems VFS1 and VFS2, see the *Enterasys X-Pedition User Reference Manual*.

<*original-file-name*>
File you wish to rename.

<*new-file-name*>
The new name of the file, which will exist in the same directory specified by the original-file-name. Users are not allowed to use a device name or directory name as part of the *new-file-name*.

Restrictions

Not available for files on the Backup CM.

Examples

To rename a file:

```
xp# file rename slot0:core tempcore
```

```
xp# file rename file1 file2
```

file type

Purpose

Display contents of a file.

Format

file type <*file-name*>

Mode

Enable.

Description

Displays the contents of a file.

Parameters

<*file-name*> Name of the file to display. The filename can include a device-name using this format: <*device-name*>**:**<*file-name*>. By default, if a device-name is not specified, it is assumed to be the **bootflash:** device. The **bootflash:** device is the default device for storing configuration files.

<*device-name*> Device name. You can specify one of the following:

Note: Device names end with a colon.

bootflash: The Control Module's NVRAM.

slot0: The PCMCIA Flash Module in slot 0 (the upper slot).

slot1: The PCMCIA Flash Module in slot 1 (the lower slot).

Note: The X-Pedition supports PCMCIA Flash Modules obtained from Enterasys Networks only. For information regarding the PCMCIA Virtual File systems VFS1 and VFS2, see the *Enterasys X-Pedition User Reference Manual*.

Restrictions

None.

Examples

To display the contents of the file startup (the startup configuration file):

```
xp# file type startup
```

Chapter 22

filters Commands

The **filters** commands let you create and apply the following types of security filters:

- **Address filters.** Address filters block traffic based on a frame's source MAC address, destination MAC address, or both. Address filters are always configured and applied on the input port.
- **Static entry filters.** Static entry filters allow or force traffic to go to a set of destination ports based on a frame's source MAC address, destination MAC address, or both. Static entry filters are always configured and applied on the input port. You can configure source static entry filters, destination static entry filters, and flow static entry filters. Source static entry filters allow or disallow frames based on their source MAC address; destination static entry filters allow or disallow frames based on their destination MAC address. Flow static entries allow or disallow traffic based on their source *and* destination MAC addresses.
- **Port-to-address locks.** Port-to-address lock filters "lock" a user to a port or set of ports, disallowing them access to other ports.
- **Secure ports.** Secure port filters shut down Layer 2 access to the X-Pedition from a specific port or drop all Layer 2 packets received by a port. Used by themselves, secure ports secure unused X-Pedition ports. When used in conjunction with static entry filters, secure ports drop all received or sent traffic (depending on the static entry filter) except traffic forced to or from the port by the static entry filter.

Command Summary

Table 17 lists the **filters** commands. The sections following the table describe the command syntax.

Table 17. filters commands

filters add address-filter name <*name*> **source-mac** <*MACaddr*> **source-mac-mask** <*MACaddr*> **dest-mac** <*MACaddr*> **dest-mac-mask** <*MACaddr*> **vlan** <*VLAN-num*> **in-port-list** <*port-list*>
filters add port-address-lock name <*name*> **source-mac** <*MACaddr*> **vlan** <*VLAN-num*> **in-port-list** <*port-list*>
filters add secure-port name <*name*> **direction source|destination vlan** <*VLAN-num*> **in-port-list** <*port-list*>
filters add static-entry name <*name*> **restriction allow|disallow|force source-mac** <*MACaddr*> **source-mac-mask** <*MACaddr*> **dest-mac** <*MACaddr*> **dest-mac-mask** <*MACaddr*> **vlan** <*VLAN-num*> **in-port-list** <*port-list*> **out-port-list** <*port-list*>
filters show address-filter [all-source|all-destination|all-flow] [source-mac <*MACaddr*> **dest-mac** <*MACaddr*>**] [ports** <*port-list*>**] [vlan** <*VLAN-num*>**]**
filters show port-address-lock ports [ports <*port-list*>**] [vlan** <*VLAN-num*>**] [source-mac** <*MACaddr*>**]**
filters show secure-port
filters show static-entry [all-source|all-destination|all-flow] ports <*port-list*> **vlan** <*VLAN-num*> **[source-mac** <*MACaddr*> **dest-mac** <*MACaddr*>**]**

filters add address-filter

Purpose

Applies an address filter.

Format

filters add address-filter name <*name*> **source-mac** <*MACaddr*> **source-mac-mask** <*MACaddr*> **dest-mac** <*MACaddr*> **dest-mac-mask** <*MACaddr*> **vlan** <*VLAN-num*> **in-port-list** <*port-list*>

Mode

Configure

Description

The **filters add address-filter** command blocks traffic based on a frame's source MAC address (**source-mac**), destination MAC address (**dest-mac**), or a flow (specified using both a source MAC address and a destination MAC address).

Parameters

name <*name*>	Specifies the name of the filter. This parameter must be less than 25 characters.
source-mac <*MACaddr*>	Specifies the source MAC address. Use this option for source or flow address filters.
source-mac-mask <*MACaddr*>	Specifies the source MAC Mask address. Use this option for source or flow address filters.
dest-mac <*MACaddr*>	Specifies the destination MAC address. Use this option for destination or flow address filters.
dest-mac-mask <*MACaddr*>	Specifies the destination MAC Mask address. Use this option for destination or flow static entries.
vlan <*VLAN-num*>	Specifies the VLAN.
in-port-list <*port-list*>	Specifies the ports to which you want to apply the filter.

Restrictions

You should apply flow filters (specified using both a source MAC address and a destination MAC address) only to ports that are using flow-based bridging.

filters add port-address-lock

Purpose

Applies a port address lock.

Format

filters add port-address-lock name <*name*> **source-mac** <*MACaddr*> **vlan** <*VLAN-num*> **in-port-list** <*port-list*>

Mode

Configure

Description

The **filters add port-address-lock** command locks a user (identified by the user's MAC address) to a specific port or set of ports. The source MAC address will be allowed to reach only those stations and other ports that are connected to a port specified by **in-port-list**.

Parameters

name <*name*>	Specifies the name of the lock filter. This parameter must be less than 25 characters.
source-mac <*MACaddr*>	Specifies the source MAC address.
vlan <*VLAN-num*>	Specifies the VLAN.
in-port-list <*port-list*>	Specifies the ports to which you want to apply the lock.

Restrictions

None.

filters add secure-port

Purpose

Applies a port security filter.

Format

filters add secure-port name <*name*> **direction source|destination vlan** <*VLAN-num*> **in-port-list** <*port-list*>

Mode

Configure

Description

The **filters add secure-port** command shuts down Layer 2 access to the X-Pedition from the ports specified by **in-port-list**. The X-Pedition drops all traffic received from these ports.

Note: You can use port-to-address lock filters to force traffic to a port secured by the **filters add secure-port** command.

Parameters

name <*name*>
Specifies the name of the filter. This parameter must be less than 25 characters.

direction source|destination
Specifies whether the filter is to secure a source port or a destination port.

vlan <*VLAN-num*>
Specifies the VLAN.

in-port-list <*port-list*>
Specifies the ports to which you want to apply the filter.

Restrictions

None.

filters add static-entry

Purpose

Applies a static entry.

Format

filters add static-entry name <*name*> **restriction allow|disallow|force source-mac** <*MACaddr*> **source-mac-mask** <*MACaddr*> **dest-mac** <*MACaddr*> **dest-mac-mask** <*MACaddr*> **vlan** <*VLAN-num*> **in-port-list** <*port-list*> **out-port-list** <*port-list*>

Mode

Configure

Description

The **filters add static-entry** command allows, disallows, or forces traffic to go to a set of destination ports based on a frame's source MAC address (**source-mac**), destination MAC address (**dest-mac**), or a flow (specified using both a source MAC address and a destination MAC address).

Parameters

name <*name*>
Specifies the name of the static-entry filter. This parameter must be less than 25 characters.

restriction allow|disallow|force
Specifies the forwarding behavior of the static entry, which can be one of the following keywords:

- **allow** Allows packets to go to the set of ports specified by out-port-list.
- **disallow** Prohibits packets from going to the set of ports specified by out-port-list.
- **force** Forces packets to go to the set of ports specified by out-port-list, despite any port locks in effect on the ports.

source-mac <*MACaddr*>
Specifies the source MAC address. Use this option for source or flow static entries.

source-mac-mask <*MACaddr*>
Specifies the source MAC address. Use this option for source or flow static entries.

dest-mac <*MACaddr*>
Specifies the destination MAC address. Use this option for destination or flow static entries.

dest-mac-mask <*MACaddr*>
Specifies the destination MAC address. Use this option for destination or flow static entries.

vlan <*VLAN-num*>
Specifies the VLAN.

in-port-list <*port-list*>
Specifies the ports to which you want to apply the static entry.

out-port-list <*port-list*>
Specifies the ports to which you are allowing, disallowing, or forcing packets.

Restrictions

You should apply flow filters (specified using both a source MAC address and a destination MAC address) only to ports that are using flow-based bridging.

filters show address-filter

Purpose

Displays the address filters.

Format

filters show address-filter [all-source|all-destination|all-flow]
[source-mac <*MACaddr*> **dest-mac** <*MACaddr*>**] [ports** <*port-list*>**] [vlan** <*VLAN-num*>**]**

Mode

Enable

Description

The **filters show address-filter** command displays the address filters currently configured on the X-Pedition.

Parameters

all-source|all-destination|all-flow
Specifies the types of filters you want to display.

source-mac <*MACaddr*>
Restricts the display to only those address filters that have been applied to this source MAC address.

dest-mac <*MACaddr*>
Restricts the display to only those address filters that have been applied to this destination MAC address.

ports <*port-list*>
Restricts the display to only those address filters that have been applied to the specified ports.

vlan <*VLAN-num*>
Restricts the display to only those address filters that have been applied to the specified VLANs.

Restrictions

None.

filters show port-address-lock

Purpose

Display the port address locks.

Format

filters show port-address-lock [ports <*port-list*>**]**
[vlan <*VLAN-num*>**] [source-mac** <*MACaddr*>**]**

Mode

Enable

Description

The **filters show port-address-lock** command displays the port-address-lock filters currently configured on the X-Pedition.

Parameters

ports <*port-list*>
Restricts the display to only those port address locks that have been applied to the specified ports.

vlan <*VLAN-num*>
Restricts the display to only those port address locks that have been applied to the specified VLANs.

source-mac <*MACaddr*>
Restricts the display to only those port address locks that have been applied to this source MAC address.

Restrictions

None.

filters show secure-port

Purpose

Display the port security filters.

Format

filters show secure-port

Mode

Enable

Description

The **filters show secure-port** command displays the secure-port filters currently configured on the X-Pedition.

Parameters

None.

Restrictions

None.

filters show static-entry

Purpose

Displays the static entry filters.

Format

filters show static-entry [all-source|all-destination|all-flow] ports <*port-list*>
vlan <*VLAN-num*> **[source-mac** <*MACaddr*> **dest-mac** <*MACaddr*>**]**

Mode

Configure

Description

The **filters show static-entry** command displays the static-entry filters currently configured on the X-Pedition.

Parameters

all-source|all-destination|all-flow
Specifies the types of static entries you want to display.

ports <*port-list*>
Restricts the display to only those static entries that have been applied to the specified ports.

vlan <*VLAN-num*>
Restricts the display to only those static entries that have been applied to the specified VLANs.

source-mac <*MACaddr*>
Restricts the display to only those static entries that have been applied to this source MAC address.

dest-mac <*MACaddr*>
Restricts the display to only those static entries that have been applied to this destination MAC address.

Restrictions

None.

Chapter 23

fddi Commands

Command Summary

The **fddi** commands allow you to define and display parameters for the FDDI modules on the XP. Table 18 lists the fddi commands. The sections following the table describe the command syntax.

Note: A hardware limitation allows FDDI and SONET modules to increment only the *ifInUcastPkts* and *ifOutUcastPkts* ifMib counters. Non-unicast packet counters (i.e., *ifInNUcastPkts* and *ifOutNUcastPkts*) do not increment and will remain 0.

On gigabit and 10-Gigabit modules, all OCMAC counters increment correctly.

Table 18. fddi commands

fddi reset <*port*>
fddi set fddi-mode <*port*> **[sac-mode|das-mode]**
fddi set fddi-fdx-mode <*port*>
fddi set mac-group <*port*> **treq** <*number*> **ma-unit-data-enable [true|false]**
fddi set mac-restricted-token <*port*> <*value*>
fddi set path-group <*port*> **tvx-lower-bound** <*number*> **tmax-lower-bound** <*number*> **treq-max** <*number*>
fddi set port-group <*port*> **ler-cutoff** <*value*> **ler-alarm** <*value*> **connection** <**a**|**b**>
fddi set ring-purger <*port*>
fddi set smt-group <*port*> **userdata** <*string or number*> **connection-policy** <*number*> **tnotify** <*number*> **stat-rpt-policy [on|off] trace-max-expiration** <*value*>
fddi set translation [fddi_ipx_snap|fddi_appletalk_arp] to **[enet_II | 802.3_raw_ipx | 802.3_snap] port** <*port*>
fddi show fddi-mode <*port-list*>**|all ports**

Table 18. fddi commands (Continued)

fddi show fddi-status <*port-list*>**|all ports**
fddi show fddi-fdx-mode <*port-list*>**|all ports**
fddi show mac-group <*port-list*>**|all ports**
fddi show mac-restricted-token <*port-list*>**|all ports**
fddi show media-type <*port-list*>**|all-ports**
fddi show path-group <*port-list*>**|all ports**
fddi show port-group <*port-list*>**|all ports**
fddi show ring-purger <*port-list*>**|all ports**
fddi show smt-config <*port-list*>**|all ports**
fddi show smt-group <*port-list*>**|all ports**
fddi show translation <*port-list*>**|all ports**
fddi show version <*port-list*>**|all ports**

fddi reset

Purpose

Resets a specific FDDI port

Format

fddi reset <*port*>

Mode

Enable

Description

The **fddi reset** command allows you to restore a port without disrupting operations on the other port.

Parameters

<*port*>	Specifies the FDDI port. To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.

Restrictions

None

Example

To reset port fi.5.1:

```
xp# fddi reset fi.5.1
```

fddi set fddi-mode

Purpose

Sets the FDDI operating mode for the specified FDDI port(s).

Format

fddi set fddi-mode <*port*> **[sac-mode|das-mode]**

Mode

Configure

Description

The **fddi set fddi-mode** allows you to configure the ports as single attachment concentrators (SAC) or dual attachment stations (DAS). SACs attach to the primary ring only. They are used primarily to attach routers to a ring through concentrators. In SAC mode, port A becomes an M port, and port B becomes an S port. The M port in a SAC is used to extend the primary ring, and connects to an A, B, or S (slave) port. The S port in a SAC connects to a single ring only and typically connects to an M port.

A DAS connects to both the primary and secondary FDDI rings. Thus, in case of a device failure, it is capable of wrapping the ring.

Parameters

<*port*>	Specifies the FDDI port. To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.
sac-mode	Sets the operating mode to single attachment concentrator (SAC).
das-mode	Sets the operating mode to dual attachment station (DAS).

Restrictions

DAS mode is not available for UTP.

Example

To set the operating mode of port fi.5.1 to DAS:

```
xp# fddi set fddi-mode fi.5.1 das-mode
```

fddi set fddi-fdx-mode

Purpose

Sets the specified FDDI port to full-duplex mode.

Format

fddi set fddi-fdx-mode <*port*>

Mode

Configure

Description

The **fddi set fddi-fdx-mode** command sets a specific FDDI port to full-duplex mode. When you set the FDDI port to full-duplex mode, it executes a protocol that detects if there is another device on the FDDI ring that is also attempting to run full duplex. If it does detect another device in full-duplex mode and it is the only other device on the ring, the ports will operate in full-duplex mode. But if the port detects that there is more than one other device on the ring, it will cease to operate in full-duplex mode.

Note: Changing the station mode on a FDDI port will negate all previously executed FDDI commands.

Parameters

<*port*> Specifies the FDDI port. To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.

Restrictions

The SmartSwitch 6000 does not support FDDI full duplex operation.

Example

To set port fi.5.2 to full-duplex mode:

```
xp# fddi set fddi-fdx-mode fi.5.2
```

fddi set mac-group

Purpose

Sets the MAC configuration parameters for the specified FDDI port

Format

fddi set mac-group <*port*> **treq** <*number*> **ma-unit-data-enable [true|false]**

Mode

Configure

Description

The **fddi set mac-group** command sets a station's MAC parameters. During the claim process, each station "bids" on the right to initialize the ring. The station's bid is its token rotation time (**treq** value). The station with the fastest rotation time wins the claim as it can support the rotation time of all the other stations.

Parameters

<*port*> Specifies the FDDI port. To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.

treq <*number*>
Specifies the station's desired token rotation time in milliseconds. This value should be less than or equal to the **treq-max** value specified in the **fddi set path-group** command. Enter a value between 4 and 173.01504. Note that because of unit conversions, treq may be rounded slightly down from the value specified.

ma-unit-data-enable [true|false]
Sets the MA_UNITDATA_ENABLE flag to true or false.

Note: This parameter is included to be consistent with the MIB only. Setting this parameter on the X-Pedition has no affect on the operation of the network.

Restrictions

None.

Example

To set the MAC configuration parameters for port fi.4.2:

```
xp# fddi set mac group fi.4.2 treq 15 ma-unit-data-enable true
```

fddi set mac-restricted-token

Purpose

Sets the MAC restricted token time out for the specified FDDI port.

Format

fddi set mac-restricted-token <*port*> <*value*>

Mode

Configure

Description

The **fddi set mac-restricted-token** command specifies how long a station can hold a restricted token. A station that holds the restricted token can use the entire network bandwidth for an extended period. Upon completion of its transmission, the station with the restricted token converts the token to non-restricted, re-issues it to the ring and normal operations continue.

Parameters

<*port*>	Specifies an FDDI port. To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.
<*value*>	Specifies the MAC restricted token time out in milliseconds. Specify a value between 0 and 10000.

Restrictions

None.

Example

To set the restricted token time out for port fi.4.2:

```
xp# fddi set mac-restricted-token fi.4.2 25
```

fddi set path-group

Purpose

Sets the PATH configuration parameters for the specified FDDI port.

Format

fddi set path-group <*port*> **tvx-lower-bound** <*number*> **tmax-lower-bound** <*number*> **treq-max** <*number*>

Mode

Configure

Description

The **fddi set path-group** command sets thresholds for the timers used by any MAC configured in the primary path. The valid transmission timer (tvx) clocks the period between valid transmissions. When the station receives a valid transmission, the tvx resets. If no valid frame, including a token, is received and the tvx expires, the station begins ring initialization.

The **tmax** value is the minimum target rotation time (TTRT) supported by a MAC. The **treq-max** value is the maximum rotation time used by a MAC.

Parameters

<*port*> Specifies an FDDI port. To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.

tvx-lower-bound <*number*>
Specifies the minimum TVX value (in milliseconds) that shall be used by any MAC configured in this path. This value should be lower than the **treq-max**. Enter a value between 0 and 5.20192.

tmax-lower-bound <*number*>

Specifies the minimum TMAX value (in milliseconds) supported by any MAC configured in this path. This value should be greater than or equal to the **treq-max**. Enter a value between 10 and 1331.69152.

treq-max <*number*> Specifies the maximum TREQ value (in milliseconds) that shall be used by any MAC configured in this path. This value must be greater than the **tvx-lower-bound** and equal to or less than the **tmax-lower-bound**.

Note: Due to unit conversions, the **tvx-lower-bound, tmax-lower-bound, and treq-max** values may be rounded down slightly from the values specified.

Restrictions

None.

Example

To set the path configuration parameters for port fi.4.2:

```
xp# fddi set path-group fi.4.2 tvx-lower-bound 5 tmax-lower-bound 18 treq-max 18
```

fddi set port-group

Purpose

Sets parameters for the specified FDDI port.

Format

fddi set port-group <*port*> **ler-cutoff** <*value*> **ler-alarm** <*value*> **connection** <**a**|**b**>

Mode

Configure

Description

The **fddi set port-group** command allows you to specify link error monitoring (LEM) thresholds for an FDDI port. If the link error rate exceeds the **ler-alarm** value, an alarm is generated. If more errors are detected after the alarm, and the **ler-cutoff** is exceeded, the link is declared faulty and the connection is broken.

Parameters

<*port*>	Specifies an FDDI port. To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.
ler-cutoff <*value*>	Specifies the desired link error rate cutoff. If exceeded, the link connection will be broken.
ler-alarm <*value*>	Specifies the desired link error rate alarm limit. If exceeded, the link connection will generate an alarm.
connection <**a**|**b**>	Specifies the actual or physical port being configured. Specify **a** to set parameters for the port on the left side of the PHY. Specify **b** to set parameters for the port on the right side of the PHY.

Restrictions

None.

Example

To set the PORT configuration parameters for port fi.4.2:

```
xp# fddi set port-group fi.4.2 ler-cutoff 17 ler-alarm 20 connection b
```

fddi set ring-purger

Purpose

Turns on the ring purger mode for the specified FDDI port.

Format

fddi set ring-purger <*port*>

Mode

Configure

Description

The **fddi set ring-purger** command allows the FDDI port to participate in ring purger election. The station “elected” as the ring purger strips the ring of obsolete or stray frames and packets. This prevents old packets from continually circling the ring.

Parameters

<*port*>	Specifies an FDDI port. To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.

Restrictions

None.

Example

To enable ring purger mode for port fi.4.2:

```
xp# fddi set ring-purger fi.4.2
```

fddi set smt-group

Purpose

Sets the station management (SMT) parameters for the specified FDDI port.

Format

fddi set smt-group <*port*> **userdata** <*string or number*>|**connection-policy** <*number*>|**tnotify** <*number*>|**stat-rpt-policy [on|off]|trace-max-expiration** <*value*>

Mode

Configure

Description

The **fddi set smt-group** command allows you to set various station management parameters. There parameters include the following:

- The connection policy specifies which type of connection the station will reject. In the policy statement, the first value represents the local port and the second value represents the port of the adjacent station. The Bit # specifies the binary bit position. To specify a connection policy, determine which connections will be rejected, calculate the decimal value of the bit (2^ Bit #) for each connection rejected, and add these values together.

Policy	**Bit #**
Reject A-A	0
Reject A-B	1
Reject A-S	2
Reject A-M	3
Reject B-A	4
Reject B-B	5
Reject B-S	6
Reject B-M	7
Reject S-A	8

Reject S-B 9

Reject S-S 10

Reject S-M 11

Reject M-A 12

Reject M-B 13

Reject M-S 14

Reject M-M 15

- Each station on an FDDI ring announces its address to its downstream neighbor by transmitting neighborhood information frames (NIF). Specify the time period between the transmission of NIFs.
- When there is an unexpected network or node change, a station can generate status reporting frames (SRF) which notify the network of the unexpected event or condition (such as a ring wrap). You can turn this feature on or off for a particular port.

Parameters

<*port*> Specifies an FDDI port. To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.

userdata <*string or number*>
Specifies a text string or numbers. May be up to 7 characters.

connection-policy <*value*>
Specifies a value representing the connection policies in effect in a node. This value is a sum of the decimal values calculated for each connection policy. Note that Bit 15, (rejectM-M), is always set and cannot be cleared.
Example: Reject A-A, S-M and M-M = (2^0 + 2^11 + 2^15)

t**notify** <*value*> Specifies the amount of time between the transmission of neighborhood information frames (nifs).

stat-rpt-policy [on|**off]**
Specifies whether the station will generate status reporting frames for unexpected events and conditions.

trace-max-expiration <*value*>
Specifies the amount of time before a trace expires. Enter a value between 6002 and 100000.

Restrictions

None.

fddi set translation

Purpose

Sets the IPX/Appletalk ARP frame translation settings for the specified FDDI port.

Format

fddi set translation [fddi_ipx_snap| fddi_appletalk_arp] to **[enet_II|802.3_raw_ipx| 802.3_snap] port** <*port*>

Mode

Configure

Description

The **fddi set translation** command allows you to specify how FDDI IPX Snap frames or FDDI Appletalk ARP frames are translated to Ethernet. FDDI IPX Snap frames are translated to Ethernet II frames, by default. Alternatively, you can specify that they be translated to 802.3 Raw IPX frames. FDDI Appletalk ARP frames are translated to 802.3 Snap frames by default. Alternatively, you can specify that they be translated to Ethernet II frames.

Parameters

fddi_ipx_snap	Specifies that the frames to be translated are IPX Snap frames.
fddi_appletalk_arp	Specifies that the frames to be translated are FDDI Appletalk ARP frames.
enet_II	Specifies that the frames will be translated as Ethernet II frames.
802.3_raw_ipx	Specifies that the frames will be translated as 802.3 Raw IPX frames.
802.3_snap	Specifies that the frames will be translated as 802.3 Snap frames.

<port>	Specifies an FDDI port. To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.

Restrictions

- IPX Snap frames cannot be translated to 802.3 Snap frames.
- Appletalk ARP frames cannot be translated to 802.3 Raw IPX frames.

Example

To specify the translation settings for port fi.4.2:

```
xp# fddi set translation fddi_ipx_snap to enet_II fi.4.2
```

fddi show fddi-mode

Purpose

Displays the operating FDDI mode for the specified port(s).

Format

fddi show fddi-mode <*port-list*>|**all ports**

Mode

Enable

Description

The **fddi show fddi-mode** command displays the operating mode for a specified port. This allows you to determine whether the specified port is operating as a Single Attachment Concentrator (SAC) or Dual Attachment Station (DAS).

Parameters

<*port-list*>	Specifies the FDDI port(s). To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.
all-ports	Specifies all FDDI ports.

Restrictions

None.

Example

To display the operating mode of port fi.5.1:

```
xp# fddi show fddi-mode fi.5.1
```

fddi show fddi-status

Purpose

Displays the FDDI status for the specified FDDI port(s).

Format

fddi show fddi-status <*port-list*>|**all ports**

Mode

Enable

Description

The **fddi show fddi-status** command displays FDDI status information for the specified ports. This includes station ID, upstream and downstream neighbors, and station state.

Parameters

<*port-list*>	Specifies the FDDI port(s). To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.
all-ports	Specifies all FDDI ports.

Restrictions

None.

Example

To display the fddi status of ports fi.5.1 and fi.5.2:

```
xp# fddi show fddi-status fi.5.(1-2)
```

fddi show fddi-fdx-mode

Purpose

Displays the FDDI full duplex value for the specified FDDI port(s).

Format

fddi show fddi-fdx-mode <*port-list*>|**all ports**

Mode

Enable

Description

The **fddi show fddi-fdx-mode** command displays the full duplex value for a specific FDDI port or ports. This allows you to determine whether or not the specified port is active and running in full-duplex mode. Note that the FDDI port will operate in full-duplex mode only if there is one other station on the ring that is running full-duplex. It will cease running in full-duplex if there are multiple stations on the ring.

Note: Because FDDI full duplex is not an industry standard, its implementation in the SSR-FDDI-02 is based on the Digital Equipment Corporation (DEC) standard and will interoperate with all DEC products and most Enterasys FDDI products.

Parameters

<port-list>	Specifies the FDDI port(s). To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.
all-ports	Specifies all FDDI ports.

Restrictions

None.

Example

To display the full-duplex status of port fi.5.2:

```
xp# fddi show fddi-fdx-mode fi.5.2
```

fddi show mac-group

Purpose

Displays the MAC configuration parameters for the specified FDDI port(s).

Format

fddi show mac-group <*port-list*>|**all ports**

Mode

Enable

Description

The **fddi show mac-group** command allows you to display the requested rotation time (treq value) and MAC Unit Data Enable flag of the specified port(s).

Parameters

<*port-list*>	Specifies the FDDI port(s). To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.
all-ports	Specifies all FDDI ports.

Restrictions

None.

Example

To display the MAC parameters of port fi.3.2:

```
xp# fddi show mac-group fi.3.2
```

fddi show mac-restricted token

Purpose

Displays the MAC restricted token time for the specified FDDI port(s).

Format

fddi show mac-restricted-token <*port-list*>|**all ports**

Mode

Enable

Description

The **fddi show mac-restricted-token** command allows you to display the restricted token time for the specified port(s). A station that holds the restricted token can use the entire network bandwidth until the restricted token time expires.

Parameters

<*port-list*>	Specifies the FDDI port(s). To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.
all-ports	Specifies all FDDI ports.

Restrictions

None.

Example

To display the restricted token time for port fi.4.1:

```
xp# fddi show mac-restricted-token fi.4.1
```

fddi show media-type

Purpose

Displays the media type for the specified FDDI port(s).

Format

fddi show media-type <*port-list*>**|all ports**

Mode

Enable

Description

The **fddi show media-type** command allows you to display the media type and PHY states of the specified FDDI port(s). The media types are Single-Mode Fiber, Multi-Mode Fiber (MMF), and Unshielded Twisted Pair (UTP).

Parameters

<*port-list*>	Specifies the FDDI port(s). To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.
all-ports	Specifies all FDDI ports.

Restrictions

None.

Example

To display the media type of ports fi.4.1 and fi.4.2:

```
xp# fddi show media-type fi.4.(1-2)
```

fddi show path-group

Purpose

Displays the PATH configuration parameters for the specified port(s).

Format

fddi show path-group <*port-list*>|**all ports**

Mode

Enable

Description

The **fddi show path-group** command allows you to display the various thresholds for the timers used by the MACs in the primary path. These include the minimum valid transmission time (tvx), which is the period between valid transmissions; the minimum tmax value, which is the lowest target token rotation time supported by a MAC; and the maximum **treq** value, which is the maximum target rotation time that may be requested by a station in the primary path.

Parameters

<*port-list*>	Specifies the FDDI port(s). To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.
all-ports	Specifies all FDDI ports.

Restrictions

None.

Example

To display the PATH parameters for port fi.3.2:

```
xp# fddi show media-type fi.3.2
```

fddi show port-group

Purpose

Displays the PORT configuration parameters for the specified FDDI port(s).

Format

fddi show port-group <*port-list*>**|all ports**

Mode

Enable

Description

The **fddi show port-group** command allows you to display PORT configuration parameters for the specified FDDI port(s). These include the port connections; the link error rate alarm (ler-alarm**)** value, which is the number of link errors detected before an alarm is generated; and the link error rate cutoff (ler-cutoff) value, which is the number of link errors detected before the link is declared faulty and the connection is broken.

Parameters

<*port-list*>	Specifies the FDDI port(s). To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.
all-ports	Specifies all FDDI ports.

Restrictions

None.

Example

To display the PORT parameters for port fi.4.2:

```
xp# fddi show port-group fi.4.2
```

fddi show ring-purger

Purpose

Displays the ring purger value for the specified port(s).

Format

fddi show ring-purger <*port-list*>|**all ports**

Mode

Enable

Description

The **fddi show ring-purger** command allows you to display the ring purger status of the specified port(s). When this feature is turned on, the FDDI port participates in ring purger election.

Parameters

<*port-list*>	Specifies the FDDI port(s). To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.
all-ports	Specifies all FDDI ports.

Restrictions

None.

Example

To display the ring purger status of port fi.4.1:

```
xp# fddi show ring-purger fi.4.1
```

fddi show smt-config

Purpose

Displays the current SMT configurations for the specified FDDI port(s).

Format

fddi show smt-config <*port-list*>**|all ports**

Mode

Enable

Description

The **fddi show smt-config** command allows you to display station management (SMT) information for the specified FDDI port(s). These include the SMT version; the treq value, which is the station's requested token rotation time; the tneg value, which is the token rotation time negotiated by the station during the claim process; and the tnotify value, which is the period between the generation of neighborhood information frames.

Parameters

<*port-list*>	Specifies the FDDI port(s). To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.
all-ports	Specifies all FDDI ports.

Restrictions

None.

Example

To display SMT information for port fi.4.2:

```
xp# fddi show smt-config fi.4.2
```

fddi show smt-group

Purpose

Displays the SMT configuration parameters for the specified FDDI port(s).

Format

fddi show smt-group <*port-list*>|**all ports**

Mode

Enable

Description

The **fddi show smt-group** command allows you to display Station Management (SMT) parameters that were set for the specified port(s). These include the user data; connection policy, which specifies the type of connections rejected by the port(s); the status report policy, which specifies whether a station generates Status Reporting Frames (SRF) when unexpected events or conditions occur; the time period between the generation of Neighborhood Information Frames (NIF); and the trace max expiration time.

Parameters

<*port-list*>	Specifies the FDDI port(s). To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.
all-ports	Specifies all FDDI ports.

Restrictions

None.

Example

To display SMT parameters configured for port fi.4.1:

```
xp# fddi show smt-group fi.4.1
```

fddi show translation

Purpose

Displays the frame translation settings for the specified port(s).

Format

fddi show translation <*port-list*> **| all-ports**

Mode

Enable

Description

The **fddi show translation** command allows you to display the IPX/Appletalk ARP frame translation settings.

Parameters

<*port-list*>	Specifies the FDDI port(s). To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.
all-ports	Specifies all FDDI ports.

Restrictions

None.

Example

To display the translation settings configured for port fi.4.1:

```
xp# fddi show translation fi.4.1
```

fddi show version

Purpose

Displays the firmware version of the specified FDDI port(s).

Format

fddi show version <*port-list*>|**all ports**

Mode

Enable

Description

The **fddi show version** command allows you to display the firmware version of the specified port(s).

Parameters

<*port-list*>	Specifies the FDDI port(s). To specify an FDDI port, use the prefix: fi. For example, to specify an FDDI port in slot 5, use fi.5.1.
all-ports	Specifies all FDDI ports.

Restrictions

None.

Example

To display the firmware version of port fi.4.1:

```
xp# fddi show version fi.4.1
```

Chapter 24

frame-relay Commands

The **frame-relay** commands allow you to define frame relay service profiles, and specify and monitor frame relay High-Speed Serial Interface (HSSI) and standard serial ports.

Command Summary

Table 19 lists the **frame-relay** commands. The sections following the table describe the command syntax.

Table 19. frame relay commands

frame-relay apply service <*service name*> **ports** <*port list*>
frame-relay clear stats-counter [frame-drop-qdepth-counter] [max-frame-enqued-counter] [frame-drop-red-counter] [rmon] ports <*port list*>
frame-relay create vc <*port*>
frame-relay define service <*service name*> **[Bc** <*number*>**] [Be** <*number*>**]** **[becn-adaptive-shaping** <*number*>**] [cir** <*number*>**]** **[high-priority-queue-depth** <*number*>**] [low-priority-queue-depth** <*number*>**]** **[med-priority-queue-depth** <*number*>**] [red on|off]** **[red-maxTh-high-prio-traffic** <*number*>**] [red-maxTh-low-prio-traffic** <*number*>**]** **[red-maxTh-med-prio-traffic** <*number*>**] [red-minTh-high-prio-traffic** <*number*>**]** **[red-minTh-low-prio-traffic** <*number*>**] [red-minTh-med-prio-traffic** <*number*>**]** **[de-mark on|off]**
frame-relay set fr-encaps-bgd ports <*port list*>
frame-relay set lmi [error-threshold <*number*>**] [full-enquiry-interval** <*number*>**]** **[monitored-events** <*number*>**] [polling-interval** <*number*>**] [state enable|disable]** **[type ansi617d-1994|q933a|rev1] ports** <*port list*>
frame-relay set payload-compress [type frf9_mode1_stac] ports <*port list*>

Table 19. frame relay commands (Continued)

frame-relay set peer-addr [ip-address <*IP address*>**] [ipx-address** <*ipx address*>**] [ports** <*port list*>**]**
frame-relay show service <*service name*>**|all**
frame-relay show stats ports <*port name*> **[last-error] [lmi] [mibII]**
frame-relay show stats ports <*port name*> **summary**

frame-relay apply service ports

Purpose

Apply a pre-defined service profile to a frame relay virtual circuit (VC).

Format

frame-relay apply service <*service name*> **ports** <*port list*>

Mode

Configure

Description

Issuing the **frame-relay apply service** command allows you to apply a previously defined service profile to a given frame relay VC.

Parameters

<*service name*>	The name of the previously defined service profile you wish to apply to the given port(s) or interfaces.
<*port list*>	The port(s) to which you wish to apply the pre-defined service profile. You can specify a single VC or a comma-separated list of VCs.

Restrictions

Usage is restricted to frame relay VCs only.

Example

To apply the service “s1” to slot 2, VC 100 on serial ports 1 and 2:

```
xp(config)# frame-relay apply service s1 ports se.2.1.100,se.2.2.100
```

frame-relay clear stats-counter

Purpose

Clears the specified statistics counter.

Format

frame-relay clear stats-counter [frame-drop-qdepth-counter] [max-frame-enqued-counter] [frame-drop-red-counter] [rmon] ports <*port list*>

Mode

Enable

Description

The **frame-relay clear stats-counter** command allows you to specify a particular statistic counter and have those statistics reset to zero. There are statistic counters on each WAN port, and you can use the **frame-relay clear stats-counter** to clear the counter for an individual WAN port or for a group of ports.

Parameters

frame-drop-qdepth-counter	Specify this optional parameter to reset the frame drop counter to zero.
max-frame-enqued-counter	Specify this optional parameter to reset the max enquedframes counter to zero.
frame-drop-red-counter	Specify this optional parameter to reset the packet drop counter to zero.
rmon	Specify this optional parameter to reset the rmon counter to zero.
<*port list*>	The WAN port(s) that you wish to clear the counter.

Restrictions

Usage is restricted to WAN ports only.

Example

To clear the frame drop counter to zero on WAN port hs.3.1:

```
xp# frame-relay clear stats-counter frame-drop-qdepth-counter port hs.3.1
```

frame-relay create vc

Purpose

Create frame relay virtual circuits (VCs).

Format

frame-relay create vc <*port*>

Mode

Configure

Description

The **frame-relay create vc** command allows you to create a frame-relay virtual circuit on a slot and port location specified in the command line.

Parameters

<*port*> The port on which you wish to create a frame relay virtual circuit. Specify the port in the following format: **media.slot.port.dlci**.

media Is the media type.

slot Is the slot number where the module is installed.

port Is the port number.

dlci Is the Data Link Connection Identifier. Specify any number between 16-1007.

Restrictions

Usage is restricted to frame relay ports only.

Example

To create a frame relay virtual circuit with a DLCI of 100 on serial port 1 of slot 3:

```
xp(config)# frame-relay create vc port se.3.1.100
```

frame-relay define service

Purpose

Configure service profiles for frame relay ports.

Format

frame-relay define service <*service name*> **[Bc** <*number*>**] [Be** <*number*>**]**
[becn-adaptive-shaping <*number*>**] [cir** <*number*>**] [high-priority-queue-depth** <*number*>**]**
[low-priority-queue-depth <*number*>**] [med-priority-queue-depth** <*number*>**] [red on|off]**
[red-maxTh-high-prio-traffic <*number*>**] [red-maxTh-low-prio-traffic** <*number*>**]**
[red-maxTh-med-prio-traffic <*number*>**] [red-minTh-high-prio-traffic** <*number*>**]**
[red-minTh-low-prio-traffic <*number*>**] [red-minTh-med-prio-traffic** <*number*>**]**
[de-mark on|off]

Mode

Configure

Description

The **frame-relay define service** command allows you to specify the following attributes for a newly created service profile:

- Number of bits per second contained in a committed burst for frame relay virtual circuits.
- Number of bits per second contained in an excessive burst for frame relay virtual circuits.
- Whether or not to simultaneously enable and specify the threshold at which adaptive shaping will activate when receiving BECN frames
- The committed information rate (in bits per second) for frame relay virtual circuits.
- The allowable queue depth for high-, low-, and medium-priority frames on frame relay VCs.
- Activation or deactivation of Random Early Discard (RED) for frame relay circuits.
- The maximum and minimum threshold values for RED high-, low-, and medium-priority traffic.

  In general, Enterasys recommends that the maximum threshold values be less than or equal to the respective high-, low-, or medium-priority queue depth. The minimum threshold values should be one-third of the respective maximum threshold.

Parameters

<service name>
The name you wish to assign to the newly created service profile.

Bc <*number*>
The number of bits per second contained in a committed burst for a frame relay virtual circuit. You can specify a number between 1 and 2,147,483,646 bits per second.

Be <*number*>
The number of bits per second contained in an excessive burst for a frame relay virtual circuit. You can specify a number between 1 and 2,147,483,646 bits per second.

becn-adaptive-shaping <*number*>
The threshold (number of frames) at which adaptive shaping will activate when receiving BECN frames. You can specify a number between 1 and 100,000 frames.

cir <*number*>
The committed information rate (in bits per second) for frame relay virtual circuits. You can specify a number between 1 and 2,147,483,646 bits.

high-priority-queue-depth <*number*>
The number of high-priority frames allowed in the frame relay queue. You can specify a number between 1 and 65,535. Enterasys recommends a value within the 5 - 100 item range. The default value is 20.

low-priority-queue-depth <*number*>
The number of low-priority frames allowed in the frame relay queue. You can specify a number between 1 and 65,535. Enterasys recommends a value within the 5 - 100 item range. The default value is 20.

med-priority-queue-depth <*number*>
The number of medium-priority frames allowed in the frame relay queue. You can specify a number between 1 and 65,535. Enterasys recommends a value within the 5 - 100 item range. The default value is 20.

red on|off
Specifying the **on** keyword enables RED for frame relay ports. Specifying the **off** keyword disables RED for frame relay ports.

red-maxTh-high-prio-traffic <*number*>
The maximum allowable number of frames for high-priority RED traffic. You can specify a number between 1 and 65,535. The default value is 12.

red-maxTh-low-prio-traffic <*number*>
The maximum allowable number of frames for low-priority RED traffic. You can specify a number between 1 and 65,535. The default value is 12.

red-maxTh-med-prio-traffic <*number*>
The maximum allowable number of frames for medium-priority RED traffic. You can specify a number between 1 and 65,535. The default value is 12.

red-minTh-high-prio-traffic <*number*>
The minimum allowable number of frames for high-priority RED traffic. You can specify a number between 1 and 65,535. The default value is 4.

red-minTh-low-prio-traffic <*number*>
The minimum allowable number of frames for low-priority RED traffic. You can specify a number between 1 and 65,535. The default value is 4.

red-minTh-med-prio-traffic <*number*>
The minimum allowable number of frames for medium-priority RED traffic. You can specify a number between 1 and 65,535. The default value is 4.

de-mark on|off
Specifying the **on** keyword enables DE marking for best traffic. Specifying the **off** keyword disables DE marking for best traffic. Default is **off**.

Restrictions

When defining a value for **Bc**, you *must* also be sure to define an appropriate value for **cir**, and vice-versa.

Examples

Suppose you wish to specify a frame relay virtual circuit with the following attributes:

- Committed burst value of 35 million and excessive burst value of 30 million
- BECN active shaping at 65 thousand frames
- Committed information rate (CIR) of 120 million bits per second
- Leave high-, low-, and medium-priority queue depths set to factory defaults
- Random Early Discard (RED) disabled

The command line necessary to set up a service profile with the above attributes would be as follows:

```
xp(config)# frame-relay define service profile1 Bc 35000000 Be 30000000 becn-adaptive-shaping
    65000 cir 120000000 red off
```

frame-relay set fr-encaps-bgd

Purpose

Force the ingress packets to be encapsulated in bridged format.

Format

frame-relay set fr-encaps-bgd ports <*port list*>

Mode

Configure

Description

Issuing the **frame-relay set fr-encaps-bgd** command allows you to use bridged format encapsulation on a given frame relay VC.

Parameters

<*port list*> The port(s) to which you wish to use bridged encapsulation. You can specify a single VC or a comma-separated list of VCs.

Restrictions

Usage is restricted to frame relay VCs only.

Example

To force the bridged encapsulation to slot 2, VC 100 on serial ports 1 and 2:

```
xp(config)# frame-relay set fr-encaps-bgd ports se.2.1.100,se.2.2.100
```

frame-relay set lmi

Purpose

Set frame relay Local Management Interface (LMI) parameters.

Format

frame-relay set lmi [error-threshold <*number*>**] [full-enquiry-interval** <*number*>**] [monitored-events** <*number*>**] [polling-interval** <*number*>**] [state enabled|disabled] [type ansi617d-1994|q933a|rev1] ports** <*port list*>

Mode

Configure

Description

The **frame-relay set lmi** command allows you to specify the following attributes:

- The number of times the router will attempt to poll an LMI interface before declaring it down. You can define a value between 1 and 10, inclusive.
- The number of status enquiries that will be sent before a full status enquiry is requested. You can define a value between 1 and 255, inclusive.
- The number of status enquiries over which various pieces of LMI information can be collected and tabulated. For example, you can tabulate the number of times an interface was declared down/lost due to a lack of proper responses to status enquiries. You can define a value between 1 and 10, inclusive.
- The number of seconds that pass between successive status enquiry messages. You can define a value between 5 and 30, inclusive.
- Whether or not LMI messages are sent. LMI messages are not sent by default.
- The LMI type for frame relay WAN ports.

Parameters

error-threshold <*number*>
The number of unanswered status enquiries that the router will make before declaring an interface to be down.

full-enquiry-interval <*number*>
The number of status enquiries that will be sent before a full report on status is compiled and transmitted.

monitored-events <*number*>
The number of status enquiries over which collection and tabulation of various pieces of LMI information will take place.

polling-interval <*number*>
The amount of time (in seconds) that will pass before a subsequent status enquiry takes place.

state enabled|disabled
Enables the sending and receiving of LMI messages. If LMI messages are enabled, the operational status of each VC is determined by the LMI messages. If LMI messages are disabled, each VC is assumed to be operationally "up". LMI messages are disabled by default.

type ansi617d-1994|q933a|rev1
The LMI type for frame relay WAN ports. You can only specify the **ansi617d-1994**, **q933a**, or **rev1** keywords to define as the LMI type for WAN ports.

ports <*port list*>
The port or ports that will assume the LMI service profile behavior.

Restrictions

None.

Examples

To set the number of status enquiries that will be sent before compilation and transmission of a full status report for serial port 2 of slot 2 to 75 enquiries:

```
xp(config)# frame-relay set lmi full-enquiry-interval 75 ports se.2.2
```

frame-relay set payload-compress

Purpose

Enable packet compression for frame-relay ports.

Format

frame-relay set payload-compress [type frf9_mode1_stac] ports<*port list*>

Mode

Configure

Description

The **frame-relay set payload-compress** command allows you to enable packet compression according to Mode 1 of FRF 9. If this command is not configured, packet compression is not enabled.

Parameters

type frf9_mode1_stac
Specifies the Stacker FRF 9, Mode 1 compression algorithm. This is the default value.

<*port list*>
The port(s) on which you wish to enable the packet compression. You can specify a single VC or a comma-separated list of VCs.

Restrictions

Usage is restricted to frame relay VCs only.

Example

To enable Stacker FRF 9, Mode 1 packet compression on slot 3, VC 300 on serial port 1:

```
xp(config)# frame-relay set payload-compress ports se.3.1.300
```

frame-relay set peer-addr

Purpose

Set the peer address in case that InArp is not supported on the remote device.

Format

frame-relay set peer-addr [ip-address <*IP address*>**] [ipx-address** <*IPX address*>**] [ports** <*port list*>**]**

Mode

Configure

Description

Issuing the **frame-relay set peer-addr** command allows you to set the peer address if it can't be resolved by InArp.

Parameters

<*IP address*> The IP address you wish to use.

<*IPX address*> The IPX address you wish to use.

<*port list*> The location of the port to which you wish to assign the address.

Restrictions

Usage is restricted to frame relay VCs only.

Example

To assign an IP address 10.1.1.1/16 to slot 2, VC 100 on serial port 1:

```
xp(config)# frame-relay set peer-addr ip-address 10.1.1.1/16 ports se.2.1.100
```

frame-relay show service

Purpose

Displays frame relay service profiles.

Format

frame-relay show service <*service name*>|**all**

Mode

Enable

Description

The **frame-relay show service** command allows you to display the available frame relay service profiles.

Parameters

<*service name*>	The name of a particular pre-defined service profile.
all	Displays all of the available frame relay service profiles.

Restrictions

None.

Example

To display the available frame relay service profiles named “prof1”:

```
xp# frame-relay show service prof1
```

frame-relay show stats

Purpose

Displays frame relay statistics.

Format

frame-relay show stats ports <*port name*> **[last-error] [lmi] [mibII]**

Mode

Enable

Description

The **frame-relay show stats** command allows you to display the following frame relay port statistics for the given port:

- The last reported frame relay error.
- The active frame relay LMI parameters.
- The MIBII statistics for frame relay WAN ports.

Parameters

ports <*port name*>
The port or ports for which you want to display statistics.

last-error
Specifying the **last-error** keyword allows you to display the last reported frame relay error for the given port.

lmi
Specifying the **lmi** keyword allows you to displays the active frame relay LMI parameters.

mibII
Specifying the **mibII** keyword allows you to displays the MIBII statistics for frame relay WAN ports.

Restrictions

The **last-error**, **mibII**, and **lmi** commands are for ports only (no VC designators allowed). Otherwise, the port name may have the “VC” designator.

Examples

To display the last recorded error and MIB II statistics and for serial port 1 of slot 3:

```
xp# frame-relay show stats ports se.3.1 last-error mibII
```

To display the VC statistics for serial port 1, slot 3, VCs 1-10:

```
xp# frame-relay show stats ports se.3.1.1-10
```

frame-relay show stats summary

Purpose

Displays a summary of all VC statistics.

Format

frame-relay show stats summary ports <*port name*>

Mode

Enable

Description

The **frame-relay show stats summary** command allows you to display all of the summary information for VC statistics.

Parameters

<*port name*> The port or ports for which you wish to display summary statistics.

Restrictions

None.

Example

To display summary statistics for serial port 1 of slot 4, VC 100:

```
xp# frame-relay show stats summary ports se.4.1.100
```

Chapter 25

garp Commands

The following commands allow you to....

Command Summary

Table 20 lists the **garp** commands. The sections following the table describe the command syntax.

Table 20. garp commands

garp show timers <*port-list*>**| all-ports**
garp set timers <*port-list*> **join** <*num*>**| leave** <*num*>**| leaveall** <*num*>

garp show timers

Purpose

Display values of GARP timers.

Format

garp show timers <*port-list*> **| all-ports**

Mode

Enable

Description

The **garp show timers** command allows display of the values of the GARP join timer, leave timer, and leaveall timer for specified port(s).

Parameters

<*port-list*> **| all-ports**	Specifies port(s) for which to display GARP timer values. Entering **all-ports** will display timer values for every port.

Restrictions

None.

Example

To display values of GARP timers for port et.1.1:

```
er# garp show timers et.1.1
```

garp set timers

Purpose

Sets the values of GARP timers.

Format

garp set timers <*port-list*> **join** <*num*> | **leave** <*num*> | **leaveall** <*num*>

Mode

Configure

Description

The **garp set timers** command allows setting of GARP join, leave, and leaveall timers for specified port(s).

Parameters

port <*port-list*>	Specifies ports for which to display GARP timer values.
join <*num*>	Sets join timer to value specified. Value must fall between 20 and 1000 cs.
leave <*num*>	Sets leave timer to value specified. Value must fall between 60 and 3000 cs.
leaveall <*num*>	Sets leave all timer to value specified. Value must fall between 1000 and 18000 cs.

Restrictions

Timers should satisfy the following relationship: **Leave >= Join*3**, and **LeaveAll > Leave**.

Examples

To set GARP timers for port et.1.1:

```
xp# garp set timers et.1.1 join 1000 leave 2500 leaveall 18000
```

Chapter 26

mtrace Command

Purpose

Trace multicast path between a source and a receiver

Format

mtrace <*source*> **[destination** <*IPaddr*>**] [group** <*IPaddr*>**] [max-hops** <*number*>**]**

Mode

User

Description

The **mtrace** command tracks the multicast path from a source to a receiver. A trace probe is sent in a reverse path from the receiver back to the source. As the probe passes from hop to hop, it collects information such as interface address and packet counts from each router. If the **mtrace** command is executed with only the source parameter then a multicast path is calculated from the *source* to the X-Pedition. One can examine the multicast path between two external hosts by specifying a receiver instead of using the X-Pedition as the default receiver.

Parameters

<*source*>	IP address of the source.
destination <*IPaddr*>	Destination IP address.
group <*IPaddr*>	Multicast destination group address.
max-hops <*number*>	Maximum number of hops to trace (default: 0, range: 0-32)

Restrictions

None.

Examples

To display the multicast path from IP address 2.2.2.2 to the X-Pedition:

```
xp# mtrace 2.2.2.2
```

To display the multicast path from 1.1.1.1 to x.y.z.w for the group 239.1.1.1:

```
xp# mtrace 1.1.1.1 destination x.y.z.w group 239.1.1.1
```

Chapter 27

gvrp Commands

The following commands allow you to display and alter GVRP parameters on the X-Pedition.

Command Summary

Table 21 lists the **gvrp** commands. The sections following the table describe the command syntax.

Table 21. gvrp commands

gvrp show statistics <*port-list*> **| all ports**
gvrp show status
gvrp show registration-mode <*port-list*> **| all-ports**
gvrp show applicant-status <*port-list*> **| all-ports**
gvrp clear statistics <*port-list*> **| all-ports**
gvrp enable dynamic-vlan-creation
gvrp enable ports <*port-list*>
gvrp set registration-mode forbidden ports <*port-list*> **| all-ports**
gvrp set applicant status non-participant ports <*port-list*> **| all-ports**
gvrp start

gvrp show statistics

Purpose

Displays various GVRP statistics for ports.

Format

gvrp show statistics <*port-list*> **| all ports**

Mode

Enable

Description

The **gvrp show statistics** command displays statistics for the specified port(s).

Parameters

None.

Restrictions

None.

Example

```
gvrp show statistics et.1.1-2
```

gvrp show status

Purpose

Shows status of GVRP.

Format

gvrp show status

Mode

Enable

Description

The **gvrp show status** command displays current status of GVRP.

Parameters

None.

Restrictions

None.

Example

```
gvrp show status
```

gvrp show registration-mode

Purpose

Shows GVRP registration-mode for specified port(s).

Format

gvrp show registration-mode ports <*port-list*> **| all-ports**

Mode

Enable

Description

The **gvrp show registration-mode** command displays the GVRP registration-mode of specified port(s).

Parameters

<*port-list*>	Specifies port(s) for which information will display.
all-ports	Displays information for all X-Pedition ports.

Restrictions

None.

Example

```
gvrp show registration-mode ports et.1.1-2
```

gvrp show applicant-status

Purpose

Shows GVRP applicant-status for specified port(s).

Format

gvrp show registration-mode ports <*port-list*> | **all-ports**

Mode

Enable

Description

The **gvrp show applicant-status** command displays the GVRP applicant-status of specified port(s).

Parameters

<*port-list*>	Specifies port(s) for which information will display.
all-ports	Displays information for all X-Pedition ports.

Restrictions

None.

Example

```
gvrp show registration-mode ports et.1.1-2
```

gvrp clear statistics

Purpose

Clears GVRP statistics for specified port(s).

Format

gvrp clear statistics <*port-list*> **| all-ports**

Mode

Enable

Description

The **gvrp clear statistics** command clears GVRP statistics for specified port(s).

Parameters

<*port-list*>	Specifies port(s) for which statistics will be cleared.
all-ports	Clears statistics on all X-Pedition ports.

Restrictions

None.

Example

```
gvrp clear statistics et.1.1-2
```

gvrp enable dynamic-vlan-creation

Purpose

Allows GVRP to dynamically create vlans.

Format

gvrp enable dynamic-vlan-creation

Mode

Configure

Description

The **gvrp enable dynamic-vlan-creation** command allows GVRP to dynamically create vlans on the X-Pedition. If not enabled, GVRP will continue to propagate vlans to other network devices; however, vlans will not be created dynamically.

Parameters

None.

Restrictions

None.

Example

To enable dynamic vlan creation:

```
xp(config)# gvrp enable dynamic-vlan-creation
```

gvrp enable ports

Purpose

Enables GVRP on specified port(s).

Format

gvrp enable ports <*port-list*>

Mode

Configure

Description

The **gvrp enable ports** command enables GVRP on specified port(s).

Parameters

<*port-list*> Specifies port(s) upon which GVRP will be enabled.

Restrictions

None.

Example

To enable GVRP on ports et.1.1 and et.1.2:

```
xp(config)# gvrp enable ports et.1.1 and et.1.2
```

gvrp set registration-mode forbidden

Purpose

Sets GVRP registration-mode for specified port(s).

Format

gvrp set registration-mode forbidden ports <*port-list*> | **all-ports**

Mode

Configure

Description

The **gvrp set registration-mode forbidden** command enables the user to disallow vlan registration on specified port(s).

Note: The system default allows vlan registration.

Parameters

<*port-list*>	Specifies port(s) on which vlan registration will be forbidden.
all-ports	Forbids vlan registration on all **X-Pedition** ports

Restrictions

None.

Example

To forbid registration on port et.1.1:

```
xp(config)# gvrp registration-mode forbidden ports et.1.1
```

gvrp set applicant-status non-participant

Purpose

Sets GVRP applicant-status for specified port(s).

Format

gvrp set applicant-status non-participant ports <*port-list*> **| all-ports**

Mode

Configure

Description

The **gvrp set applicant-status non-participant** command stops GVRP from propagating vlan information on specified port(s). These port(s) may still create vlans from other network devices if so configured.

Parameters

<*port-list*>	Specifies port(s) on which to halt vlan propagation.
all-ports	Halts vlan propagation on all **X-Pedition** ports

Restrictions

None.

Example

To set port et.1.1 to a non-participant:

```
xp(config)# gvrp applicant-status non-participant ports et.1.1
```

gvrp start

Purpose

Starts GVRP on the X-Pedition.

Format

gvrp start

Mode

Configure

Description

The **gvrp start** command enables GVRP.

Parameters

None.

Restrictions

None.

Example

To enable GVRP:

```
xp(config)# gvrp start
```

Chapter 28

igmp Commands

The **igmp** commands let you display and set IGMP parameters.

Command Summary

Table 22 lists the **igmp** commands. The sections following the table describe the command syntax.

Table 22. igmp commands

igmp enable interface <*name/ipAddr*>
igmp enable vlan <*vlan-name*>
igmp set interface <*name/ipAddr*> **[allowed-groups** <*group-list*>**|not-allowed-groups** <*group-list*>**] [use-all-ports]**
igmp join group <*ipAddr*> **interface** <*name/ipAddr*>
igmp set queryinterval <*num*>
igmp set responsetime <*num*>
igmp set vlan <*vlan-name*> [**host-timeout** <*num*>] [**querier-timeout** <*num*>] [**router-timeout** <*num*> **leave-timeout** <*num*>]
igmp show interfaces [group <*ipAddr*>**|interface** <*name/ipAddr*>**]**
igmp show memberships [group <*ipAddr*>|*port* <*num*>**]**
igmp show timers
igmp show vlans
igmp start-snooping

igmp enable interface

Purpose

Enables IGMP on an interface.

Format

igmp enable interface <*name/ipAddr*>

Mode

Configure

Description

The **igmp enable interface** command enables IGMP on the specified interface.

Parameters

<*name/ipAddr*> Name or IP address of the interface on which you are enabling IGMP.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

- IGMP is not enabled on tunnels.
- Because DVMRP and PIM-SM run in separate processes on the X-Pedition, current IGMP functionality may be used only with DVMRP. PIM-SM must use a separate group of commands called “PIM IGMP.”

Example

To enable IGMP on interface 10.50.1.2:

```
xp(config)# igmp enable interface 10.50.1.2
```

igmp enable vlan

Purpose

Enables IGMP snooping on a VLAN.

Format

igmp enable vlan <*vlan-name*>

Mode

Configure

Description

The **igmp enable vlan** command enables IGMP snooping on a specified VLAN. By default, IGMP snooping is disabled on all VLANs.

Note: The **igmp start-snooping** command must be present for the **igmp enable vlan** command to function properly, and the **igmp start-snooping** command supports IGMP-enabled VLANs only—it is not intended for use with IGMP-enabled interfaces. See *igmp start-snooping* on page 371.

Parameters

<*vlan-name*>
Is the name of the VLAN where IGMP snooping is to be enabled.

Note: The X-Pedition will display VLAN names up to 32 characters in length.

Restrictions

Layer 3 multicasting and layer-2 snooping cannot be run simultaneously on the same VLAN.

Example

To enable igmp snooping on VLAN blue:

```
xp(config)# igmp enable vlan blue
```

igmp set interface

Purpose

Configures IGMP parameters.

Format

igmp set interface <*name/ipAddr*>
[allowed-groups <*group-list*>|**not-allowed-groups** <*group-list*>**] [use-all-ports]**

Mode

Configure

Description

Sets IGMP parameters on a per-interface basis to control group restrictions and optimization.

Parameters

interface <*name/ipAddr*>
The name of the interface.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

allowed-groups <*group-list*>
Restricts the groups to only those specified.

not-allowed-groups <*group-list*>
Allows any groups besides those specified.

Note: Specify only one of the above options, as they are mutually exclusive.

use-all-ports
Disables per-port IGMP control. By default, per-port IGMP control is enabled.

Note: If the traffic is being supplied by a dvmrp tunnel, which uses CPU-based switching, then for efficiency reasons, port based optimization is not used by this traffic.

Restrictions

None.

Examples

The following is an example of the **igmp set interface** command:

```
xp(config)# igmp set interface 200.1.1.1 allowed-groups 225.2.0.0/16
```

The above command will allow only memberships to groups falling in the specified range. Outside this range, all groups are implicitly ignored.

igmp join group

Purpose

Allows you to configure a static igmp group onto an interface.

Format

igmp set join group <*ipAddr*> **interface** <*name/ipAddr*>

Mode

Configure

Description

The **igmp set join group** command allows an interface to join an igmp group statically. Most interfaces join igmp groups dynamically, outside the control of the user. This command allows the user to configure an igmp group onto an interface statically.

Parameters

group <*ipAddr*>
Specifies the multicast address.

interface <*name/ipAddr*>
Specifies the interface name or IP address.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

None.

Examples

To configure the igmp group ‘255.2.0.0’ on interface 200.1.1.1:

```
xp(config)# igmp set join group 255.2.0.0 interface 200.1.1.1
```

igmp set queryinterval

Purpose

Configures IGMP Host Membership Query interval.

Format

igmp set queryinterval <*num*>

Mode

Configure

Description

Sets the IGMP Host Membership Query time interval. The interval you set applies to all ports on the X-Pedition.

Parameters

<*num*> A value from 20 – 3600 seconds. The default is 125 seconds.

Restrictions

None.

Example

To set the query interval to 30 seconds:

```
xp(config)# igmp set queryinterval 30
```

igmp set responsetime

Purpose

Configures IGMP Host Membership response wait time.

Format

igmp set responsetime <*num*>

Mode

Configure

Description

Sets the wait time for IGMP Host Membership responses. The wait time you set applies to all ports on the X-Pedition.

Parameters

<*num*> Response wait time in seconds. Specify a number from 10 – 3599. The default is 10.

Restrictions

None.

Examples

To set the Host Membership response wait time to 20 seconds:

```
xp(config)# igmp set responsetime 20
```

igmp set vlan

Purpose

Sets parameters for IGMP snooping on a VLAN.

Format

igmp set vlan <*vlan-name*> [**host-timeout** <*num*>] [**querier-timeout** <*num*>] [**router-timeout** <*num*> **leave-timeout** <*num*>] [**filter-ports** <*port-list*>] [**permanent-ports** <*port-list*>]

Mode

Configure

Description

The **igmp set vlan** command allows you to set parameters for VLAN-based IGMP snooping.

Parameters

<*vlan-name*>
The name of the VLAN for which you will set IGMP snooping parameters.

Note: The X-Pedition will display VLAN names up to 32 characters in length.

host-timeout <*num*>
Allows adjusting to long host timeout values that may have been set up for the IGMP querier. The default value is 250 seconds.

querier-timeout <*num*>
Allows adjusting to long timeout values that may have been set up for the IGMP querier. The default value is 260 seconds.

router-timeout <*num*>
Allows adjusting to long timeout values that may have been set up for the routers. Different versions of DVMRP can have different time outs. The default value is 140 seconds.

leave-timeout <*num*>
Allows quicker timeout if IGMP v2 leave messages are used. The value is nominally 10 seconds.

filter-ports <*port-list*>
Allows forced filtering of certain ports from multicast data. Setting ports as filter ports ensures that no host there will join any memberships. A port can optionally be either a permanent port or a filter port, but not both.

permanent-ports <*port-list*>
Allows forcing of mulicast data if present on certain ports. A port can optionally be either a permanent port or a filter port, but not both.

Restrictions

None.

Example

To set parameters for IGMP snooping on the VLAN blue:

```
xp(config)# igmp set vlan blue host-timeout 125 querier-timeout 130 router-timeout 70
```

igmp show interfaces

Purpose

Shows the interfaces running IGMP.

Format

igmp show interfaces [group <*ipAddr*>**|interface** <*name/ipAddr*>**]**

Mode

Enable

Description

The **igmp show interfaces** command shows memberships on a specified interface or for a multicast group address. When you use the command to show interfaces by group, all interfaces containing the group membership are shown.

Note: This command is similar to **igmp show memberships**, except where the **igmp show interfaces** command shows interface details, the **igmp show memberships** command shows ports.

Parameters

group <*ipAddr*> Address of a multicast group.

interface <*name/ipAddr*>Name or address of a interface.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

None.

Example

To show information about the interfaces running IGMP:

```
xp# igmp show interfaces

Address: 172.1.1.10 Subnet: 172.1.1/24 Met: 1 Thr: 1
Name : mls15 State: Up Querier Leaf Igmp Dvmrp

Address: 207.135.89.64 Subnet: 207.135.89.0/25 Met: 1 Thr: 1
Name : company State: Up Querier Leaf Igmp Dvmrp
Groups : 224.0.1.12
224.1.127.255
224.0.1.24
224.2.127.253
224.2.127.254

Address: 10.135.89.10 Subnet: 10.135.89.0/25 Met: 1 Thr: 1
Name : test State: Up Querier Igmp Dvmrp

Address: 207.135.122.11 Subnet: 207.135.122.8/29 Met: 1 Thr: 1
Name : mbone State: Up Igmp Dvmrp
Groups : 224.0.1.11
224.0.1.12
224.2.127.254
239.255.255.255
224.2.127.253
```

igmp show memberships

Purpose

Displays IGMP host memberships.

Format

igmp show memberships [group <*ipAddr*>|*port* <*num*>**]**

Mode

Enable

Description

The **igmp show memberships** command displays IGMP host members on a specific interface and/or for a particular multicast group.

Parameters

group <*ipAddr*>	Address of the multicast group for which to display host memberships.
port <*num*>	Port numbers on which the members reside.

Restrictions

None.

Examples

To display host members for multicast group 225.0.1.20:

```
xp(config)# igmp show memberships group 225.0.1.20
```

To display host members for multicast group 225.0.1.20 on port et.1.1:

```
xp(config)# igmp show memberships group 225.0.1.20 port et.1.1
```

The following is a fuller example.

```
xp(config)# igmp show memberships

Group : 224.0.1.11 Ports: et.1.1
Group : 224.0.1.12 Ports: et.1.1
et.5.1
Group : 224.0.1.24 Ports: et.5.1
Group : 224.1.127.255 Ports: et.5.1
Group : 224.2.127.253 Ports: et.1.1
et.5.1
Group : 224.2.127.254 Ports: et.1.1
et.5.1
Group : 239.255.255.255 Ports: et.1.1
```

igmp show timers

Purpose

Displays IGMP timers.

Format

igmp show timers

Mode

Enable

Description

The **igmp show timers** command displays IGMP timers.

Parameters

None.

Restrictions

None.

igmp show vlans

Purpose

Displays IGMP VLANs.

Format

igmp show vlans [detail] [name <*name*>**] [timers]**

Mode

Enable

Description

The **igmp show vlans** command displays IGMP VLANs.

Parameters

detail Shows all IGMP membership information

name <*name*> Shows IGMP membership information for the specified VLAN

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display VLAN and interface names up to 32 characters in length.

timers Shows all IGMP L2 snooping related timers

Restrictions

None.

igmp start-snooping

Purpose

Starts passive IGMP snooping on enabled VLANs.

Format

igmp start-snooping

Mode

Configure

Description

The **igmp start-snooping** command starts IGMP snooping on enabled VLANs. This task is independent of L3 multicasting.

Note: The **igmp start-snooping** command must be present for the **igmp enable vlan** command to function properly, and the **igmp start-snooping** command supports IGMP-enabled VLANs only—it is not intended for use with IGMP-enabled interfaces. See *igmp enable vlan* on page 357.

Parameters

None.

Restrictions

None.

Chapter 29

interface Commands

The **interface** commands allow the user to create AppleTalk, IP, and IPX interfaces. They also allow the addition of network mask and broadcast address information to existing IP interfaces, and they display configuration information for AppleTalk, IP, and IPX interfaces.

Command Summary

Table 23 lists the **interface** commands. The sections following the table describe the command syntax.

Table 23. interface commands

interface add appletalk <*InterfaceName*> **zone** <*ZoneName*> **[default]**
interface add ip <*InterfaceName*> **address-netmask** <*IPaddr-mask*> **peer-address [**<*IPaddr*>**] [broadcast** <*IPaddr*>**]**
interface add ipx <*InterfaceName*> **address** <*IPXaddr*> **[peer-address** <*IPXaddr*>**] [output-mac-encapsulation** <*MACencap*>**]**
interface create appletalk <*InterfaceName*> **vlan** <*name*>**|port** <*port*> **cable-range** <*range*> **[zone** <*ZoneName*>**] [address** <*Net.Node*>**] [up|down]**
interface create appletalk <*InterfaceName*> **vlan** <*name*>**|port** <*port*> **noseed [up|down]**
interface create ip <*InterfaceName*> **address-netmask** <*IPaddr-mask*> **[broadcast** <*IPaddr*>**] [peer-address** <*IPaddr*>**] vlan** <*name*>**|port** <*port*> **mtu** <*num*> **[output-mac-encapsulation** <*MACencap*>**] [up|down] [mac-addr** <*MACaddr-spec*>**]**
interface create ipx <*InterfaceName*> **address** <*IPXaddr*> **peer-address [**<*IPXaddr*>**] vlan** <*name*> **| port** <*port*> **[output-mac-encapsulation** <*MACencap*>**] [up|down] [mac-addr** <*MACaddr-spec*>**] [mtu** <*num*>**]**

Table 23. interface commands (Continued)

interface show appletalk <*InterfaceName*>**|all [brief]**
interface show ip <*InterfaceName*>**|all [brief]**
interface show ipx <*InterfaceName*>**|all [brief]**

interface add appletalk

Purpose

Adds zones to an existing AppleTalk interface.

Format

interface add appletalk <*InterfaceName*> **zone** <*ZoneName*> **[default]**

Mode

ARE-Configure

Description

The **interface add appletalk** command configures additional zones for an existing interface.

Note: The interface must already exist. To create an interface, enter the **interface create appletalk** command.

Parameters

<*InterfaceName*> Name of the IP interface; for example, int4.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

<*ZoneName*> Name of the additional zone; for example, “myzone.” Up to 253 zones may be assigned to an interface.

default Changes the default zone for the local network attached to the interface. Specified zone name will become the new default zone. If this parameter is not used, the default zone will not change.

Restrictions

You must be in ARE-Configure mode before using this command. To learn more about this mode, please see Chapter 6, *are Commands*.

You can use this command only on an interface that has already been created using the **interface create appletalk** command.

Example

To configure an additional zone with the name myzone, and to make it the new default:

```
xp(are-config)# interface add appletalk int4 zone myzone default
```

interface add ip

Purpose

Configure secondary addresses for an existing interface.

Format

interface add ip <*InterfaceName*> **address-netmask** <*IPaddr-mask*> **peer-address [**<*IPaddr*>**]**
[broadcast <*IPaddr*>**]**

Mode

Configure

Description

The **interface add ip** command configures secondary addresses for an existing IP interface. Use this command to configure a secondary IP address and netmask, a secondary peer-address, and a secondary broadcast address.

Note: The interface must already exist. To create an interface, enter the **interface create ip** command.

Parameters

<*InterfaceName*> Name of the IP interface; for example, int4.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

address-netmask Secondary IP address and netmask of this interface. You can specify the address and mask information using the traditional format (example: 10.1.2.3/255.255.0.0) or the CIDR format (example: 10.1.2.3/16). If you specify an address without mask information, the X-Pedition uses the natural mask for the address (/8 for Class A, /16 for Class B or /24 for Class C).

peer-address Secondary IP address of the peer for this port. Primarily used for setting up connection with another WAN port or setting up a VC with another ATM port.
For WAN and ATM ports only.

broadcast <*IPaddr*> Secondary broadcast address of this interface.

Restrictions

You can use this command only on an interface that has already been created using the **interface create ip** command.

Example

To configure a secondary address of 10.23.4.36 with a 24-bit netmask (255.255.255.0) on the IP interface int4:

```
xp(config)# interface add ip int4 address-mask 10.23.4.36/24
```

interface add ipx

Purpose

Configure secondary addresses for an existing IPX interface.

Format

interface add ipx <*InterfaceName*> **address** <*IPXaddr*> **[peer-address** <*IPXaddr*>**] [output-mac-encapsulation** <*MACencap*>**]**

Mode

Configure

Description

The **interface add ipx** command configures secondary addresses for an existing IPX interface.

Note: The interface must already exist. To create an interface, enter the **interface create ipx** command.

Parameters

<*InterfaceName*> Name of the IP interface; for example, int4.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

address Secondary IPX network address of this interface, specified in a hexadecimal number.

peer-address Secondary IPX address of the peer for this port. Primarily used for setting up connection with another WAN port. The **peer-address** contains the network address, a period (**.**), then the mac address. This can be illustrated as follows: **a1b2c3d4.aa:bb:cc:dd:ee:ff**
For WAN ports only.

output-mac-encapsulation
The output MAC encapsulation associated with this interface. You can specify one of the following:

–**ethernet_ii** (the default)

–ethernet 802.3

–ethernet_snap

–ethernet_802.2_ipx

Restrictions

- You can use this command only on an interface that has already been created using the **interface create ipx** command.
- IPX is not supported in partially meshed WAN networks unless each node has a unique network address.

Example

To configure a secondary address of 10 (hexadecimal) on the IPX interface int4 with an 802.3 output encapsulation scheme:

```
xp(config)# interface add ipx int4 address 10 output-mac-encapsulation ethernet_802.3
```

interface create appletalk

Purpose

Creates an AppleTalk interface.

Format

interface create appletalk <*InterfaceName*> **vlan** <*name*>|**port** <*port*> **cable-range** <*range*> **[zone** <*ZoneName*>**] [address** <*Net.Node*>**] [up|down]**

Mode

ARE-Configure

Description

The **interface create appletalk** command creates and configures an AppleTalk/ARE interface. Configuration of an AppleTalk interface can include creating an interface in a disabled (**down**) state instead of the default enabled (**up**) state. In using this command, you *must* specify a cable range. Otherwise, you should use the **interface create appletalk noseed** command.

Interfaces on the X-Pedition are logical interfaces. Therefore, you can associate an interface with a single port or with multiple ports.

- To associate an interface with a single port, use the **port** option with the **interface create appletalk** command.
- To associate an interface with multiple ports, first create a VLAN and add ports to it, then use the **vlan** option with the **interface create appletalk** command.

Note: You must use either the **port** option or the **vlan** option with the **interface create appletalk** command.

Parameters

<*InterfaceName*>
Name of the AppleTalk interface; for example, int4.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

vlan <*name*>
Name of the VLAN associated with this interface.

Note: The X-Pedition will display VLAN names up to 32 characters in length.

port <*port*>
The port associated with this interface.

cable-range <*range*>
Sets the range of network numbers assigned to this interface. Valid numbers include 1 to 65279. Cable ranges cannot overlap across interfaces that are part of the same network. For example, if one interface has the cable range 1-100 assigned to it, no other interface on the network may have any cable range that includes any number between 1 and 100. A cable range must also be continuous. For example, a cable range of 1-50, 60-100 is invalid.

zone <*ZoneName*>
Sets the default zone (up to 32 characters) for the local network connected to the interface. This default zone can be changed using the **interface add appletalk zone** command. If no zone is specified, a default zone will automatically be assigned to the interface.

address <*Net.Node*>
Assigns the network and node number to this interface. Valid network numbers range from 1 to 65279. Valid node numbers range from 1 to 253. If no address is specified, a random valid address will automatically be assigned to the interface.

Note: The network number must lie within the previously specified **cable-range**. For example, if you set the cable range value at 1-2, then an appropriate network number would be 1 (i.e. 1.121). The address 3.121 would be considered inappropriate.

up
Sets the state of the interface to up. The interface will activate and attempt to pass traffic. (This is the default state.)

down
Sets the state of the interface to down. The interface will be created, however it will pass no traffic.

Restrictions

You must be in ARE-Configure mode before using this command. To learn more about this mode, please see Chapter 6, *are Commands*.

Note: If you use a VLAN to create an AppleTalk interface, you must use an AppleTalk procotol-based VLAN.

Examples

To create an interface called “app7” with the cable range 100-1100 and address 1050.88, enter the following command. The interface is associated with port et.1.3.

```
xp(are-config)# interface create appletalk app7 port et.1.3 cable-range 100-1100 address 1050.88
```

To create an interface called “app1” associated with the VLAN called “marketing” and a cable range of 10-200, enter the following command. The interface is created in the down (disabled) state.

```
xp(are-config)# interface create appletalk app1 vlan marketing cable-range 10-200 down
```

interface create appletalk noseed

Purpose

Creates a seeded AppleTalk interface.

Format

interface create appletalk <*InterfaceName*> **vlan** <*name*>|**port** <*port*> **noseed [up|down]**

Mode

ARE-Configure

Description

The **interface create appletalk noseed** command creates and configures a seeded AppleTalk interface. This means that, upon creation, the interface will attempt to "seed" itself on the network. It will take its cable range and default zone from another network router. This command should be used when ever two routers are connected. One router will act as the "seed," giving a cable range and default zone to the secondary, "non-seeded" router.

Configuration of a seeded AppleTalk interface can also include creating an interface in a disabled (**down**) state instead of the default enabled (**up**) state.

Interfaces on the X-Pedition are logical interfaces. Therefore, you can associate an interface with a single port or with multiple ports.

- To associate an interface with a single port, use the **port** option with the **interface create appletalk noseed** command.
- To associate an interface with multiple ports, first create a VLAN and add ports to it, then use the **vlan** option with the **interface create appletalk noseed** command.

Note: You must use either the **port** option or the **vlan** option with the **interface create appletalk noseed** command.

Parameters

<*InterfaceName*>
Name of the AppleTalk interface; for example, int4.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

vlan <*name*>
Name of the VLAN associated with this interface.

Note: The X-Pedition will display VLAN names up to 32 characters in length.

port <*port*>
The port associated with this interface.

noseed
Prompts the interface to attempt to "seed" itself on the network. This parameter should be used in place of the **cable-range** and **zone** parameters, since the interface will attempt to gain a cable range and zone from another router on the network.

up
Sets the state of the interface to up. The interface will activate and attempt to pass traffic. (This is the default state.)

down
Sets the state of the interface to down. The interface will be created, however it will pass no traffic.

Restrictions

You may only use this command in ARE-Configure mode. For more about this mode, please see Chapter 6, *are Commands*.

Examples

To create an interface called "aps8" which will attempt to seed itself on the network, enter the following command. The interface is associated with port et.1.5.

```
xp(are-config)# interface create appletalk aps8 port et.1.5 noseed
```

interface create ip

Purpose

Create an IP interface.

Format

interface create ip <*InterfaceName*> **address-netmask** <*IPaddr-mask*> **[broadcast** <*IPaddr*>**]**
[peer-address <*IPaddr*>**] vlan** <*name*>**|port** <*port*> **mtu** <*num*>
[output-mac-encapsulation <*MACencap*>**] [up|down] [mac-addr** <*MACaddr-spec*>**]**
[type broadcast|point-to-point]

Mode

Configure

Description

The **interface create ip** command allows you to create and configure an IP interface name, IP address, netmask, broadcast address, and the (subnet) mask to use when you create the interface.

Note: When you create an IP interface, the **interface create ip** command requires a logical name for each interface. If you use an IP interface name that begins with **en** or **lo**, the X-Pedition will disable the proxy ARP on the interface.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display VLAN and interface names up to 32 characters in length.

If you define a class C address and do not specify a mask, the router will assign the 24-bit (255.255.255.0) class C mask and the broadcast address will set all 8 host bits to 1. The XP also allows you the flexibility to use variable-length subnet masking in your network. If you specify a 27-bit (255.255.255.224) mask to have 6 subnets, the router broadcast will set all 5 host bits to 1. Traditional routers use this same functionality.

Router interface address:

```
172.16.1.129/27
172.16.1.129/255.255.255.224
```

Broadcast address with traditional routing:

```
172.16.1.159
```

Host interface address:

```
172.16.1.140/24
172.16.1.159/255.255.255.0
```

Broadcast address with traditional routing:

```
172.16.1.255
```

If the hosts on this network use the class C mask, they will recognize the traditional router broadcast as another host address (the hosts would use the traditional 172.16.1.255 for broadcasts). In this situation, you could configure the router broadcast to 172.16.1.255—then the hosts on the network would recognize the broadcast address. Do not specify the broadcast address if all devices on the subnet are using the same mask.

The X-Pedition is pre-allocated a pool of 64 MAC addresses. By default, the X-Pedition configures each new IP interface with the *base* MAC address—the lowest MAC address in the pool. However, you can use the **mac-addr** option to assign a different MAC address to an interface. Interfaces on the X-Pedition are logical interfaces; therefore, you can associate an interface with a single port or with multiple ports. You can also create an interface in a *disabled* (**down**) state instead of the default *enabled* (**up**) state.

- To associate an interface with a single port, use the **port** option with the **interface create** command.
- To associate an interface with multiple ports, first create an IP VLAN and add ports to it, then use the vlan option with the **interface create** command.

Note: You must use either the **port** option or the **vlan** option with the **interface create** command.

Parameters

<InterfaceName>
Name of the IP interface; for example, int4.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

address-netmask
IP address and netmask of this interface. You can specify the address and mask information using the traditional format (example: 10.1.2.3/255.255.0.0) or the CIDR format (example: 10.1.2.3/16). If you specify an address without mask information, the X-Pedition uses the natural mask for the address (/8 for Class A, /16 for Class B or /24 for Class C).

broadcast
IP address and netmask of this interface. You can specify the address and mask information using the traditional format (example: 10.1.2.3/255.255.0.0) or the CIDR format (example: 10.1.2.3/16). If you specify an address without mask information, the X-Pedition uses the natural mask for the address (/8 for Class A, /16 for Class B or /24 for Class C).

peer-address
IP address of the peer for this port. Primarily used for setting up connection with another WAN port or setting up a VC with another ATM port.
For WAN and ATM ports only.

vlan <*name*>
Name of the VLAN associated with this interface.

Note: The X-Pedition will display VLAN names up to 32 characters in length.

port <*port*>
Port associated with this interface.

mtu
Sets the MTU or *Maximum Transmission Unit* (in bytes) for this interface. By default, this value is equal to the MTU of the physical port minus some Layer-2 overhead (usually 22 bytes). However, users may set the interface MTU to anything less than the MTU of the physical port minus the Layer-2 overhead. In cases where an interface is assigned to a VLAN or SmartTRUNK, the interface MTU must be less than the MTU of the port in the VLAN or SmartTRUNK with the lowest value. When working with jumbo-frame capable ports, users may first need to use the **port set** <*port*> **mtu** <*mtu*> command to increase the physical port MTU size.

up
Sets the state of the interface to up. (This is the default state.)

down
Sets the state of the interface to down.

output-mac-encapsulation
The output MAC encapsulation associated with this interface. You can specify one of the following:

- **ethernet_ii** (the default)
- **ethernet_snap**

mac-addr <*MACaddr-spec*>
Sets the MAC address for this interface. You can specify one of the following:

- A specific MAC address – specify the entire MAC address as follows: xx:xx:xx:xx:xx:xx
- An offset from the base MAC address in the pool – specify the offset. For example, to specify an offset of 10 from the base MAC address, enter “10”. For example, if the base MAC address is 00:E0:63:02:00:00 and you specify an offset of 10, the X-Pedition assigns MAC address 00:E0:63:02:00:0A to the interface.
- The base MAC address – specify the **basemac** keyword. This is the default.

type broadcast| point-to-point
Sets the type of interface. Specify one of the following:

- **broadcast** (the default)
- **point-to-point** (the default for PPP)

Note: If you connect the ATM interface to a router that uses a firmware version older than 8.2.0.0, you will need to set the interface type to point-to-point.

Restrictions

You must enter the peer address to set the ATM interface type to point-to-point.

Examples

To create a VLAN called IP3, add ports et.3.1 through et.3.4 to the VLAN, then create an IP interface on the VLAN:

```
xp(config)# vlan create IP3 ip
xp(config)# vlan add ports et.3.1-4 to IP3
xp(config)# interface create ip int3 address-mask 10.20.3.42/24 vlan IP3
```

To create an interface called “int7” with the address 10.50.89.88 and a 16-bit subnet mask, enter the following command. The interface is associated with port et.1.3.

```
xp(config)# interface create ip int7 address-mask 10.50.89.88/16 port et.1.3
```

To create an interface called “int1” with a broadcast address of 10.10.42.255, enter the following command. The interface is associated with the VLAN called “marketing”. The interface is created in the *down* (disabled) state.

```
xp(config)# interface create ip int1 address-mask 10.10.42.17/255.255.255.0 broadcast 10.10.42.255
    vlan marketing down
```

interface create ipx

Purpose

Create an IPX interface.

Format

interface create ipx <*InterfaceName*> **address** <*IPXaddr*> **peer-address [**<*IPXaddr*>**] vlan** <*name*> **| port** <*port*> **[output-mac-encapsulation** <*MACencap*>**] [up|down] [mac-addr** <*MACaddr-spec*>**] [mtu** <*num*>**]**

Mode

Configure

Description

The **interface create ipx** command creates and configures an IPX interface. Configuration of an IPX interface can include information such as the interface's name, IPX address, VLAN, port, and output MAC encapsulation. You can also create an interface in the disabled (**down**) state instead of the default enabled (**up**) state.

The X-Pedition is pre-allocated a pool of 64 MAC addresses (the maximum). By default, each new IPX interface is automatically configured with the lowest MAC address in the pool (the "base" MAC address). However, you can assign an interface a different MAC address by using the **mac-addr** option.

Parameters

<*InterfaceName*>
Name of the IPX interface; for example, int9.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

address <*ipxAddr*>
IPX address of this interface.

peer-address
IPX address of the peer for this port. Primarily used for setting up connection with another WAN port. The **peer-address** contains the network address, a period (**.**), then the mac address. This can be illustrated as follows: **a1b2c3d4.aa:bb:cc:dd:ee:ff**
For WAN ports only.

vlan <*name*>
Name of the VLAN associated with this interface.

Note: The X-Pedition will display VLAN names up to 32 characters in length.

port <*port*>
Port associated with this interface.

up
Sets the state of the interface to up. (This is the default state.)

down
Sets the state of the interface to down.

output-mac-encapsulation
The output MAC encapsulation associated with this interface. You can specify one of the following:

- **ethernet_ii** (the default)
- **ethernet_snap**
- **ethernet_802.2_ipx**

Note: When using line cards introduced prior to the "AA" series, SNA/DLC/NetBIOS traffic may not bridge properly. The issue in bridging DLC packets occurs where the length field within an IEEE 802.3 frame indicates less than 46 bytes of data.

The X-Pedition removes the length field information of incoming IEEE 802.3, 802.2, and Ethernet SNAP packets, then recalculates the field prior to re-transmission. Consequently, the calculation is based on the length of the entire data field. A packet entering the X-Pedition whose length field indicates a data field of less than 46 bytes will exit with the length field recalculated incorrectly. This can be a problem with LLC2 and legacy IPX applications. Typically, such packets exist only in SNA and NetBIOS/NetBEUI environments.

mac-addr <*MACaddr-spec*>
Sets the MAC address for this interface. You can specify one of the following:

- A specific MAC address – specify the entire MAC address as follows: xx:xx:xx:xx:xx:xx or xxxxxx:xxxxxx
- An offset from the IPX base MAC address (base+2) in the pool – specify the offset. For example, to specify an offset of 10 from the IPX base MAC address, enter "10". If, for example, the IPX base MAC address is 00:E0:63:02:00:00 and you specify an offset of 10, the X-Pedition assigns IPX MAC address 00:E0:63:02:00:0A to the interface. You may enter any number between 1 and 61.

  Note: The X-Pedition has a pool of 64 MAC addresses (base to base+63); by default, IPX uses (base+2). If you would like to use another of the available MAC addresses, enter a number between 1 and 61; the interface MAC will then become ((IPXbaseMAC) + <*number*>).

- The IPX base MAC address – specify the **basemac** keyword. This is the default, therefore you need only enter the keyword in order to make the selection explicit.

mtu

Sets the MTU or *Maximum Transmission Unit* (in bytes) for this interface. By default, this value is equal to the MTU of the physical port minus some Layer-2 overhead (usually 22 bytes). However, users may set the interface MTU to anything less than the MTU of the physical port minus the Layer-2 overhead. In cases where an interface is assigned to a VLAN or SmartTRUNK, the interface MTU must be less than the MTU of the port in the VLAN or SmartTRUNK with the lowest value. When working with jumbo-frame capable ports, users may first need to use the **port set** <*port*> **mtu** <*mtu*> command to increase the physical port MTU size.

Restrictions

IPX is not supported in partially meshed WAN networks unless each node has a unique network address.

Examples

The following commands create a VLAN called IPX10, add all the ports on the line card in slot 1 to the VLAN, and create an IPX interface called "int10" with the IPX address a98d7c6f, associated with VLAN IPX10.

```
xp(config)# vlan create IPX10 ipx
xp(config)# vlan add ports et.1.* to IPX10
xp(config)# interface create ipx int10 address a98d7c6f vlan IPX10
```

The following command creates an interface called "int5" with the IPX address 82af3d57 for port et.1.3. The interface is added in the down (disabled) state.

```
xp(config)# interface create ipx int5 address 82af3d57 port et.1.3 down
```

To create an interface called "int6" with the MAC address 00:01:02:03:04:05 and IPX address 82af3d58 for port et.1.4.

```
xp(config)# interface create ipx int6 address 82af3d58 port et.1.4
    mac-addr 00:01:02:03:04:05
```

To create an interface called "int7" for a VLAN called "IPX-VLAN" on port et.1.4 with the MAC address at the base of the X-Pedition's MAC address pool:

```
xp(config)# interface create ipx int7 address 82af3d59 vlan IPX-VLAN et.1.4 mac-addr basemac
```

The following command creates an interface called "int7" for a VLAN called "IPX-VLAN" on port et.1.4 with a MAC address offset by 10 from the base of the X-Pedition's MAC address pool. If the

base MAC address in the X-Pedition's MAC address pool is 00:E0:63:02:00:00, the offset of 10 gives the interface the MAC address 00:E0:63:02:00:0A.

```
xp(config)# interface create ipx int7 address 82af3d59 vlan IPX-VLAN et.1.4 mac-addr 10
```

The following commands create an ATM virtual channel on an ATM port and associate the port with an IPX interface. This allows IPX routing between two IPX interfaces. As with any IPX interface, IPX routing using RIP (the default) will begin when you configure an IPX interface.

```
xp(config)# atm create vcl port at.3.1.1.100
xp(config)# interface create ipx finance address 01234567 peer-address 01234567.00:00:1d:a9:8c:a1
port at.3.1.1.100
xp(config)# interface create ipx marketing address 01234569 port et.1.1
```

interface show appletalk

Purpose

Displays configuration of all AppleTalk interfaces.

Format

interface show appletalk <*InterfaceName*> **| all [brief]**

Mode

Enable

Description

The **interface show appletalk** command displays configuration information for all AppleTalk/ARE interfaces defined on the system.

Parameters

<*InterfaceName*> **| all**
Name of the AppleTalk interface; for example, app4. Specify **all** to show configuration information about all AppleTalk interfaces on the X-Pedition.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

brief
Display a brief summary of the interface in tabular form.

Restrictions

None.

Examples

To display configuration information for the AppleTalk interface called “app7”:

```
xp# interface show appletalk app7
```

.To display configuration information for all AppleTalk interfaces:

```
xp# interface show appletalk all
```

interface show ip

Purpose

Display configuration of an IP interface.

Format

interface show ip <*InterfaceName*> **| all [brief]**

Mode

Enable

Description

The **interface show ip** command displays configuration information for an IP interface.

Note: You can display exactly the same information from within the ip facility using the **ip show interfaces** command.

Parameters

<*InterfaceName*> **| all**
Name of the IP interface; for example, int4. Specify **all** to show configuration information about all the IP interfaces on the X-Pedition.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

brief Displays a brief summary of the interface in tabular form.

Restrictions

None.

Examples

To display configuration information for the IP interface called “int7”:

```
xp# interface show ip int7
```

.To display configuration information for all IP interfaces:

```
xp# interface show ip all
```

interface show ipx

Purpose

Display configuration of an IPX interface.

Format

interface show ipx <*InterfaceName*> **| all [brief]**

Mode

Enable

Description

The **interface show ipx** command displays configuration information for an IPX interface.

Note: You can display exactly the same information from within the ip facility using the **ipx show interfaces** command.

Parameters

<*InterfaceName*> **| all**
Name of the IPX interface; for example, int9. Specify **all** to show configuration information about all the IPX interfaces on the X-Pedition.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

brief Shows a brief summary of the interface in tabular form.

Restrictions

None.

Examples

To display configuration information for the IPX interface called “int8”:

```
xp# interface show ipx int8
```

To display configuration information for all IPX interfaces:

```
xp# interface show ipx all
```

Chapter 30

ip Commands

The **ip** commands let you display route table entries and various IP related tables.

Command Summary

Table 24 lists the **ip** commands. The sections following the table describe the command syntax.

Table 24. ip commands

ip add route <*ipAddr-mask*>**|default gateway** <*hostname-or-IPaddr*> **[host] [interface** <*hostname-or-IPaddr*>**] [intf-list** <*IPaddr-list*>**] [preference** <*num*>**] [retain] [reject] [no-install] [blackhole] [gate-list** <*gateway list*>**]**
ip clear reverse-flows
ip disable dns-lookup|fast-icmp|forwarding| icmp-redirect {interface <*name*>**|all} |proxy-arp {interface** <*name*>**|all} |source-routing|icmp-message {echo-reply timestamp-reply time-exceeded destination-unreachables} |default-route-check**
ip dos disable port-attack-protection|directed-broadcast-protection
ip enable {directed-broadcast interface <*interface name*>**| all} | {reverse-flow all| policy| NAT| load-balance|normal} | {local-proxy-arp interface** <*interface name*> **| all} | limit-ip-option-pkts rate-threshold** <*num*>
ip helper-address interface <*interface-name*> <*helper-address*> **|all-interfaces** [<*udp-port#*>]
ip l3-hash module <*num*>**|all variant** <*num*>
ip set data-receive-size|control-receive-size <*num*>
ip set port <*port-list*> **forwarding-mode destination-based| host-flow-based**

Table 24. ip commands (Continued)

ip show connections [no-lookup]
ip show hash-variant <*num*>**|all**
ip show helper-address
ip show interfaces [<*interface-name*>**] [brief]**
ip show reverse-flows
ip show routes [show-protocol direct|default|ospf|ospf-ase|rip|bgp|static] [show-arps] [show-multicast] [show-summary] [verbose]
ip show stack-queues

ip add route

Purpose

Configure a static route.

Format

ip add route <*ipAddr-mask*>|**default gateway** <*hostname-or-IPaddr*> **[host] [interface** <*hostname-or-IPaddr*>**] [intf-list** <*IPaddr-list*>**] [preference** <*num*>**] [retain] [reject] [no-install] [blackhole] [gate-list** <*gateway list*>**]**

Mode

Configure

Description

The **ip add route** command creates a static route entry in the route table. The static route can be a default route, a route to a network, or a route to a specific host.

Parameters

<*ipAddr-mask*> IP address and netmask of the destination. You can specify the address and mask information using the traditional format (example: 10.1.2.3/255.255.0.0) or the CIDR format (example: 10.1.2.3/16). If you specify an address without mask information, the X-Pedition uses the natural mask for the address (/8 for Class A, /16 for Class B or /24 for Class C).

gateway <*hostname-or-IPaddr*>
IP address or hostname of the next hop router for this route.

host Specifies that this route is a route to a host.

interface The next hop interface associated with this route. When this option is specified, gateways are only considered valid when they are on one of these interfaces.

<*IPaddr-list*> The next hop interfaces associated with this route. When you specify this option, the only gateways considered valid are those identified on the list of interfaces.

preference The preference of this static route. The preference controls how this route competes with routes from other protocols. The parameter takes a value between 0-255. The default preference is 60.

retain If specified, this option prevents this static route from being removed from the forwarding table when the routing service (GateD) is gracefully shut down. Normally gated removes all routes except interface routes during a graceful

shutdown. The retain option can be used to insure that some routing is available even when GateD is not running.

reject If specified, install this route as a reject route. Instead of forwarding a packet like a normal route, reject routes cause packets to be dropped and unreachable messages to be sent to the originator of the packet.

no-install If specified, the route will not be installed in the forwarding table when it is active but will be eligible for exporting to other protocols.

blackhole This option is the same as the reject option with the exception that unreachable messages are not sent.

gate-list <*gateway list*>
Allows you to specify up to four gateways for a particular destination host or network.

Restrictions

None

Examples

To configure the router 10.4.1.1 as the default gateway for this X-Pedition:

```
xp(config)# ip add route default gateway 10.4.1.1
```

To configure the gateway 10.4.78.11 as the gateway for any packet destined for the subnet 10.4.14.0/24:

```
xp(config)# ip add route 10.4.14.0/24 gateway 10.4.78.11
```

To configure the gateway 10.4.78.11 as the gateway for any packet destined for the subnet 10.4.14.0/24:

```
xp(config)# ip add route 10.4.14.0/24 gateway 10.4.78.11
```

To configure the gateway 10.4.16.99 as the gateway to the host 10.4.15.2:

```
xp(config)# ip add route 10.4.15.2 host gateway 10.4.16.99
```

To configure a reject route entry for packets destined for the subnet 10.14.3.0/24:

```
xp(config)# ip add route 10.14.3.0/24 gateway 10.1.16.99 reject
```

ip clear reverse-flows

Purpose

Clears reverse flow statistics.

Format

ip clear reverse-flows

Mode

Enable

Description

The **ip clear reverse-flows** command deletes all reverse flow statistics. Reverse flows are IP traffic flows in the opposite direction, where source information becomes destination information and vice versa.

Parameters

None

Restrictions

None

Example

To clear the reverse flow statistics:

```
xp# ip clear reverse-flows
```

ip disable

Purpose

Disables IP options on the X-Pedition.

Format

ip disable dns-lookup|fast-icmp|forwarding|
icmp-redirect {interface <*name*>**|all} |proxy-arp {interface** <*name*>**|all} |source-routing**
|icmp-message {echo-reply timestamp-reply time-exceeded destination-unreachables}
|default-route-check

Mode

Configure

Description

The **ip disable** command allows you to disable features that are enabled by default on the X-Pedition.

Parameters

dns-lookup
Disables DNS name lookup for all commands. Sometimes a DNS server is too slow to respond and this can cause a command that displays information about many hosts to take a long time to finish. Disabling DNS lookup displays all host addresses as IP addresses instead of host names.

fast-icmp
Disables the fast ICMP feature on the X-Pedition. By default, the X-Pedition installs ICMP flows to be switched along the fast path in hardware if the ICMP flow is meant to be routed. ICMP echo requests are installed as control priority for packets destined for the X-Pedition. When this feature is disabled, all ICMP packets are handled via the slow path in software.

forwarding
Disables the router's ability to forward IP packets. No IP packets will be forwarded to any IP interface if this command is used.

icmp-redirect {interface <*interface name*>**|all}**
Disables ICMP redirection on the specified IP interface. If you specify the all keyword, ICMP redirection is disabled for all network interfaces.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

proxy-arp {interface <*name*>**|all}**
Disables the proxy ARP feature on the specified IP interface. By default, the **X-Pedition** acts as a proxy for ARP requests with destination addresses of hosts to which the **X-Pedition** can route traffic. Unless you actually require the use of proxy ARP, it is advisable to disable it on the **X-Pedition**. If you specify the all keyword, the proxy ARP feature is disabled for all network interfaces.

Note: If you remove an interface on which you used use the **ip disable proxy-arp interface** command to disable the proxy ARP feature, you will disable proxy-arp for the entire system. Any unrouted Layer-2 packets will be lost—they will not know to route to another port.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

source-routing
Causes the **X-Pedition** to drop packets that have the SOURCE_ROUTE option set in the IP header. By default, packets that have the SOURCE_ROUTE option set are forwarded using the next-hop address in the IP packet.

icmp-message {echo-reply timestamp-reply time-exceeded destination-unreachables}
Disables the ability to send out ICMP messages. ICMP messages are used to communicate errors in packet traffic to other **X-Pedition**s. You can disable the following ICMP response messages: **echo-reply**, **timestamp-reply**, **time-exceeded**, and **destination-unreachables**.

default-route-check
Allows a default route to be set through the management (en0) interface.

Restrictions

None

Examples

To disable ICMP redirection on the “int4” network interface:

```
xp(config)# ip disable icmp-redirect int4
```

To disable DNS name lookup for all commands:

```
xp(config)# ip disable icmp-redirect dns-lookup
```

To prevent the X-Pedition from acting as a proxy for ARP requests with destination addresses of hosts to which the X-Pedition can route traffic:

```
xp(config)# ip disable proxy-arp interface all
```

ip dos disable

Purpose

Disables denial of service (DOS) features on the X-Pedition.

Format

ip dos disable directed-broadcast-protection|port-attack-protection

Mode

Configure

Description

By default, the X-Pedition installs flows in the hardware so that packets sent as directed broadcasts are dropped in hardware if directed broadcast is not enabled on the interface where the packet is received. You can disable this behavior with the **ip dos disable directed-broadcast-protection** command.

Similarly, the X-Pedition installs flows to drop packets destined for the X-Pedition for which service is not provided by the X-Pedition. This prevents packets for unknown services from slowing the CPU. You can disable this behavior with the **ip dos disable port-attack-protection** command, causing these packets to be processed by the CPU.

Parameters

directed-broadcast-protection
Disables the directed-broadcast-protection feature of the X-Pedition. By default the X-Pedition drops packets sent as directed broadcasts if directed broadcast is not enabled on the interface where the packet is received. This command causes directed broadcast packets to be processed on the X-Pedition even if directed broadcast is not enabled on the interface receiving the packet.

port-attack-protection
Disables the port-attack-protection feature of the X-Pedition. By default, packets that are destined for the X-Pedition, but do not have a service defined for them on the X-Pedition, are dropped. This prevents packets for unknown services from slowing the X-Pedition's CPU. This command disables this behavior, allowing packets destined for the X-Pedition that do not have a service defined for them on the X-Pedition to be processed by the X-Pedition's CPU.

Restrictions

None

Examples

To cause directed broadcast packets to be processed on the X-Pedition, even if directed broadcast is not enabled on the interface receiving the packet:

```
xp(config)# ip dos disable directed-broadcast-protection
```

To allow packets destined for the X-Pedition, but do not have a service defined for them on the X-Pedition, to be processed by the X-Pedition's CPU:

```
xp(config)# ip dos disable port-attack-protection
```

ip enable

Purpose

Enables IP options on the X-Pedition.

Format

ip enable {directed-broadcast interface <*interface name*>**| all} | {reverse-flow all| policy| NAT| load-balance|normal} | {local-proxy-arp interface** <*interface name*> **| all} | limit-ip-option-pkts rate-threshold** <*num*>

Mode

Configure

Description

The **ip enable** command allows you to configure the router to forward directed broadcast packets received on an interface, to set up reverse flows, and to limit the number of IP packets containing the option field that the Control Module will process per second.

Directed broadcast packets are network or subnet broadcast packets which are sent to a router to be forwarded as broadcast packets. They can be misused to create Denial Of Service attacks. The X-Pedition protects against this possibility by ***not*** forwarding directed broadcasts, by default. To enable the forwarding of directed broadcasts, use the **ip enable directed-broadcast** command.

Reverse flows in this case are Layer-3 flows heading in the opposite direction to the corresponding IP flows. IP flows are defined by the source and destination IP addresses, source and destination TCP/UDP port, Type of Service and transport protocol.

When an IP packet includes the option field in its IP header, the packet routes to the Control Module for further processing—even if the packet does not match any hardware flow. This can significantly degrade Control Module performance when several of these packets arrive over a short time. The **rate-threshold** option allows you to limit the rate of IP-option packets— if the rate exceeds this limit, the X-Pedition drops the packets. This feature is disabled by default.

Parameters

directed-broadcast interface <*interface name*>**|all**

This is the name of the specified IP interface. If you specify the **all** keyword, directed broadcast forwarding is enabled for all network interfaces.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

reverse-flow all| policy| NAT| load-balance| normal
Enables the ability to set up reverse flows. Specify **all** to disable any type of reverse flow to be set up. Specify **policy** to disable setting up reverse flows for policy routed packets. Specify **NAT** to disable setting up reverse flows for NAT packets. Specify **load-balance** to disable setting up reverse flows for load balance packets. Specify **normal** to disable setting up reverse flows for normally routed packets.

local-proxy-arp interface <*interface name*> **| all**
The local-proxy-arp parameter allows you to configure the **X-Pedition** to respond to all ARP requests it processes with its own MAC address, regardless of whether it is the owner of the IP address being requested. Implement this option only after carefully considering the network implications.

limit-ip-option-pkts rate-threshold <*num*>
The number of packets per second (0-3000) to use as the rate limit for IP option packets (enter 0 to drop every packet). Each time the X-Pedition receives a packet without an ip-option field defined, the router establishes a flow—L3-Aging removes the flow if it sits unused for a specific amount of time. Packets with an IP-option field defined do not use a standard flow; rather, they route directly to the CM. The X-Pedition collects statistics every second on the rate of IP option packets flowing to the CM-if the packet rate exceeds the limit specified, all IP option packets received in the same one-second period will be dropped and the router will increment the drop-flow count. The X-Pedition clears all dropped flows and begins processing IP option packets (if their rate falls below the threshold limit) each time half of the L3 aging period lapses-by default this period is 15 seconds.

Restrictions

The **limit-ip-option-pkts rate-threshold** <*num*> option will not work if L3 aging is disabled.

Examples

To enable directed broadcast forwarding on the “int4” network interface:

```
xp(config)# ip enable directed-broadcast interface int4
```

To enable directed broadcast forwarding for all network interfaces:

```
xp(config)# ip enable directed-broadcast interface all
```

To enable reverse flows for policy routed packets:

```
xp(config)# ip enable reverse-flow policy
```

To set the limit on the rate of IP option packets to 100 packets/seconds:

```
xp(config)# ip enable limit-ip-option-pkts rate-threshold 100
```

ip helper-address

Purpose

Configure the router to forward specific UDP broadcast packets across interfaces.

Format

ip helper-address interface <*interface-name*> <*helper-address*>|**all-interfaces** [<*udp-port#*>]

Mode

Configure

Description

The **ip helper-address** command allows the user to forward specific UDP broadcast from one interface to another. Typically, broadcast packets from one interface are not forwarded (routed) to another interface. However, some applications use UDP broadcast to detect the availability of a service. Other services, for example BOOTP/DHCP require broadcast packets to be routed so that they can provide services to clients on another subnet. An IP helper can be configured on each interface to have UDP broadcast packets forwarded to a specific host for a specific service or forwarded to all other interfaces.

The **ip helper-address** command allows the user to specify a UDP port number for which UDP broadcast packets with that destination port number will be forwarded. By default, if no UDP port number is specified, the X-Pedition will forward UDP broadcast packets for the following services:

- BOOTP/DHCP (port 67 and 68)
- DNS (port 53)
- NetBIOS Name Server (port 137)
- NetBIOS Datagram Server (port 138)
- Time Service (port 37)

Parameters

<*interface-name*> Name of the IP interface where UDP broadcast is to be forwarded to the helper address.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

<*helper-address*>|**all-interfaces**
Address of the host where UDP broadcast packets should be forwarded. If **all-**

	interfaces is specified, UDP broadcast packets are forwarded to all interfaces except the interface on which the broadcast packet was received.
<*udp-port*>	Destination UDP port number of the broadcast packets to forward. If not specified, packets for the six default services will be forwarded to the helper address.

Restrictions

If you specify an interface name, the name must belong to an existing IP interface.

Examples

To forward UDP broadcast packets received on interface int1 to the host 10.1.4.5 for the six default UDP services:

```
xp(config)# ip helper-address interface int1 10.1.4.5
```

To forward UDP broadcast packets received on interface int2 to the host 10.2.48.8 for packets with the destination port 111 (port mapper):

```
xp(config)# ip helper-address interface int2 10.2.48.8 111
```

To forward UDP broadcast packets received on interface int3 to all other interfaces:

```
xp(config)# ip helper-address interface int3 all-interfaces
```

ip l3-hash

Purpose

Changes the hashing algorithm used for the L3 lookup table.

Format

ip l3-hash module <*num*>|**all variant** <*num*>

Mode

Configure

Description

The X-Pedition's L3 Lookup table is organized as a hash table. The hash function reduces the destination and source MAC addresses to 16-bit quantities each. The hashing algorithm generates a uniform distribution within the MAC address space. However, given a particular set of addresses, the distribution may cause addresses to clump together in the table. To minimize the risk of thrashing in the tables, three variations to the basic hashing algorithm are defined. Only one variation is in effect on a line card at any given time. You can use the ip **l3-hash** command to set which variation is in effect for a line card.

Swizzling shifts the hash value by a certain amount of bits, producing more random distribution across the L3 lookup table.

Auto-hashing periodically queries the L3 table for hash bucket overflow on a port. If there are more overflows than a certain threshold level, auto-hashing will automatically change the hash mode for that port. Eventually a 'best' hash mode for the particular traffic will be found, which will provide a more even distribution across the L3 lookup table.

To see the effect changing the hashing algorithm has on the hash bucket, use the **statistics show l3-stat** command in the X-Pedition's Diagnostic mode.

Parameters

module <*num*>|**all**
Is a slot number on the X-Pedition. Specify any number between 1 and 16. The hashing algorithm change affects all ports on the line card in the slot. The **all** option causes the hashing algorithm to change on all ports on all slits.

variant <*num*>
Causes a variation to the basic hashing algorithm to be made. Valid variant numbers are: 0-3, 4-7 (swizzled), and 8 (auto-hashed). If you specify 0, the default hashing algorithm is used.

Restrictions

None.

Example

To change the default hashing algorithm used for the L3 lookup table on all ports on slot 7:

```
xp(config)# ip l3-hash module 7 variant 1
```

ip set data-receive-size | control-receive-size

Purpose

Sets the size of the stack data and control receive queues.

Format

ip set data-receive-size|control-receive-size <*num*>

Mode

Configure

Description

The **ip set data-receive-size|control-receive-size** command allows you to tune the size of the data and control pipes that reside between the IP stack and internal drivers on the Control Module.

Parameters

data-receive-size <*num*>
Sets the size of the stack data receive queue. Specify a value from 256-1024 bytes. The default is 512 bytes.

control-receive-size <*num*>
Sets the size of the stack control receive queue. Specify a value from 256-1024 bytes. The default is 512 bytes.

Restrictions

None.

Example

To set the size of the stack data receive queue to 1024 bytes:

```
xp(config)# ip set data-receive-size 1024
```

ip set port forwarding-mode

Purpose

Causes the X-Pedition, when processing an IP packet, to extract only certain fields from a layer-4 flow, rather than the entire flow.

Format

ip set port <*port-list*> **forwarding-mode destination-based|host-flow-based**

Mode

Configure

Description

The X-Pedition's flow identifying logic normally extracts the complete application (layer-4) flow from an IP packet. The **ip set port forwarding-mode** command causes the X-Pedition to extract only certain flow-related fields from the packet's L3 header, rather than the full layer-4 flow. This allows ports to route packets based on destination address alone, or on destination and source address only. As a result, in environments that do not have any filtering or RSVP requirements, the flow table can be used much more efficiently.

Parameters

port <*port-list*>
Modifies the flow extraction behavior on the specified ports. All ports must have an IP interface configured for them.

destination-based
If the packet is a unicast packet, causes the *destination IP address*, *TOS* and *L4 protocol* fields to be the only fields extracted from the IP packet. These fields and the *port of entry* field are set into the flow block being constructed. All of the other fields are set to zero.

For L3 multicast packets, the *destination IP address*, *source IP address, TOS* and *L4 protocol* fields are the only fields extracted from the IP packet. These along with the *port of entry* are the only fields set in the flow block. The remaining fields are set to zero. The flow lookup then proceeds as normal.

host-flow-based

For both unicast and multicast packets, the *destination IP address*, *source IP address, TOS* and the *L4 protocol* are the only fields extracted from the IP packet. These along with the *port of entry* are set in the flow block. The remaining flow block fields are set to zero. The flow lookup then proceeds as normal.

Restrictions

None

Example

To cause the X-Pedition to extract only the *destination IP address*, *TOS*, and *L4 protocol* fields from a layer-4 flow when processing an IP packet on port et.1.1:

```
xp(config)# ip set port et.1.1 forwarding-mode destination-based
```

To cause the X-Pedition to extract only the *destination IP address*, *source IP address, TOS,* and *L4 protocol* type from a layer-4 flow when processing an IP packet on port et.1.1:

```
xp(config)# ip set port et.1.1 forwarding-mode host-flow-based
```

ip show connections

Purpose

Show all TCP/UDP connections and services.

Format

ip show connections [no-lookup]

Mode

Enable

Description

The **ip show connections** command displays all existing TCP and UDP connections to the X-Pedition as well as TCP/UDP services available on the X-Pedition.

Parameters

no-lookup By default, when displaying an IP address, this command attempts to do a reverse DNS lookup to look for the hostname associated with the IP address and display the hostname instead. If you do not want the reverse DNS lookup to occur, specify the **no-lookup** option.

Restrictions

None.

Example

The following example displays all established connections and services of the X-Pedition.

```
xp# ip show connections
Active Internet connections (including servers)
Proto Recv-Q Send-Q  Local Address          Foreign Address        (state)
tcp        0      0  *:gated-gii            *:*                    LISTEN
tcp        0      0  *:http                 *:*                    LISTEN
tcp        0      0  *:telnet               *:*                    LISTEN
udp        0      0  127.0.0.1:1025         127.0.0.1:162
udp        0      0  *:snmp                 *:*
udp        0      0  *:snmp-trap            *:*
udp        0      0  *:bootp-relay          *:*
udp        0      0  *:route                *:*
udp        0      0  *:*                    *:*
```

ip show hash-variant

Purpose

Display IP hash variant per module.

Format

ip show hash-variant <*num*>|**all**

Mode

Enable

Description

The **ip show hash-variant** command displays hash variant information. There are a total of 16 modules using the hash variant feature (1-16).

Enabling hash variant causes a variation to the basic hashing algorithm. This variation will prevent clustering of hash values and will provide a more even distribution across the L3 lookup table. Valid variant numbers are: 0-3, 4-7 (swizzled), and 8 (auto-hashed). The default hashing algorithm is 0.

Swizzling shifts the hash value by a certain amount of bits, causing a more random distribution across the L3 lookup table. Auto-hashing allows the X-Pedition to auto-select a hashing algorithm optimized for ‘best case’ L3 table distribution.

Parameters

<*num*>|**all**	Specifies the module. Specify any number between 1-16. Specify **all** to display hash variant information for all modules.

Restrictions

None.

Example

To display IP hash variant information on all 16 modules:

```
xp# ip show hash-variant all
IP Module                    Hash Variant
-----------------------------------------------------
Module 2                      variant-0
Module 3                      variant-0
Module 4                      variant-0
Module 5                      variant-1
Module 6                      variant-0
Module 7                      variant-0
Module 8                      variant-2
Module 9                      variant-0
Module 10                     variant-7
Module 11                     variant-0
Module 12                     variant-6
Module 13                     variant-0
Module 14                     variant-0
Module 15                     variant-0
```

ip show helper-address

Purpose

Display the configuration of IP helper addresses.

Format

ip show helper-address [<*interface-name*>**]**

Mode

Enable

Description

The **ip show helper-address** command displays the configuration of IP helper addresses configured on the system. One can specify the optional parameter, *interface-name*, to show only the IP helper addresses configured for that interface. If the command is executed without specifying an interface name then the IP helper address configuration of all interfaces are shown.

Parameters

<*interface-name*> Name of the IP interface to display any configured IP helper addresses.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

If you specify an interface name, the name must belong to an existing IP interface.

Example

The following example shows that interface int4 has one helper address configured while interface int3 has one helper address configured for the port mapper service (port 111).

```
xp# ip show helper-address
Interface      IP address     Helper Address
---------      ----------     --------------
int6           10.1.17.1      none
int5           10.1.16.1      none
int4           10.1.15.1      10.4.1.45
int1           10.1.12.1      none
int0           10.1.11.1      none
int3           10.1.14.1      10.5.78.122(111)
```

ip show interfaces

Purpose

Display the configuration of IP interfaces.

Format

ip show interfaces [<*interface-name*>**] [brief] | all**

Mode

Enable

Description

The **ip show interfaces** command displays the configuration of an IP interface. If you issue the command without specifying an interface name the configuration of all IP interfaces is displayed. This command displays the same information as the **interface show ip** command.

Parameters

<*interface-name*> Name of the IP interface; for example, xp4. If you do not specify an interface name, the X-Pedition displays all the IP interfaces.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

brief Shows a brief summary of the interface in tabular form.

Restrictions

If you specify an interface name, the name must belong to an existing IP interface.

Example

To display the configuration of the IP interface “int1”:

```
xp# ip show interfaces int1
int1: flags=9862<BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,LINK0,MULTICAST>
      VLAN: IP2
      Ports:
      inet 10.1.12.1/24 broadcast 10.1.12.255
```

ip show reverse-flows

Purpose

Display reverse flow statistics.

Format

ip show reverse-flows

Mode

Enable

Description

The **ip show reverse-flows** command displays the reverse flow statistics. Reverse flows are IP traffic flows in the opposite direction, where source information becomes destination information and vice versa. This command shows the number of reverse flow packets.

Parameters

None

Restrictions

None

Example

To display the reverse flow statistics:

```
xp# ip show reverse-flows
IP Reverse Flow Statistics :
Total reverse-flow packets                    : 0
Successful reverse-flow packets               : 0
Unsuccessful reverse-flow packets             : 0
Arphold packets                              : 0
Find Flow entry success packets               : 0
Sum of arp hold and flow entry success packets : 0
```

ip show routes

Purpose

Display the IP routing table.

Format

ip show routes [show-protocol direct|default|ospf|ospf-ase|rip|bgp|static] [show-arps] [show-multicast] [show-summary] [verbose]

Mode

Enable

Description

The **ip show routes** command displays the IP routing table. Different command options can be used to show different aspects of the routing table.

Parameters

show-protocol Shows only the IP routes that belong to one of these specified protocols:

- **direct** Shows all direct routes.
- **default** Shows all default routes.
- **ospf** Shows all OSPF (Open Shortest Path First) routes.
- **ospf-ase** Shows all OSPF (Open Shortest Path First) Autonomous System-External routes.
- **rip** Shows all RIP (Routing Information Protocol) routes.
- **bgp** Shows all BGP (Border Gateway Protocol) routes.
- **static** Shows all manually defined routes.
- **show-arps** By default, ARP entries are not shown. To show ARP entries (if any are present), specify the **show-arps** option.

show-multicast By default, routes to multicast destinations are not shown. To show routes to multicast destinations, specify the **show-multicast** option.

show-summary Shows a summary of all route entries.

verbose Show the routing table in verbose mode (the additional information is useful for debugging). A list of definitions for the verbose command follows.

U: Up	Interface is up.
G: Gateway	This is a route to another network through the gateway specified.
H: Host	This is a route to a host through the gateway specified.
R: Reject	The router will return a “host unreachable” message upon receipt of a packet destined for this network and drop the packet.
D: Dynamic	The router received an ICMP redirect message for this route and installed it.
M: Modified	The router has changed an existing route because of an ICMP redirect.
C: Cloning	A directly connected interface route that may have more specific routes generated from it (e.g., ARP entries). Cloned routes may not be on point-to-point interfaces.
S: Static	This is a manually configured route.
W: Cloned	A route was “cloned” from another route that was “C” or “c”.
c: Pr Cloning	A route learned from a routing protocol may be “cloned.” Cloned routes may not be on point-to-point interfaces.
B: Blackhole	The router will drop—quietly—any packets received on this interface destined for this network. A “host unreachable” message will—not—be sent.

Restrictions

None.

Example

The following example displays the contents of the routing table. It shows that some of the route entries are for locally connected interfaces ("directly connected"), while some of the other routes are learned from RIP.

```
xp# ip show routes
Destination          Gateway              Owner     Netif
-----------          -------              -----     -----
10.1.0.0/16          50.1.1.2             RIP       to-linux2
10.2.0.0/16          50.1.1.2             RIP       to-linux2
10.3.0.0/16          50.1.1.2             RIP       to-linux2
10.4.0.0/16          50.1.1.2             RIP       to-linux2
14.3.2.1             61.1.4.32            Static    int61
21.0.0.0/8           50.1.1.2             RIP       to-linux2
30.1.0.0/16          directly connected     -       to-goya
50.1.0.0/16          directly connected     -       to-linux2
61.1.0.0/16          directly connected     -       int61
62.1.0.0/16          50.1.1.2             RIP       to-linux2
68.1.0.0/16          directly connected     -       int68
69.1.0.0/16          50.1.1.2             RIP       to-linux2
127.0.0.0/8          127.0.0.1            Static    lo
127.0.0.1            127.0.0.1              -       lo
210.11.99.0/24         directly connected    -       int41
```

ip show stack-queues

Purpose

Display the IP stack queues.

Format

ip show stack-queues

Mode

Enable

Description

The **ip show stack-queues** command displays the IP stack queues drop and size information.

Parameters

None

Restrictions

None

Chapter 31

ip-redundancy (vrrp) Commands

The **ip-redundancy** commands let you display and configure the Virtual Router Redundancy Protocol (VRRP) on the X-Pedition. VRRP is defined in RFC 2338.

Notes:

- Do not use an IP address for load-balancing that is already configured for VRRP.
- Interfaces configured with PVCs do not support VRRP.
- The X-Pedition supports only 512 instances of VRRP. An instance is defined as one virtual router running on one interface. Running a single virtual router on four interfaces is considered four instances of VRRP, as is running four virtual routers on a single interface.

Command Summary

Table 25 lists the **ip-redundancy** commands. The sections following the table describe the command syntax.

Table 25. ip-redundancy commands

ip-redundancy associate vrrp <*vrid*> **interface** <*interface*> **id** <*vrid*> **address** <*ip address/mask*>
ip-redundancy clear vrrp-stats interface <*interface*> **id** <*vrid*>
ip-redundancy create vrrp <*vrid*> **interface** <*interface*>

Table 25. ip-redundancy commands (Continued)

ip-redundancy set vrrp <*vrid*> **interface** <*interface*> **[priority** <*num*>**]** **[adv-interval** <*num*>**] [preempt-mode enabled|disabled|owner-disabled]** **[auth-type none|text] [auth-key** <*key*>**] [warmup-period** <*num*>**] [icmp-response]**
ip-redundancy show vrrp interface <*interface*> **[id**<*vrid*>**] [verbose] [summary]**
ip-redundancy start vrrp <*vrid*> **interface** <*interface*>
ip-redundancy trace vrrp [events <*enabled* | *disabled*>**] [state-transitions** <*enabled* | *disabled*>**] [packet-errors** <*enabled* | *disabled*>**] [all** <*enabled* | *disabled*>**]** <*option*>

ip-redundancy associate

Purpose

Associates an IP address with a virtual router.

Format

ip-redundancy associate vrrp <*vrid*> **interface** <*interface*> **address** <*ipaddr/mask*>

Mode

Configure

Description

The **ip-redundancy associate** command adds an IP address to the list of IP addresses associated with a virtual router.

Parameters

<*vrid*> Is the identifier of a virtual router. Specify a number between 1-255

<*interface*> Is the name of the interface where the virtual router resides.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display VLAN and interface names up to 32 characters in length.

Note: Do not use an IP address for VRRP that is already configured for load-balancing.

<*ipaddr/mask*> Is the IP address and subnet mask to be associated with the virtual router.

Restrictions

- Interfaces configured with PVCs do not support VRRP.
- Do not use an IP address for VRRP that is already configured for load-balancing.

Example

To add IP address/mask 1.2.3.4/16 to the list of IP addresses associated with virtual router 1 on interface int1:

```
xp(config)# ip-redundancy associate vrrp 1 interface int1 address 1.2.3.4/16
```

ip-redundancy clear vrrp-stats

Purpose

Clears statistics gathered for VRRP.

Format

ip-redundancy clear vrrp-stats interface <*interface*> **[id** <*vrid*>**]**

Mode

Enable

Description

The **ip-redundancy clear vrrp-stats** command is used in conjunction with the **ip-redundancy show vrrp** command, which displays information about the virtual routers associated with an interface. When you specify the **verbose** option with the **ip-redundancy show vrrp** command, additional statistics are shown, including the number of times a Backup router became the Master, the number of VRRP advertisements received, and counts of VRRP packets that contain errors. When you run the **ip-redundancy clear vrrp-stats** command, these statistics are reset to zero.

Parameters

<*interface*> Causes VRRP statistics to be cleared for all virtual routers on the specified interface.

<*vrid*> Causes VRRP statistics to be cleared for the virtual router with the specified VRID. Enter a number between 1-255.

Restrictions

Interfaces configured with PVCs do not support VRRP.

Example

To clear statistics for virtual router 1 on interface int1:

```
xp# ip-redundancy clear vrrp-stats interface int1 id 1
```

ip-redundancy create

Purpose

Creates a virtual router.

Format

ip-redundancy create vrrp <*vrid*> **interface** <*interface*>

Mode

Configure

Description

The **ip-redundancy create** command creates a virtual router on a specified interface.

Parameters

<*vrid*> Is the identifier of the virtual router to create. Specify a number between 1-255.

<*interface*>Is the interface on which to create the virtual router.

Restrictions

- Interfaces configured with PVCs do not support VRRP.
- The X-Pedition supports only 512 instances of VRRP. An instance is defined as one virtual router running on one interface. Running a single virtual router on four interfaces is considered four instances of VRRP, as is running four virtual routers on a single interface.

Example

To create a virtual router with an identifier (VRID) of 1 on interface int1:

```
xp(config)# ip-redundancy create vrrp 1 interface int1
```

ip-redundancy set

Purpose

Sets parameters for a virtual router.

Format

ip-redundancy set vrrp <*vrid*> **interface** <*interface*> **[priority** <*num*>**]**
[adv-interval <*num*>**] [preempt-mode enabled|disabled|owner-disabled] [auth-type none|text]**
[auth-key <*key*>**] [warmup-period** <*num*>**] [icmp-response]**

Mode

Configure

Description

The **ip-redundancy set** command lets you specify parameters for a virtual router, including backup priority, advertisement interval, whether the router can preempt a Master router that has a lower priority, the type of authentication used, and warm up time.

Parameters

<*vrid*> Is the identifier of a virtual router. Specify a number between 1-255.

<*interface*> Is the name of the interface where the virtual router resides.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

priority <*num*> Specifies the backup priority to be used by this virtual router. This number must be between 1-254. The default is 100. The priority number applies only if the virtual router is not the IP address owner. The priority of the IP address owner is always 255 and cannot be changed.

adv-interval <*num*> Is the interval between VRRP advertisements in seconds. The default is 1 second.

preempt-mode Specifies whether a backup router can preempt a Master router with a lower priority. Use one of the following keywords:

- **enabled** Preempt mode is enabled. A backup router can pre-empt a lower-priority Master router.
- **disabled** Pre-empt mode is disabled. A backup router cannot pre-empt a lower-priority Master router.

owner-disabled
Pre-empt mode is disabled. A backup router cannot pre-empt a lower-priority Master router, even if it is the Owner.

auth-type Specifies the type of authentication used for VRRP exchanges between routers. Use one of the following keywords:

none VRRP exchanges are not authenticated (the default).

text VRRP exchanges are authenticated with a clear-text password.

auth-key <*key*> Is the clear-text password used to authenticate VRRP exchanges. If you specify the **text** keyword, you must also specify the **auth-key** parameter.

warmup-period <*num*> Specifies the amount of delay (in seconds) before this virtual router is initialized, following a system reboot. Specify any number between 0 and 180. This delay is used to prevent a virtual router from preempting an existing Master before having received all of the routing updates from neighboring routers. (Default delay is 30 seconds).

icmp-response Specifies whether the backup router will respond to ICMP echo requests (pings) to the virtual IP address when the backup router is in master state.

Restrictions

Interfaces configured with PVCs do not support VRRP.

Examples

To specify 200 as the priority used by virtual router 1 on interface int1:

```
xp(config)# ip-redundancy set vrrp 1 interface int1 priority 200
```

To set the advertisement interval to 3 seconds:

```
xp(config)# ip-redundancy set vrrp 1 interface int1 adv-interval 3
```

To prevent a Backup router from taking over as Master from a Master router that has a lower priority:

```
xp(config)# ip-redundancy set vrrp 1 interface int1 preempt-mode disabled
```

To authenticate VRRP exchanges on virtual router 1 on interface int1 with a password of ‘yellow’:

```
xp(config)# ip-redundancy set vrrp 1 interface int1 auth-type text auth-key yellow
```

When enterprise customers run an X-Pedition in a VRRP configuration, the customers may not know if a problem exists with the Backup router. As a result, the X-Pedition feature set includes the ability to ping the Backup router while the router is in a non-Master state. When a Backup VRRP router assumes mastership, RFC 2338 specifies that it must not answer to ICMP echo requests (pings) destined to the associated virtual address. In some network situations, however, you may want to permit the Backup router to respond with an ICMP Echo Response when it is in the Master state. Use the following command to enable ICMP Echo Response:

ip-redundancy set vrrp <*vrID*> **interface** <*interface*> **icmp-response**

ip-redundancy show

Purpose

Shows information about virtual routers.

Format

ip-redundancy show vrrp interface <*interface*> **[id** <*vrid*>**] [verbose] [summary]**

Mode

Enable

Description

The **ip-redundancy show vrrp** command displays configuration information about virtual routers on an interface. You can display information for one virtual router or for all the virtual routers on an interface. If you specify the verbose option, additional statistics are shown, including the number of times a Backup router became the Master, the number of VRRP advertisements received, and counts of VRRP packets that contain errors. These statistics are gathered from the time you start the virtual router, or from the time you last ran the **ip-redundancy clear vrrp-stats** command.

Parameters

<*interface*> Is the name of the interface where the virtual router resides. If you do not specify the <*vrid*> parameter, information about all virtual routers on the interface is displayed.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

<*vrid*> Is the identifier of a virtual router. Specify a number between 1-255.

verbose Causes VRRP statistics to be displayed for each virtual router

Restrictions

- Interfaces configured with PVCs do not support VRRP.
- The X-Pedition supports only 512 instances of VRRP. An instance is defined as one virtual router running on one interface. Running a single virtual router on four interfaces is considered four instances of VRRP, as is running four virtual routers on a single interface.

Examples

To display information about all virtual routers on interface int1:

```
xp# ip-redundancy show vrrp interface int1

VRRP Virtual Router 100 - Interface int1
-----------------------------------------
Uptime               0 days, 0 hours, 0 minutes, 17 seconds.
State               Backup
Priority             100 (default value)
Virtual MAC address    00005E:000164
Advertise Interval     1 sec(s) (default value)
Preempt Mode          Enabled (default value)
Authentication        None (default value)
Primary Address       10.8.0.2
Associated Addresses   10.8.0.1
                     100.0.0.1

VRRP Virtual Router 200 - Interface int1
-----------------------------------------
Uptime               0 days, 0 hours, 0 minutes, 17 seconds.
State               Master
Priority             255 (default value)
Virtual MAC address    00005E:0001C8
Advertise Interval     1 sec(s) (default value)
Preempt Mode          Enabled (default value)
Authentication        None (default value)
Primary Address       10.8.0.2
Associated Addresses   10.8.0.2
```

To display VRRP statistics for virtual router 100 on interface int1:

```
xp# ip-redundancy show vrrp 1 interface int1 verbose

VRRP Virtual Router 100 - Interface int1
-----------------------------------------
Uptime               0 days, 0 hours, 0 minutes, 17 seconds.
State                Backup
Priority             100 (default value)
Virtual MAC address    00005E:000164
Advertise Interval     1 sec(s) (default value)
Preempt Mode           Enabled (default value)
Authentication         None (default value)
Primary Address        10.8.0.2
Associated Addresses   10.8.0.1
                       100.0.0.1

Stats:
  Number of transitions to master state           2
  VRRP advertisements rcvd                      0
  VRRP packets sent with 0 priority               1
  VRRP packets rcvd with 0 priority               0
  VRRP packets rcvd with IP-address list mismatch  0
  VRRP packets rcvd with auth-type mismatch       0
  VRRP packets rcvd with checksum error           0
  VRRP packets rcvd with invalid version          0
  VRRP packets rcvd with invalid VR-Id            0
  VRRP packets rcvd with invalid adv-interval     0
  VRRP packets rcvd with invalid TTL              0
  VRRP packets rcvd with invalid 'type' field     0
  VRRP packets rcvd with invalid auth-type        0
  VRRP packets rcvd with invalid auth-key         0
```

ip-redundancy start vrrp

Purpose

Starts a virtual router.

Format

ip-redundancy start vrrp <*vrid*> **interface** <*interface*>

Mode

Configure

Description

The **ip-redundancy start vrrp** command starts a virtual router on the specified interface.

Parameters

<*vrid*> Is the identifier of a virtual router. Specify a number between 1-255.

<*interface*> Is the name of the interface where the virtual router resides.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

- Interfaces configured with PVCs do not support VRRP.
- The X-Pedition supports only 512 instances of VRRP. An instance is defined as one virtual router running on one interface. Running a single virtual router on four interfaces is considered four instances of VRRP, as is running four virtual routers on a single interface.

Example

To start virtual router 1 on interface int1:

```
xp# ip-redundancy start vrrp 1 interface int1
```

ip-redundancy trace

Purpose

Traces VRRP events.

Format

ip-redundancy trace vrrp [events <*enabled* | *disabled*>**] [state-transitions** <*enabled* | *disabled*>**] [packet-errors** <*enabled* | *disabled*>**] [all** <*enabled* | *disabled*>**]** <*option*>

Mode

Configure

Description

The **ip-redundancy trace vrrp** command displays messages when certain VRRP events take place on the X-Pedition. Use this command to display messages when a virtual router changes from one state to another (i.e., from Backup to Master), a VRRP packet error is detected, or when any VRRP event occurs.

Parameters

events	Displays a message when VRRP receives any type of event. This option is disabled by default.
state-transitions	Displays a message when a VRRP router changes from one state to another. This option is enabled by default.
packet-errors	Displays a message when a VRRP packet error is detected. This option is enabled by default.
all enabled|disabled	Enables or disables all VRRP tracing.

Restrictions

- Interfaces configured with PVCs do not support VRRP.
- The X-Pedition supports only 512 instances of VRRP. An instance is defined as one virtual router running on one interface. Running a single virtual router on four interfaces is considered four instances of VRRP, as is running four virtual routers on a single interface.

Chapter 32

ip-router Commands

The **ip-router** commands let you configure and monitor features and functions that work across the various routing protocols.

Command Summary

Table 26 lists the **ip-router** commands. The sections following the table describe the command syntax.

Table 26. ip-router commands

ip-router authentication add key-chain <*option-list*>
ip-router authentication create key-chain <*option-list*>
ip-router find route <*ip-addr*> **[ignore-state]**
ip-router global add <*option-list*>
ip-router global set <*option-list*>
ip-router global set trace-options <*option-list*>
ip-router global set trace-state on|off
ip-router global use provided_config
ip-router kernel trace <*option-list*> **detail|send|receive**
ip-router policy add filter <*option-list*>
ip-router policy add optional-attributes-list <*option-list*>
ip-router policy aggr-gen destination <*name*> <*option-list*>
ip-router policy create aggregate-export-source <*option-list*>

Table 26. ip-router commands (Continued)

ip-router policy aggr-gen destination <*number-or-string*> **[source** <*number-or-string*>**] [filter** <*number-or-string*>**| network** <*ipAddr/mask*>**] [exact| refines| between** <*low-high*>**] [preference** <*number*>**| restrict]**
ip-router policy create aggr-gen-source <*option-list*>
ip-router policy create aspath-export-source <*number-or-string*> <*option-list*>
ip-router policy create bgp-export-destination <*number-or-string*> <*option-list*>
ip-router policy create bgp-export-source <*number-or-string*> <*option-list*>
ip-router policy create bgp-import-source <*number-or-string*> **[autonomous-system** <*number*>**]| [aspath-regular-expression** <*string*>**| origin** <*value*>**] [optional-attribute-list** <*num-or-string*>**] [preference** <*num*>**| restrict] [unicast] [multicast] [sequence-number** <*number*>**]**
ip-router policy create direct-export-source <*option-list*>
ip-router policy create filter <*option-list*>
ip-router policy create optional-attributes-list <*option-list*>
ip-router policy create ospf-export-destination <*number-or-string*> <*option-list*>
ip-router policy create ospf-export-source <*number-or-string*> <*option-list*>
ip-router policy create ospf-import-source <*number-or-string*> **[tag** <*num*>**][preference** <*num*>**|restrict] [unicast] [multicast]**
ip-router policy create rip-export-destination <*number-or-string*> <*option-list*>
ip-router policy create rip-export-source <*number-or-string*> <*option-list*>
ip-router policy create rip-import-source <*number-or-string*> **[interface** <*name-or-IPaddr*>**|gateway** <*name-or-IPaddr*>**][preference** <*num*>**|restrict] [unicast] [multicast] [sequence-number** <*number*>**]**
ip-router policy create static-export-source <*option-list*>
ip-router policy create tag-export-source <*number-or-string*> <*option-list*>
ip-router policy export destination <*option-list*>
ip-router policy import source <*imp-src-id*> **[[filter** <*filter-id*>**| network** <*ipAddr/mask*>**] [exact|refines|between** <*low-high*>**]] [preference** <*number*>**|restrict] [unicast] [multicast]**
ip-router policy redistribute from-proto <*protocol*> <*option-list*> **to-proto rip|ospf|bgp**
ip-router policy summarize route <*ipAddr/mask*> **| default [from-network** <*ipAddr/mask*>**] [exact | refines | between** <*low-high*>**] [preference** <*number*>**|restrict] [unicast] [multicast] [source-proto** <*protocol*>**] [type aggregate| generation] [brief]**
ip-router show configuration-file active|permanent

Table 26. ip-router commands (Continued)

ip-router show rib [detail]
ip-router show route [<*ip-addr-mask*>**| default] [detail]**
ip-router show state [all] [memory] [timers] [to-file] [to-terminal] [task <*string*>|all|gii |icmp|inet|interface|krt |route]

ip-router authentication add key-chain

Purpose

Add a key to an existing key-chain.

Format

ip-router authentication add key-chain <*option-list*>

Mode

Configure

Parameters

<*option-list*>
Specifies the options you are adding. Specify one of the following:

key <*string*>
Adds a new key to an existing key-chain. The key can be up to 16 characters long.

type primary|secondary
Specifies whether the key is a primary key or a secondary key within the key chain.

Restrictions

None.

ip-router authentication create key-chain

Purpose

Create a key-chain and associate an identifier with it.

Format

ip-router authentication create key-chain <*option-list*>

Mode

Configure.

Parameters

<*option-list*>
Specifies the options you are adding. Specify one of the following:

key <*string*>
Specifies a key to be included in this key chain. The key can be up to 16 characters long.

type primary|secondary
Specifies whether the key is a primary key or a secondary key within the key chain.

id
Specifies an integer between 1 and 255. This option is only necessary for MD5 authentication method.

Restrictions

None.

ip-router find route

Purpose

Find the active route in the RIB which the packet will use.

Format

ip-router find route **<*ip-addr*>** [**ignore-state]**

Mode

Enable.

Parameters

<*ip-addr*>
Specifies the destination of the packet.

ignore-state
This optional parameter allows inactive routes to be considered in route determination.

Restrictions

None.

ip-router global add

Purpose

Add an interface or martian. Martians are invalid addresses that are rejected by the routing software.

Format

ip-router global add interface <*name-or-IPaddr*>

ip-router global add martian <*ipAddr/mask*>|**default [host] [allow]**

Mode

Configure

Parameters

interface <*name-or-IPaddr*>
Makes an interface known to the IP router.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

martian <*ipAddr/mask*>|**default [host] [allow]**
Adds a martian. Specify the following options:

<*ipAddr/mask*>	The IP address and netmask for the martian.
default	Adds default martian.
host	Specifies that this martian is a host address.
allow	Allows a subset of a range that was disallowed.

Restrictions

None.

ip-router global set

Purpose

Set various global parameters required by various protocols.

Format

ip-router global set <*option-list*>

Mode

Configure

Parameters

<*option-list*>
Specify one of the following:

autonomous-system <*num1*> **loops** <*num2*>
The autonomous system number. <*num1*> sets the as number for the router. It is only required if the router is going to run BGP. Specify a number from 1 – 65534. <*num2*> controls the number of times the as may appear in the as-path. Default is 1. It is only required if the router is going to run protocols that support as-path, such as BGP.

router-id <*hostname-or-IPaddr*>
The router ID for use by BGP and OSPF. The most preferred address is any address other than 127.0.0.1 on the loopback interface. If there are no secondary addresses on the loopback interface, then the default router ID is set to the address of the first interface which is in the up state that the X-Pedition encounters (except the interface en0, which is the Control Module's interface). The address of a non point-to-point interface is preferred over the local address of a point-to-point interface.

interface <*interface-name*>|**all preference** <*num*> **down-preference** <*num*> **passive autonomous-system** <*num*>

Specify the following:

<*interface-name*>|**all**
Specify an interface that was added using the *ip-router global add interface* command, or **all** for all interfaces.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

preference <*num*>
Sets the preference for routes to this interface when it is up and functioning. Specify a number from 0 – 255. Default value is 0.

down-preference <*num*>
Sets the preference for routes to this interface when it is down. Specify a number from 0 – 255. Default value is 255.

passive
Prevents changing of route preference to this interface if it is down.

autonomous-system <*num*>
The AS that will be used to create as-path associated with the route created from the definition of this interface.

Restrictions

None.

ip-router global set trace-options

Purpose

Set various trace options.

Format

ip-router global set trace-options <*option-list*>

Mode

Configure

Parameters

<*option-list*>

Specify which trace options you will set:

startup	Trace startup events.
parse	Trace lexical analyzer and parser of gated config files.
debug	Trace lexical analyzer and parser in detail.
adv	Trace allocation and freeing of policy blocks.
symbols	Trace symbols read from kernel at startup.
if-list	Trace the reading of the kernel interface list.
all	Turn on all tracing.
general	Turn on normal and route tracing.
state	Trace state machine transitions in protocols.
normal	Trace normal protocol occurrences—the X-Pedition always traces abnormal occurrences.
policy	Traces the application of policy to imported and exported routes.
task	Traces system interfaces and task processing associated with this protocol or peer.
timer	Traces timer usage by this protocol or peer.
route	Traces routing table changes for routes installed by this protocol or peer.
none	Specifies that all tracing should be turned off for this protocol or peer.

Restrictions

None.

ip-router global set trace-state

Purpose

Enable or disable tracing.

Format

ip-router global set trace-state on|off

Mode

Configure

Parameters

on|off Specifies whether you are enabling or disabling tracing. Specify **on** to enable tracing or specify **off** to disable tracing. The default is **off**.

Restrictions

None.

ip-router global use provided_config

Purpose

Causes the X-Pedition to use the configuration file stored in the Control Module's NVRAM.

Format

ip-router global use provided_config

Mode

Configure

Parameters

None.

Restrictions

This command requires that you first copy the GateD configuration into the Control Module's NVRAM. To do this, enter the following command in Enable mode:

```
xp# copy tftp-server to gated.conf
TFTP server [10.50.89.88]? 10.50.89.88
Source filename [tmp/gated.conf]?
########
%TFTP-I-XFERRATE, Received 5910 bytes in 0.1 seconds
```

ip-router kernel trace

Purpose

Provides trace capabilities between the Routing Information Base and the Forwarding Information Base.

Format

ip-router kernel trace <*option-list*> **detail|send|receive**

Mode

Configure

Parameters

<*option-list*>

Specifies the kernel trace options. Specify one or more of the following:

packets	Packets exchanged with the kernel.
routes	Routes exchanged with the kernel.
redirect	Redirect messages received from the kernel.
interface	Interface messages received from the kernel.
other	All other messages received from the kernel.
remnants	Routes read from the kernel when the X-Pedition routing process starts.
request	The X-Pedition routing process requests to Add/Delete/Change routes in the kernel forwarding table.
info	Informational messages received from the routing socket, such as TCP loss, routing lookup failure, and route resolution request.

Restrictions

None.

ip-router policy add filter

Purpose

Adds a route filter. Routes are specified by a set of filters that will match a certain set of routes by destination, or by destination and mask.

Format

ip-router policy add filter <*number-or-string*> **network** <*ipAddr/mask*> **[exact|refines|between** <*low-high*>**][host-net]**

Mode

Configure

Parameters

filter <*number-or-string*>
Specifies the identifier of the route filter.

network <*IP-address*>
Specifies networks that are to be filtered. Matching usually requires both an address and a mask, although the mask can be implied. If no additional options qualifying the networks to be filtered are specified, then any destination that falls in the range implied by this network-specification is matched, so the mask of the destination is ignored. If a natural network is specified, the network, any subnets, and any hosts will be matched. If you specify the **exact**, **refines**, or **between** parameters, the mask of the destination is also considered.

exact
Specifies that the mask of the routes to be filtered must match the supplied mask exactly. This is used to match a network, but not subnets or hosts of that network

refines
Specifies that the mask of the routes to be filtered must be more specific (i.e. longer) than the supplied mask. This is used to match subnets.

between <*low-high*>
Specifies that the mask of the destination must be as or more specific (i.e., as long as longer) than the lower limit (the first number parameter) and no more specific (i.e. as long as or shorter) than the upper limit (the second parameter).

host-net
This option qualifies that the specified network is a host. To match, the address must exactly match the specified and the network mask must be a host mask (i.e. all ones). This is equivalent to a network specification of host/255.255.255.255 along with the exact option.

Restrictions

None.

ip-router policy add optional-attributes-list

Purpose

Expands a previously created optional-attributes-list.

Format

ip-router policy add optional-attributes-list <*option-list*>

Mode

Configure

Parameters

<*option-list*>
Specifies the options. Specify one or more of the following:

optional-attributes-list <*number-or-string*>
Specifies the identifier for the optional attributes list you are expanding.

community-id <*number*>
Specifies a community identifier portion of a community split. This is combined with the autonomous system value entered to create a value for the community attribute.

autonomous-system <*number*>
Specifies the autonomous system portion of a community split. This would be combined with the community id value entered to create a value for the community attribute. Specify a number from 1 – 65534.

no-export
Specifies that all routes received with this attribute value ***will not*** be advertised outside a BGP confederation boundary.

well-known-community
Specifies one of the well-known communities.

no-advertise
Specifies that all routes received with this attribute value ***will not*** be advertised to other BGP peers.

no-export-subconfed
Specifies that all routes received with this attribute value ***will not*** be advertised to external BGP peers (this includes peers in other members autonomous systems inside a BGP confederation).

reserved-community <*number*>
Specifies one of the reserved communities which is not well-known. A reserved

community is one which is in one of the following ranges (0x0000000 - 0x0000FFFF) or (0xFFFF0000 - 0xFFFFFFFF).

Restrictions

None.

ip-router policy aggr-gen destination

Purpose

Creates an aggregate or generate route.

Format

ip-router policy aggr-gen destination <*number-or-string*> **[source** <*number-or-string*>**] [filter** <*number-or-string*>**| network** <*ipAddr/mask*>**] [exact| refines| between** <*low-high*>**] [preference** <*number*>**| restrict]**

Mode

Configure

Parameters

destination <*number-or-string*>
Is the identifier of the aggregate-destination that specifies the aggregate/summarized route.

source <*number-or-string*>
Is the identifier of the aggregate-source that contributes to an aggregate route.

filter <*number-or-string*>
Specifies the filter for an aggregate/generate.

network <*ipAddr/mask*>
This option specifies networks which are to be aggregated. Matching usually requires both an address and a mask, although the mask can be implied. If no additional options qualifying the networks to be aggregated are specified, then any destination that falls in the range implied by this network-specification is matched, so the mask of the destination is ignored. If a natural network is specified, the network, any subnets, and any hosts will be matched. If you specify the **exact**, **refines**, or **between** parameters, the mask of the destination is also considered.

exact
This option specifies that the mask of the routes to be aggregated must match the supplied mask exactly. This is used to match a network, but not subnets or hosts of that network

refines
This option specifies that the mask of the routes to be aggregated must be more specific (i.e. longer) than the supplied mask. This is used to match subnets.

between <*low-high*>
Specifies that the mask of the destination must be as or more specific (i.e., as long as longer) than the lower limit (the first number parameter) and no more specific (i.e. as long as or shorter) than the upper limit (the second parameter).

preference <*number*>
This option specifies the preference to be assigned to the resulting aggregate route.

restrict
Specifies that routes matching the filter are not to be imported.

unicast
This option specifies that the resulting aggregate should be installed into the unicast rib. If neither unicast nor multicast is specified, the aggregate will belong to the unicast rib only.

multicast
This option specifies that the resulting aggregate should be installed into the multicast rib. If neither unicast nor multicast is specified, the aggregate will belong to the unicast rib only.

Restrictions

None.

ip-router policy create aggregate-export-source

Purpose

Creates a source for exporting aggregate routes into other protocols.

Format

ip-router policy create aggregate-export-source
<*number-or-string*> **[metric** <*number*>**|restrict]**

Mode

Configure

Parameters

<*number-or-string*>	Specifies the identifier of the aggregate export source.
metric <*number*>	Specifies the metric to be associated with the exported routes.
restrict	Specifies that nothing is exported from the specified source.

Restrictions

None.

ip-router policy create aggr-gen-dest

Purpose

Creates an aggregate-generation destination. An aggregate-generation destination is one of the building blocks needed to create an aggregate/generate route.

Format

ip-router policy create aggr-gen-dest <*number-or-string*>
network <*ipAddr/mask*>|**default [type aggregate**|**generation] [preference** <*number*>**]**|**[brief]**

Mode

Configure

Parameters

<*number-or-string*>
Specifies the identifier of an aggregate-generation destination.

network <*ipAddr/mask*>|**default**
Specifies the aggregate or generated route.

type aggregate
Specifies that the destination is an aggregate.

type generation
Specifies that the destination is a generate.

preference <*num*>
Specifies the preference to be assigned to the resulting aggregate route. The default preference is 130.

brief
Used to specify that the AS path should be truncated to the longest common AS path. The default is to build an AS patch consisting of SETs and SEQUENCES of all contributing AS paths.

Restrictions

None.

ip-router policy create aggr-gen-source

Purpose

Creates a source for the routes contributing to a aggregate/generate route.

Format

ip-router policy create aggr-gen-source <*number-or-string*>
protocol all|static|direct|aggregate|rip|ospf|bgp [autonomous-system <*number*>**][aspath-regular-expression** <*string*>**][tag** <*number*>**][preference** <*number*>**|restrict]**

Mode

Configure

Parameters

<*number-or-string*>
Specifies the identifier of an aggregate-generation source.

protocol <*string*>
Specifies the protocol of the contributing aggregate source. Specify one of the following:

- **all**
- **static**
- **direct**
- **aggregate**
- **rip**
- **ospf**
- **bgp**

autonomous-system <*number*>
Restricts selection of routes to those learned from the specified autonomous system. This selection may also be carried out by using route filters to explicitly list the set of routes to be accepted. Specify a number from 1 – 65534.

aspath-regular-expression <*string*>
Restricts selection of routes to those specified by the aspath.

tag <*number*>
Restricts selection of routes to those identified by a tag.

preference <*number*>
Specifies the preference to assign to the contributing routes.

restrict
Indicates that these routes cannot contribute to the aggregate.

Restrictions

None.

ip-router policy create aspath-export-source

Purpose

Create an export source where routes to be exported are identified by the autonomous system path associated with them. This command applies only if you are using BGP.

Format

ip-router policy create aspath-export-source <*number-or-string*> <*option-list*>

Mode

Configure

Parameters

<*number-or-string*>
Specifies a name or number for the Autonomous System path export source.

<*option-list*>
Specifies the Autonomous System path source options you are setting. Specify one of the following:

protocol <*name*>
Specifies the protocol by which the routes to be exported were learned. Specify one of the following:

•all

•static

•direct

•aggregate

•rip

•ospf

•bgp

aspath-regular-expression <*string*>
Specifies an aspath regular expression which should be satisfied for the route to be exported.

origin <*string*>
Specifies whether the origin of the routes to be exported was an interior gateway protocol or an exterior gateway protocol. Specify one of the following:

–any

–igp

–egp

–incomplete

metric <*num*>
Specifies metric associated with the exported routes.

restrict
Specifies that nothing is exported from the specified source.

Note: You can specify **metric** or **restrict** even if you specified **protocol**, **aspath-regular-expression**, or **origin**.

Restrictions

None.

ip-router policy create bgp-export-destination

Purpose

Create an export destination for BGP routes.

Format

ip-router policy create bgp-export-destination
<*number-or-string*> <*option-list*>

Mode

Configure

Parameters

<*number-or-string*>
Creates a BGP export destination and associates an identifier (tag) with it.

<*option-list*>
Specifies the BGP export destination options you are setting. Specify the following:

autonomous-system <*num*>
Specifies the autonomous system of the peer-group to which we would be exporting. Specify a number from 1 – 65535.

optional-attribute-list <*num-or-string*>
Specifies the identifier of the optional-attribute-list which contains the optional attributes which are to be sent along with these exported routes. This option may be used to send the BGP community attribute. Any communities specified in the optional-attributes-list are sent in addition to any received with the route or those specified with the 'set peer-group' or 'set peer-host' commands.

metric <*num*>
Specifies the metric to be associated with the BGP exported routes.

restrict
Restricts the export of BGP routes to the specified destination.

sequence-number <*num*>
Specifies the relative position of this export-destination in a list of bgp export-destinations.

Restrictions

None.

ip-router policy create bgp-export-source

Purpose

Create a source for exporting bgp routes into other protocols.

Format

ip-router policy create bgp-export-source <*number-or-string*> <*option-list*>

Mode

Configure

Parameters

<*number-or-string*>
Creates a BGP export source and associates an identifier (tag) with it.

<*option-list*>
Specifies the BGP export source options you are setting. Specify the following:

autonomous-system <*num*>
Specifies the autonomous system of the peer-group from which we would be exporting. A route filter could alternatively be used to explicitly list a set of routes to be accepted. Specify a number from 1 – 65534.

metric <*num*>
Specifies the metric to be associated with the BGP exported routes.

restrict
Restricts the export of BGP routes from the specified source.

Restrictions

None.

ip-router policy create bgp-import-source

Purpose

Create a source for importing BGP routes.

Format

ip-router policy create bgp-import-source <*number-or-string*> **[autonomous-system** <*number*>**]**| **[aspath-regular-expression** <*string*>| **origin** <*value*>**] [optional-attribute-list** <*num-or-string*>**] [preference** <*num*>| **restrict] [unicast] [multicast] [sequence-number** <*number*>**]**

Mode

Configure

Parameters

<*number-or-string*>
Creates a BGP import source and associates an identifier (tag) with it.

<*option-list*>
Specifies the BGP import source options you are setting. Specify the following:

autonomous-system <*number*>
Specifies the autonomous system of the peer-group from which we would be exporting. A route filter could alternatively be used to explicitly list a set of routes to be accepted. Specify a number from 1 – 65534.

aspath-regular-expression <*string*>
Specifies the as path regular expression that must be satisfied for the route to be exported. A route filter could alternatively be used to explicitly list a set of routes to be announced.

origin <*value*>
Specifies the origin attribute. Specify one of the following:

any	Specifies that the origin attribute can be any one of **igp**, **egp** and **incomplete**.
igp	Specifies that the origin attribute of the imported routes is IGP.
egp	Specifies that the origin attribute of the imported routes is EGP.
incomplete	Specifies that the origin attribute of the imported routes is incomplete.

optional-attribute-list <*num-or-string*>
Specifies the identifier of the optional-attribute-list. This option allows the specification of import policy based on the path attributes found in the BGP update. If multiple communities are specified in the aspath-opt option, only updates carrying all of the

specified communities will be matched. If none is specified, only updates lacking the community attribute will be matched.

preference <*num*>
Specifies the preference to be associated with the BGP imported routes.

restrict
Specifies that nothing is exported from the specified source.

unicast
This option specifies that the imported routes should be installed into the unicast rib. If neither unicast nor multicast is specified, the routes will belong to the unicast rib only.

multicast
This option specifies that the imported routes should be installed into the multicast rib. If neither unicast nor multicast is specified, the routes will belong to the unicast rib only.

sequence-number <*number*>
Indicates the position this bgp import source will have in a list of BGP import sources. Enter a value between 1 and 128, inclusive.

Restrictions

None.

ip-router policy create direct-export-source

Purpose

Creates an export source for interface routes.

Format

ip-router policy create direct-export-source <*number-or-string*> **[interface** <*name-or-IPaddr*>**][metric** <*num*>|**restrict]**

Mode

Configure

Parameters

<*number-or-string*>
Creates a source for exporting **interface** (**direct**) routes and associates an identifier with it.

interface <*name-or-IPaddr*>
This option qualifies that the direct routes should be associated with the specific interface.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

metric <*num*>
Specifies the metric to be associated with the exported routes.

restrict
Restricts the export of routes from the specified source.

Restrictions

None.

ip-router policy create filter

Purpose

Creates a route filter. Routes are filtered by specifying a set of filters that will match a certain set of routes by destination, or by destination and mask.

Format

ip-router policy create filter <*number-or-string*> **network** <*ipAddr/mask*> **[exact|refines|between** <*low-high*>**][host-net]**

Mode

Configure

Parameters

filter <*number-or-string*>
Specifies the identifier of the route filter.

network <*IP-address*>
This option specifies networks which are to be filtered. Matching usually requires both an address and a mask, although the mask can be implied. If no additional options qualifying the networks to be filtered are specified, then any destination that falls in the range implied by this network-specification is matched, so the mask of the destination is ignored. If a natural network is specified, the network, any subnets, and any hosts will be matched. If you specify the **exact**, **refines**, or **between** parameters, the mask of the destination is also considered.

exact
This option specifies that the mask of the routes to be filtered must match the supplied mask exactly. This is used to match a network, but not subnets or hosts of that network

refines
This option specifies that the mask of the routes to be filtered must be more specific (i.e. longer) than the supplied mask. This is used to match subnets.

between <*low-high*>
Specifies that the mask of the destination must be as or more specific (i.e., as long as longer) than the lower limit (the first number parameter) and no more specific (i.e. as long as or shorter) than the upper limit (the second parameter).

host-net
This option qualifies that the specified network is a host. To match, the address must exactly match the specified and the network mask must be a host mask (i.e. all ones). This is equivalent to a network specification of host/255.255.255.255 along with the exact option.

Restrictions

None.

ip-router policy create optional-attributes-list

Purpose

Creates an optional-attributes-list for BGP.

Format

ip-router policy create optional-attributes-list <*option-list*>

Mode

Configure

Parameters

<*option-list*>
Specifies the options you are setting. Specify the following:

<*number-or-string*>
Specifies the identifier for the attributes list.

community-id <*number*>
Specifies a community identifier portion of a community split. This is combined with the autonomous system value entered to create a value for the community attribute.

autonomous-system <*number*>
Specifies the autonomous system portion of a community split. This would be combined with the community id value entered to create a value for the community attribute. Specify a number from 1 – 65534.

no-export
Specifies that all routes received with this attribute value ***will not*** be advertised outside a BGP confederation boundary.

well-known-community
Specifies one of the well-known communities.

no-advertise
Specifies that all routes received with this attribute value ***will not*** be advertised to other BGP peers.

no-export-subconfed
Specifies that all routes received with this attribute value ***will not*** be advertised to external BGP peers (this includes peers in other members autonomous systems inside a BGP confederation).

reserved-community <*number*>
Specifies one of the reserved communities which is not well-known. A reserved

community is one which is in one of the following ranges (0x0000000 - 0x0000FFFF) or (0xFFFF0000 - 0xFFFFFFFF).

Restrictions

None.

ip-router policy create ospf-export-destination

Purpose

Create a destination for exporting routes into OSPF.

Format

ip-router policy create ospf-export-destination
<*number-or-string*> **[tag** <*num*>**][type 1|2][metric** <*num*>**|restrict]**

Mode

Configure

Parameters

<*number-or-string*>
Creates an OSPF export destination and associates an identifier with it.

tag <*num*>
Tag to be associated with exported OSPF routes.

type 1|2
Specifies that OSPF routes to be exported are type 1 or type 2 ASE routes. Specify 1 or 2.

metric <*num*>
Specifies the metric to be associated with the exported routes.

restrict
Restricts the export of the specified routes.

Restrictions

It is not possible to create OSPF intra- or inter-area routes by exporting routes from the routing table into OSPF. You can only export from the routing table into OSPF ASE routes.

ip-router policy create ospf-export-source

Purpose

Create a source for exporting OSPF routes into other protocols.

Format

ip-router policy create ospf-export-source
<*number-or-string*> **[type ospf|ospf-ase][metric** <*num*>**|restrict]**

Mode

Configure

Parameters

<*number-or-string*>
Creates an OSPF export source and associates an identifier with it.

type ospf
Exported routes are OSPF routes.

type ospf-ase
Exported routes are OSPF ASE routes.

metric <*num*>
Specifies the metric to be associated with the exported routes.

restrict
Specifies that nothing is to be exported from this source.

Restrictions

None.

ip-router policy create ospf-import-source

Purpose

Create a source for importing OSPF routes.

Format

ip-router policy create ospf-import-source <*number-or-string*> **[tag** <*num*>**][preference** <*num*>**|restrict] [unicast] [multicast]**

Mode

Configure

Parameters

<*number-or-string*>
Creates an OSPF import source and associates an identifier with it.

tag <*num*>
Tag to be associated with the imported routes.

preference <*num*>
Preference associated with the imported OSPF routes.

restrict
Specifies that matching **ospf-ase** routes are not imported.

unicast
This option specifies that the imported routes should be installed into the unicast rib. If neither unicast nor multicast is specified, the routes will belong to the unicast rib only.

multicast
This option specifies that the imported routes should be installed into the multicast rib. If neither unicast nor multicast is specified, the routes will belong to the unicast rib only.

Restrictions

None.

ip-router policy create rip-export-destination

Purpose

Create a destination for exporting routes into RIP.

Format

ip-router policy create rip-export-destination <*number-or-string*>
[interface <*name-or-IPaddr*>**|gateway** <*name-or-IPaddr*>**] [metric** <*num*>**|restrict]**

Mode

Configure

Parameters

<*number-or-string*>
Specifies an identifier for the RIP export destination:

interface <*name-or-IPaddr*>**|all**
Specifies router interfaces over which to export routes. Specify **all** to export routes to all interfaces.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

gateway <*name-or-IPaddr*>
Specifies the gateway that will receive the exported routes.

metric <*num*>
Specifies the metric to be associated with the exported routes. Specify a number from 1 – 16.

restrict
Restricts the export of routes to the specified destination.

Restrictions

When you use this command in conjunction with the *gateway* option in **ip-router policy export**, you must use **rip add source-gateways** for each address indicated in the gateway option.

ip-router policy create rip-export-source

Purpose

Create a source for exporting RIP routes into other protocols

Format

ip-router policy create rip-export-source
<*number-or-string*> **[interface** <*name-or-IPaddr*>|**gateway** <*name-or-IPaddr*>**][metric** <*num*>|**restrict]**

Mode

Configure

Parameters

<*number-or-string*>
Specifies an identifier for the RIP export source:

interface <*name-or-IPaddr*>
Indicates that only routes learned over specified interfaces are exported.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

gateway <*name-or-IPaddr*>
Indicates that only routes learned over specified gateways are exported.

metric <*num*>
Specifies the metric to be associated with the exported routes.

restrict
Indicates that nothing is exported from the specified source.

Restrictions

None.

ip-router policy create rip-import-source

Purpose

Create a source for importing RIP routes.

Format

ip-router policy create rip-import-source <*number-or-string*>
[interface <*name-or-IPaddr*>**|gateway** <*name-or-IPaddr*>**][preference** <*num*>**|restrict]**
[unicast] [multicast] [sequence-number <*number*>**]**

Mode

Configure

Parameters

<*number-or-string*>
Specifies an identifier for the RIP import source:

interface <*name-or-IPaddr*>
Indicates that only routes learned over specified interfaces are imported.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

gateway <*name-or-IPaddr*>
Indicates that only routes learned over specified gateways are imported.

preference <*num*>
Specifies the preference to be associated with the imported routes.

restrict
Indicates that nothing is imported from the specified source.

unicast
This option specifies that the imported routes should be installed into the unicast rib. If neither unicast nor multicast is specified, the routes will belong to the unicast rib only.

multicast
This option specifies that the imported routes should be installed into the multicast rib. If neither unicast nor multicast is specified, the routes will belong to the unicast rib only.

sequence-number <*number*>
This value indicates the position this rip_import_source would have in the list of rip import sources configured. Enter a value between 1 and 128, inclusive.

Restrictions

None.

ip-router policy create static-export-source

Purpose

Creates a source for exporting static routes into other protocols.

Format

ip-router policy create static-export-source <*number-or-string*>
[interface <*name-or-IPaddr*>**][metric** <*num*>**|restrict]**

Mode

Configure

Parameters

<*number-or-string*>
Creates a source for exporting **static** routes and associates an identifier with it.

interface
This option qualifies that the **static** routes should be associated with the specific interface.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

metric <*num*>
Specifies the metric to be associated with the exported routes.

restrict
Restricts the export of routes from the specified source.

Restrictions

None.

ip-router policy create tag-export-source

Purpose

Create an export source where routes to be exported are identified by the tag associated with them.

Format

ip-router policy create tag-export-source <*number-or-string*>
protocol all|static|direct|aggregate|rip|ospf|bgp
[tag <*number*>**][metric** <*number*>**|restrict]**

Mode

Configure

Parameters

<*number-or-string*>
Specifies the identifier of an tag-export source.

protocol <*string*>
Specifies the protocol of the contributing source. Specify one of the following:

- **all**
- **static**
- **direct**
- **aggregate**
- **rip**
- **ospf**
- **bgp**

tag <*number*>
Restricts selection of routes to those identified by a tag.

metric <*number*>
Specifies the metric to assign to the exported routes.

restrict
Indicates that the matching routes are not exported.

Restrictions

None.

ip-router policy export destination

Purpose

Creates an export policy from the various building blocks.

Format

ip-router policy export destination <*exp-dest-id*>
[**source** <*exp-src-id*> [**filter** <*filter-id*>|[**network** <*ipAddr/mask*> [**exact**|**refines**|**between** <*low-high*>] [**metric** <*number*>|**restrict**]]]]

Mode

Configure

Parameters

<*exp-dest-id*>
Is the identifier of the export-destination which determines where the routes are to be exported. If no routes to a particular destination are to be exported, then no additional parameters are required.

<*exp-src-id*>
If specified, is the identifier of the export-source which determines the source of the exported routes. If a export-policy for a given export-destination has more than one export-source, then the *ip-router policy export destination* <*exp-dest-id*> command should be repeated for each <*exp-src-id*>.

<*filter-id*>
If specified, is the identifier of the route-filter associated with this export-policy. If there is more than one route-filter for any export-destination and export-source combination, then the *ip-router policy export destination* <*exp-dest-id*> *source* <*exp-src-id*> command should be repeated for each <*filter-id*>.

network <*ipAddr/mask*>
Specifies networks which are to be exported. Matching usually requires both an address and a mask, although the mask can be implied. If no additional options qualifying the networks to be exported are specified, then any destination that falls in the range implied by this network-specification is matched, so the mask of the destination is ignored. If a natural network is specified, the network, any subnets, and any hosts will be matched. If you specify the **exact**, **refines**, or **between** parameters, the mask of the destination is also considered.

exact
This option specifies that the mask of the routes to be exported must match the supplied mask exactly. This is used to match a network, but not subnets or hosts of that network

refines
This option specifies that the mask of the routes to be exported must be more specific (i.e. longer) than the supplied mask. This is used to match subnets.

between <*low-high*>
Specifies that the mask of the destination must be as or more specific (i.e., as long as longer) than the lower limit (the first number parameter) and no more specific (i.e. as long as or shorter) than the upper limit (the second parameter).

restrict
Specifies that routes matching the filter are not to be exported.

metric <*number*>
Specifies the metric to be associated with the routes that match the specified filter.

Restrictions

None.

ip-router policy import source

Purpose

Creates an import policy.

Format

ip-router policy import source <*imp-src-id*> **[[filter** <*filter-id*>| **network** <*ipAddr/mask*>**] [exact|refines|between** <*low-high*>**]] [preference** <*number*>**|restrict] [unicast] [multicast]**

Mode

Configure

Parameters

<*imp-src-id*>
Is the identifier of the import-source that determines the source of the imported routes. If no routes from a particular source are to be imported, then no additional parameters are required.

<*filter-id*>
If specified, is the identifier of the route-filter associated with this import-policy. If there is more than one route-filter for any import-source, then the *ip-router policy import source* <*imp-src-id*> command should be repeated for each <*filter-id*>.

network <*ipAddr/mask*>
Specifies networks which are to be imported. Matching usually requires both an address and a mask, although the mask can be implied. If no additional options qualifying the networks to be imported are specified, then any destination that falls in the range implied by this network-specification is matched, so the mask of the destination is ignored. If a natural network is specified, the network, any subnets, and any hosts will be matched. If you specify the **exact**, **refines**, or **between** parameters, the mask of the destination is also considered.

exact
This option specifies that the mask of the routes to be imported must match the supplied mask exactly. This is used to match a network, but not subnets or hosts of that network

refines
This option specifies that the mask of the routes to be imported must be more specific (i.e. longer) than the supplied mask. This is used to match subnets.

between <*low-high*>
Specifies that the mask of the destination must be as or more specific (i.e., as long as longer) than the lower limit (the first number parameter) and no more specific (i.e., as long as or shorter) than the upper limit (the second parameter).

preference <*number*>
Specifies the preference with which the imported routes that match the specified filter should be installed.

restrict
Specifies that routes matching the filter are not to be imported.

unicast
This option specifies that the imported routes should be installed into the unicast rib. If neither unicast nor multicast is specified, the routes will belong to the unicast rib only.

multicast
This option specifies that the imported routes should be installed into the multicast rib. If neither unicast nor multicast is specified, the routes will belong to the unicast rib only.

Restrictions

None.

ip-router policy redistribute

Purpose

Creates a simple route redistribution policy

Format

ip-router policy redistribute from-proto <*protocol*> **to-proto** <*protocol*> **[network** <*ipAddr/mask*> **[exact|refines|between** <*low-high*>**]] [metric** <*number*>**|restrict] [source-as** <*number*>**] [target-as** <*number*>**] [tag] [ase-type]**

Mode

Configure

Parameters

from-proto <*protocol*>
Specifies the protocol of the source routes. The values for the from-proto parameter are **rip**, **ospf**, **bgp**, **direct**, **static**, **aggregate**, or **ospf-ase**.

to-proto <*protocol*>
Specifies the destination protocol where the routes are to be exported. The values for the to-proto parameter are **rip**, **ospf**, or **bgp**.

network <*ipAddr/mask*>
Provides a means to define a filter for the routes to be distributed. The network parameter defines a filter that is made up of an IP address and a mask. Routes that match the filter are considered as eligible for redistribution.

Matching usually requires both an address and a mask, although the mask can be implied. If no additional options qualifying the networks to be redistributed are specified, then any destination that falls in the range implied by this network-specification is matched, so the mask of the destination is ignored. If a natural network is specified, the network, any subnets, and any hosts will be matched. If you specify the **exact**, **refines**, or **between** parameters, the mask of the destination is also considered.

exact
This option specifies that the mask of the routes to be redistributed must match the supplied mask exactly. This is used to match a network, but not subnets or hosts of that network.

refines
This option specifies that the mask of the routes to be redistributed must be more specific (i.e. longer) than the supplied mask. This is used to match subnets.

between <*low-high*>
Specifies that the mask of the destination must be as or more specific (i.e., as long as longer)

than the lower limit (the first number parameter) and no more specific (i.e. as long as or shorter) than the upper limit (the second parameter).

restrict
Specifies that routes matching the filter are not to be redistributed.

metric
Indicates the metric to be associated with the redistributed routes.

tag
Tag to be associated with the exported OSPF routes.

ase-type
Routes exported from the GateD routing table into OSPF default to type **1** ASEs. This default may be explicitly overridden here. Thus, this option should be used to specify if the routes are to be exported as OSPF Type 1 or Type 2 ASE routes.

Note: Each protocol (RIP, OSPF, and BGP) has a configurable parameter that specifies the default-metric associated with routes exported to that protocol. If a metric is not explicitly specified with the redistribute command, then it is inherited from the default-metric associated with the protocol to which the routes are being exported.

Restrictions

None.

ip-router policy summarize route

Purpose

This command creates a simple aggregate or generation.

Format

ip-router policy summarize route <*ipAddr/mask*> **| default [from-network** <*ipAddr/mask*>**] [exact | refines | between** <*low-high*>**] [preference** <*number*>**|restrict] [unicast] [multicast] [source-proto** <*protocol*>**] [type aggregate| generation] [brief]**

Mode

Configure

Parameters

route <*ipAddr/mask*>**| default**
The summarized network. Specify default for default networks.

from-network
Specifies the network to be summarized. If no additional options that qualify the networks to be filtered are specified, then any destination that falls in the range implied by this network specification is matched; the mask of the destination is ignored. If a natural network is specified, the network and any subnets and any hosts will be matched.

exact Specifies that the mask of the routes to be summarized must match the supplied mask exactly. This is used to match a network, but no subnets or hosts of that network.

refines Specifies that the mask of the routes to be summarized must be more specific (i.e., longer) than the supplied mask. This is used to match subnets and/or hosts of a network, but not the network.

between <*low-high*>
Specifies that the mask of the routes to be summarized must be as or more specific (i.e., as long as or longer) than the lower limit (the first number value) and no more specific (i.e., as long as or shorter) than the upper limit (the second number value).

preference <*number*>
If specified, is the metric to be associated with the routes that match the filter.

restrict If specified, routes that match the filter are not to be summarized.

unicast
This option specifies that the resulting aggregate should be installed into the unicast rib. If neither unicast nor multicast is specified, the aggregate will belong to the unicast rib only.

multicast
This option specifies that the resulting aggregate should be installed into the multicast rib. If neither unicast nor multicast is specified, the aggregate will belong to the unicast rib only.

source-proto <*protocol*>
Specifies the protocol of the source routes. Specify one of the following:

all	All protocols.
bgp	BGP routes.
direct	Direct routes.
isis-level-1	IS-IS level 1 routes.
isis-level-2	IS-IS level 2 routes.
ospf	OSPF routes.
rip	RIP routes.
static	Static routes.

type aggregate| generation
Specifies whether the object being created is an aggregate or a generation.

brief Specifies that the AS path is to be truncated to the longest common AS path. The default is to build an AS path that consists of SETs and SEQUENCES of all contributing AS paths.

Restrictions

None.

ip-router show configuration file

Purpose

Display the active or startup configuration file in GateD format.

Format

ip-router show configuration-file active|permanent

Mode

Enable

Parameters

active	Shows the active GateD configuration file in RAM; this is the default.
permanent	Shows the permanent GateD configuration file in NVRAM, if available.

Restrictions

None.

ip-router show rib

Purpose

Display routing information base.

Format

ip-router show rib [detail]

Mode

Enable

Description

The **ip-router show rib** command shows the route-manager's routing information base (RIB). For any given network, the routing daemon could have multiple routes. The active route to any network is shown with a plus (+) sign next to it. The last active route is shown with a minus (-) next to it. If a route has been the last active route and is also the current active route, then it is shown with a asterisk (*) sign next to it. The legend is as follows:

- "+" Active Route
- "-" Last Active
- "*" Both

If the detail option is used, then additional information is displayed about these routes. The announcements bits for the active route are shown which shows the protocol into which this route is advertised.

Parameters

detail Allows you to view additional information about the routes in the RIB.

Restrictions

None.

Examples:

A sample output of the **ip-router show rib** command is shown below:

```
xp# ip-router show rib
Routing Tables:
Generate Default: no
Destinations: 63776    Routes: 63776
Holddown: 0    Delete: 53811    Hidden: 1
Codes: Network - Destination Network Address
      S - Status + = Best Route, - = Last Active, * = Both
      Src - Source of the route :
      Ag - Aggregate, B - BGP derived, C - Connected
      R - RIP derived, St - Static, O - OSPF derived
      OE - OSPF ASE derived, D - Default
      Next hop - Gateway for the route; Next hops in use: 4
      Netif - Next hop interface
      Prf1 - Preference of the route, Prf2 - Second Preference of the route
      Metrc1 - Metric1 of the route, Metrc2 - Metric2 of the route
      Age - Age of the route
Network/Mask       S Src Next hop            Netif Prf1 Metrc1 Metrc2       Age
------------       - --- --------            ----- ---- ------ ------       ---
3/8              *   B 134.141.178.33       mls0  170              70:34:28
4/8              *   B 134.141.178.33       mls0  170              70:34:28
4.17.106/24       *   B 134.141.178.33       mls0  170              70:34:28
4.17.115/24       *   B 134.141.178.33       mls0  170              70:34:28
4.24.148.128/25   *   B 134.141.178.33       mls0  170              70:34:28
6/8              *   B 134.141.178.33       mls0  170              70:34:28
6.80.137/24       *   B 134.141.178.33       mls0  170              70:34:28
9.2/16           *   B 134.141.178.33       mls0  170              70:34:28
9.20/17          *   B 134.141.178.33       mls0  170              70:34:28
10.50/16          *   C 10.50.90.1           en    0      0      0 113:31:09
10.60.90/24       *   C 10.60.90.1           mls2  0      0      0 113:31:09
12/8             *   B 134.141.178.33       mls0  170              70:34:28
12.1.248/24       *   B 134.141.178.33       mls0  170              70:34:28
12.2.19/25        *   B 134.141.178.33       mls0  170              12:47:48
12.2.76/24        *   B 134.141.178.33       mls0  170              31:03:36
12.2.97/24        *   B 134.141.178.33       mls0  170              1:41:30
12.2.109/24       *   B 134.141.178.33       mls0  170              87:55:47
12.2.169/24       *   B 134.141.178.33       mls0  170              113:31:01
12.3.63/24        *   B 134.141.178.33       mls0  170              70:34:28
12.4.5/24         *   B 134.141.178.33       mls0  170              70:34:28
12.4.126/24       *   B 134.141.178.33       mls0  170              70:34:28
12.4.164/24       *   B 134.141.178.33       mls0  170              70:34:28
12.4.175/24       *   B 134.141.178.33       mls0  170              95:47:57
12.4.196/22       *   B 134.141.178.33       mls0  170              70:34:28
12.5.48/21        *   B 134.141.178.33       mls0  170              70:34:28
12.5.164/24       *   B 134.141.178.33       mls0  170              113:31:01
12.5.252/23       *   B 134.141.178.33       mls0  170              70:34:28
12.6.42/23        *   B 134.141.178.33       mls0  170              70:34:28
12.6.97/24        *   B 134.141.178.33       mls0  170              70:34:28
```

ITo see a specific route, use the **ip-router show route** command.

ip-router show route

Purpose

Display specific route information from RIB.

Format

ip-router show route [<*ip-addr-mask*>**| default] [detail]**

Mode

Enable

Description

This command shows a specific route in the route-manager's routing information base (RIB). For any given network, the routing daemon could have multiple routes. The active route to any network is shown with a plus (+) sign next to it. The last active route is shown with a minus (-) next to it. If a route has been the last active route and is also the current active route, then it is shown with a asterisk (*) sign next to it. The legend is as follows:

- "+" Active Route
- "-" Last Active
- "*" Both

If the detail option is used, then additional information is displayed about this route. The announcements bits for the active route are shown which shows the protocol into which this route is advertised.

Parameters

<*ipAddr/mask*>**|default**
Allows you to specify a particular IP address mask for the RIB route in question, or refer to the default address mask.

detail
Allows you to view additional information about the routes in the RIB.

Restrictions

None.

Example

A sample output of the **ip-router show route detail** command is shown below.

```
xp# ip-router show route 10.12.1.0/255.255.255.252 detail
10.12.1          mask 255.255.255.252
entries 2    announce 1
TSI:
RIP 150.1.255.255mc  <> metric 1
RIP 222.1.1.255mc  <> metric 1
BGP_Sync_64805 dest 10.12.1/2 metric 0
BGP group type Routing AS 64805 no metrics
Instability Histories:
*Direct    Preference:   0
*NextHop: 10.12.1.2          Interface: 10.12.1.2(to-c4500)
State: <Int Active Retain>
Age: 5:12:10    Metric: 0    Metric2: 0    Tag: 0
Task: IF
Announcement bits(5):
2-KRT 4-RIP.0.0.0.0+520 5-RIP.0.0.0.0+520
6-BGP_Sync_64805
7-BGP_Group_64805
AS Path: IGP (Id 1)
OSPF    Preference: -10
*NextHop: 10.12.1.1          Interface: 10.12.1.2(to-c4500)
State: <NotInstall NoAdvise Int Hidden Gateway>
Local AS: 64805
Age: 1:20:05   Metric: 1    Metric2: -1    Tag: 0
Task: OSPF
AS Path: (64805) IGP (Id 9551)
Cost: 1    Area: 0.0.0.0    Type: Net    AdvRouter:
172.23.1.14
```

In this case there are two routes to network 10.12.1.0/255.255.255.252—one of them is a direct route and other route is learned through OSPF. The direct route has a better preference (lower preference is considered better preference), and is thus the active route. The direct route has been installed since 5 hours, 12 minutes and 10 seconds. This direct route is being announced to the Forwarding Information Base (FIB) which is indicated by KRT, over two RIP interfaces (which is indicated by 4-RIP.0.0.0.0+520, 5-RIP.0.0.0.0+520) and also to the BGP internal peer-group for autonomous system 64805.

To see all the routes in the RIB, use the **ip-router show rib** command.

Field Definitions

Field	Description
TSI	Indicates that the X-Pedition has stored some internal bookkeeping for the protocols listed here.
Instability Histories	The Flap histories associated with a particular route (i.e., how many times the route went up and down).
[protocol]	Indicates the route type (i.e., Direct OSPF, RIP, BGP, Static, or Aggregate).
Preference	The preference that the route has compared to other routes with the same destination.
NextHop	IP address of the next hop to use when routing a packet to this network.
Interface	The IP address and name where the NextHop is located.
State	The internal state of the route. For example, "ActiveU" indicates that this is the Unicast route currently used in the FIB or Forwarding Information Base. "Gateway" denotes a route that is more than one hop away.
Age	The amount of time lapsed since the route was last refreshed.
Metric	The routing metric. For example, the cost of the link for OSPF; or the hop count for RIP.
Metric2	Alternate metric for routing decision.
Tag	AS tag that is propagated through an OSPF.
Task	Task that contributed the information displayed with this output (e.g., IF, OSPF, RIP, BGP).
Announcement bits	Listed by task, which and how many protocols provide access to the route.
AS Path	IGP/EGP info and BGP AS path if it exists.
Local AS	The Local Autonomous System (AS) on which this route resides.
Cost	Link cost as calculated by OSPF.
Area	The area from which this route originated.
Type	The LSA type (e.g., 4,5,7).
AdvRouter	The router ID of the router that originated this LSA.

ip-router show state

Purpose

Displays the state of GateD.

Format

ip-router show state [all] [memory] [timers] [to-file] [to-terminal]
[task <*string*>|all|gii |icmp|inet|interface|krt |route]

Mode

Enable

Parameters

all	Shows all output.
memory	Shows memory allocations.
timers	Shows various GateD timers.
to-file	Saves the routing-process state in the gated.dmp file.
to-terminal	Displays the routing-process state on the console.
task	Shows task-specific information. The default is to show information for all tasks. You can specify a task using the **following options:**

<string>	Displays information for the task specified.
all	Shows information for all tasks.
gii	Shows GII information.
icmp	Shows information for the ICMP task.
inet	Shows information for the INET task.
interface	Shows information for the Interface task.
krt	Shows information for the KRT task.
route	Shows information for the route task.

Restrictions

None.

Chapter 33

ip-policy Commands

The **ip-policy** commands let you set up policies that cause the X-Pedition to forward packets to a specified IP address based on information in a packet's L3/L4 IP header fields.

Command Summary

Table 27 lists the **ip-policy** commands. The sections following the table describe the command syntax.

Table 27. ip-policy commands

ip-policy <*name*> **apply local**|**interface** <*name*>|**all**
ip-policy clear all|**policy-name** <*name*>|**all**
ip-policy <*name*> **deny acl** <*aclname*>|**everything-else** [**sequence** <*num*>]
ip-policy <*name*> **permit acl** <*aclname*>|**everything-else** [**sequence** <*num*>] **next-hop-list** <*ip-addr-list*> **action policy-first**|**policy-last**|**policy-only**
ip-policy <*name*> **set [pinger on] [load-policy round-robin**| **ip-hash sip**|**dip**|**both]**
ip-policy show [**all**] [**policy-name** <*name*>|**all**] [**interface** <*name*>|**all]**

ip-policy apply

Purpose

Applies an IP policy to an interface.

Format

ip-policy <*name*> **apply local|interface** <*InterfaceName*>|**all**

Mode

Configure

Description

Once you have defined an IP policy, you use the **ip-policy apply** command to apply the IP policy to an interface. Once the IP policy is applied to the interface, packets start being forwarded using the policy.

Parameters

<*name*> Is the name of a previously defined IP policy.

<*InterfaceName*> Is the name of the inbound interface to which you are applying the IP policy. The interface name must be less than 32 characters.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

local Causes packets generated by the X-Pedition to be forwarded according to the IP policy.

all Causes the IP policy to be applied to all IP interfaces.

Restrictions

IP policies can be applied to IP interfaces only.

Examples

To apply IP policy p1 to interface int4:

```
xp(config)# ip-policy p1 apply interface int4
```

To apply IP policy p2 to all IP packets generated on the X-Pedition:

```
xp(config)# ip-policy p2 apply local
```

ip-policy clear

Purpose

Clears IP policy statistics.

Format

ip-policy clear all|policy-name <*name*>**|all**

Mode

Enable

Description

The **ip-policy clear** command is used in conjunction with the **ip-policy show** command, which gathers statistics about IP policies. The **ip-policy clear** command lets you reset IP policy statistics to zero.

Parameters

<*name*> Is the name of an active IP policy.

all Causes statistics to be cleared for all IP policies.

Restrictions

None.

Examples

To clear statistics for IP policy p1:

```
xp# ip-policy clear policy-name p1
```

To clear statistics for all IP policies:

```
xp(config)# ip-policy clear all
```

ip-policy deny

Purpose

Specifies which packets cannot be subject to policy-based routing.

Format

ip-policy <*name*> **deny acl** <*aclname*>|**everything-else** [**sequence** <*num*>]

Mode

Configure

Description

The **ip-policy deny** command allows you to specifically prevent packets matching a profile from being forwarded with an IP policy. These packets are routed using dynamic routes instead.

Note: Since there is an implicit deny rule at the end of all IP policies, all packets that do not match any policy are forwarded using dynamic routes.

Parameters

<*name*>
Is the name of an IP policy.

acl <*aclname*>
Is the name of the ACL profile of the packets to be excluded from IP policy-based forwarding. Profiles are defined with the **acl** command. The ACL may contain either **permit** or **deny** keywords. The **ip-policy deny** command only looks at the following ACL rule parameter values: protocol, source IP address, destination IP address, source port, destination port, and TOS.

everything-else
Keyword that specifies an action to be performed for packets that do not match any of the previously-defined ACLs. Specifies that packets that are not ***specifically*** permitted to use policy-based routing are forwarded using dynamic routes.

sequence <*num*>
If an IP policy is composed of more than one **ip-policy** statement, specifies the order in which the statement is evaluated. Possible values are 1-65535. The **ip-policy** statement with the lowest sequence number is evaluated first.

Restrictions

ACLs for non -IP protocols cannot be used for IP policy routing.

Examples

To create a profile called “prof1” for telnet packets from 9.1.1.5 to 15.1.1.2:

```
xp(config)# acl prof1 permit ip 9.1.1.5 15.1.1.2 any any telnet 0
```

Note: See *acl permit|deny ip* on page 28 for more information on creating profiles for IP policy routing.

To create an IP policy called “p3” that prevents packets matching prof1 (that is, telnet packets from 9.1.1.5 to 15.1.1.2) from being forwarded using an IP policy:

```
xp(config)# ip-policy p3 deny acl prof1
```

To create a policy called “p4” that prevents all packets that have not been specifically permitted to use policy-based routing (using the **ip-policy permit** command) from being forwarded using an IP policy:

```
xp(config)# ip-policy p4 deny acl everything-else
```

ip-policy permit

Purpose

Specifies gateways and actions for IP policies

Format

ip-policy <*name*> **permit acl** <*aclname*>|**everything-else** [**sequence** <*num*>]
[next-hop-list <*ip-addr-list*>|**null**] [**action policy-first**|**policy-last**|**policy-only**]

Mode

Configure

Description

The **ip-policy permit** command allows you to specify the next-hop gateway where packets matching a given profile should be forwarded. You can specify up to four next-hop gateways for an IP policy. Packets matching a profile you defined with an **acl** command are forwarded to the next-hop gateway.

You can specify when to apply the IP policy route with respect to dynamic or statically configured routes. You can cause packets to use the IP policy route first, then the dynamic route if the next-hop gateway is unavailable; use the dynamic route first, then the IP policy route; or drop the packets if the next-hop gateway is unavailable.

Parameters

<*name*>
Is the name of an IP policy.

acl <*aclname*>
Is the name of the ACL profile of the packets to be forwarded using an IP policy. Profiles are created with the **acl** command. The ACL may contain either **permit** or **deny** keywords. The **ip-policy permit** command only looks at the following ACL rule parameter values: protocol, source IP address, destination IP address, source port, destination port, and TOS.

everything-else
Specifies that all packets not ***specified*** using policy-based routing (i.e., with the **ip-policy deny** command) are forwarded to the next-hop gateway.

sequence <*num*>
If an IP policy is composed of more than one **ip-policy** statement, specifies the order in which the statement is evaluated. Possible values are 1-65536. The **ip-policy** statement with the lowest sequence number is evaluated first.

next-hop-list <*ip-addr-list*>|**null**
Is the IP address of one or more next-hop gateways. Packets matching the profile specified in <*aclname*> are forwarded to one of the gateways specified here. You can specify up to four gateways for each profile. If you specify more than one gateway, enclose the list of IP addresses in quotes. You can define how the packet load is distributed among multiple gateways with the **ip-policy set load-policy** command.

To drop packets that match the profile, use the **null** keyword.

action policy-first|policy-last|policy-only
Specifies how IP policies are applied with respect to dynamic or statically configured routes. The following options are available:

policy-first Causes packets matching the specified profile to use the IP policy route first. If the next-hop gateway specified in the IP policy is not reachable, the dynamic route is used instead.

policy-last Causes packets matching the specified profile to be routed using dynamic routes first. If a dynamic route is not available, then all packets matching the profile are routed using the IP policy gateway.

policy-only Causes packets matching the specified profile to use the IP policy route. If the next-hop gateway specified in the IP policy is not reachable, then the packets are dropped.

Restrictions

ACLs for non IP protocols cannot be used for IP policy routing.

Examples

To create a profile called “prof1” for telnet packets from 9.1.1.5 to 15.1.1.2:

```
xp(config)# acl prof1 permit ip 9.1.1.5 15.1.1.2 any any telnet 0
```

Note: See *acl permit|deny ip* on page 28 for more information on creating profiles for IP policy routing.

To cause packets matching prof1 (that is, telnet packets from 9.1.1.5 to 15.1.1.2) to be forwarded to 10.10.10.10:

```
xp(config)# ip-policy p5 permit acl prof1 next-hop-list 10.10.10.10
```

To cause all packets that have not been specified using policy-based routing (using the **ip-policy deny** command) to be forwarded to 10.10.10.10:

```
xp(config)# ip-policy p5 permit acl everything-else next-hop-list 10.10.10.10
```

To cause packets matching prof1 to use dynamic routes if 10.10.10.10 is not available:

```
xp(config)# ip-policy p5 permit acl prof1 next-hop-list 10.10.10.10 action policy-first
```

To cause packets matching prof1 to be dropped if 10.10.10.10 is not available:

```
xp(config)# ip-policy p5 permit acl prof1 next-hop-list 10.10.10.10 action policy-only
```

ip-policy set

Purpose

Controls how packets are distributed among the next hop gateways in an IP policy and queries the availability of next-hop gateways.

Format

ip-policy <*name*> **set [pinger on] [load-policy round-robin| ip-hash sip|dip|both]**

Mode

Configure

Description

If you specify more than one next-hop gateway in an IP policy, you can use the **ip-policy set** command to control how the load is distributed among the next-hop gateways. You can cause each new flow to use the first available next-hop gateway in the **ip-policy permit** statement, or you can cause flows to use all the next-hop gateways in the **ip-policy permit** statement sequentially. You can also control which information in the IP packet to use to determine the next-hop gateway.

In addition, you can use the **ip-policy set** command to have the X-Pedition query the availability of the next-hop gateways specified in an IP policy. When this option is active, the X-Pedition periodically queries the next-hop gateways via ICMP_ECHO_REQUESTS. Only gateways that respond to these requests are used for forwarding packets.

Parameters

<name>
Is the name of an IP policy.

pinger on
Causes the X-Pedition to check the availability of next-hop gateways by querying them with ICMP_ECHO_REQUESTS. Only gateways that respond to these requests are used for forwarding packets.

Note: Some hosts may have disabled responding to ICMP_ECHO packets. Make sure each next-hop gateway can respond to ICMP_ECHO packets before using this option.

load-policy round-robin
If an IP policy has more than one next-hop gateway, specifies how the packets are distributed among the gateways. Two options are available:

round-robin Uses a sequential order to pick the next gateway in the list for each new flow.

ip-hash		Uses the following information in the IP packet to determine the next hop gateway.
	sip	Uses the source IP based selection.
	dip	Uses the destination IP based selection.
	both	Uses both source IP and destination IP for selection.

Restrictions

None.

Examples

To set up 10.10.10.10 and 10.10.10.5 as next-hop gateways for IP policy p6:

```
xp(config)# ip-policy p6 permit profile prof1 next-hop-list '10.10.10.10 10.10.10.5'
```

To distribute flows among these two next-hop gateways in a sequential manner:

```
xp(config)# ip-policy p6 set load-policy round-robin
```

ip-policy show

Purpose

Displays information about active IP policies.

Format

ip-policy show [all] [policy-name <*name*>**|all] [interface** <*name*>**|all]**

Mode

Enable

Description

The **ip-policy show** command displays information about active IP policies, including profile definitions, policy configuration settings, and next-hop gateways. The command also displays statistics about packets that have matched an IP policy statement as well as the number of packets that have been forwarded to each next-hop gateway.

Parameters

policy-name <*name*>**|all**
Is the name of an IP policy. Use the **all** keyword to display all active policies.

Note: The **ip-policy show all** command works identically to the **ip-policy show policy-name all** command

interface <*name*>**|all**
Displays information about IP policies applied to a specified interface. When you use the **all** keyword, the command displays information about IP policies applied to all interfaces (i.e., an IP policy used by all interfaces). If no IP policy is shared by all interfaces, the following error message will appear:

%PBR-I-NOALL, No policy applied to all IP interfaces

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

None.

Example

To display information about IP policy p1:

```
xp# ip-policy show policy-name p1

--------------------------------------------------------------------------------
IP Policy name     : p1 ❶
Applied Interfaces : int1  ❷
Load Policy        : first available  ❸

❹             ❺              ❻             ❼       ❽       ❾  ❿
ACL         Source IP/Mask     Dest. IP/Mask      SrcPort   DstPort   TOS Prot
---         --------------     -------------      --------- --------- --- ----
prof1       9.1.1.5/32         15.1.1.2           any       any         0 IP
prof2       2.2.2.2/32         anywhere           any       any         0 IP
everything  anywhere           anywhere           any       any         0 IP

                                        Next Hop Information
                                        --------------------
 ⓫   ⓬    ⓭         ⓮   ⓯                  ⓰          ⓱   ⓲
Seq   Rule   ACL        Cnt Action                Next Hop       Cnt Last
---   ----   --------   --- -----------           --------       --- ----
10    permit prof1      0   Policy Only           11.1.1.2       0   Dwn
20    permit prof2      0   Policy Last           1.1.1.1        0   Dwn
                                                  2.2.2.2        0   Dwn
                                                  3.3.3.3        0   Dwn
999   permit everything 0   Policy Only             drop           N/A N/A
65536 deny   deny       0   N/A                   normal fwd     N/A N/A
㉑
```

Legend:

1. The name of the IP policy.
2. The interface where the IP policy was applied.
3. The load distribution setting for IP-policy statements that have more than one next-hop gateway; either first available (the default) or round-robin.
4. The names of the profiles (created with an **acl** statement) associated with this IP policy.
5. The source address and filtering mask of this flow.
6. The destination address and filtering mask of this flow.
7. For TCP or UDP, the number of the source TCP or UDP port.
8. For TCP or UDP, the number of the destination TCP or UDP port.
9. The TOS value in the packet.
10. IP protocol (ICMP, TCP UDP).

11. The sequence in which the statement is evaluated. IP policy statements are listed in the order they are evaluated (lowest sequence number to highest).
12. The rule to apply to the packets matching the profile: either permit or deny
13. The name of the profile (ACL) of the packets to be forwarded using an IP policy.
14. The number of packets that have matched the profile since the IP policy was applied (or since the **ip-policy clear** command was last used)
15. The method by which IP policies are applied with respect to dynamic or statically configured routes; possible values are Policy First, Policy Only, or Policy Last.
16. The list of next-hop gateways in effect for the policy statement.
17. The number of packets that have been forwarded to this next-hop gateway.
18. The state of the link the last time an attempt was made to forward a packet; possible values are up, dwn, or N/A.
19. Implicit deny rule that is always evaluated last, causing all packets that do not match one of the profiles to be forwarded normally (with dynamic routes).

Chapter 34

ipx Commands

The **ipx** commands let you add entries to the IPX SAP table for SAP servers and display the IPX forwarding database, RIP table, and SAP table.

Command Summary

Table 28 lists the **ipx** commands. The sections following the table describe the command syntax.

Table 28. ipx commands

ipx add route <*networkaddr*> <*nextroutnextnode*> <*metric*> <*ticks*>
ipx add sap <*type*> <*SrvcName*> <*node*> <*socket*> <*metric*> <*interface-network*>
ipx find rip <*address*>
ipx find sap <*type*>|**all** <*SrvcName*>|**all** <*network*>|**all** <*entrytype*>
ipx l3-hash module <*num*>|**all variant** <*num*>
ipx set interface ifname <*string*>| **ipg** <*num*> | **ripintvl** <*num*> | **sapintvl** <*num*>
ipx set rip buffers <*buffer-size*>| **packets-per-iteration** <*num*>
ipx set ripreq buffers <*buffer-size*>| **packets-per-iteration** <*num*>
ipx set sap buffers <*buffer-size*>| **packets-per-iteration** <*num*>|**topn** |**delay** <*num*>
ipx set sapgns buffers <*buffer-size*> |**round-robin** |**packets-per-iteration** <*num*>
ipx set type20 propagation on
ipx set port forwarding-mode destination-based
ipx show buffers

Table 28. ipx commands (Continued)

ipx show hash-variant <*num*>**|all**
ipx show interfaces <*interface*> **|all [brief]**
ipx show rib destination
ipx show servers hops|net|name|type
ipx show summary
ips show routes
ipx show packets-per-iteration
ipx show stack-queues

ipx add route

Purpose

Add an IPX RIP route entry to the routing table.

Format

ipx add route <*networkaddr*> <*nextroutnextnode*> <*metric*> <*ticks*>

Mode

Configure

Description

The **ipx add route** command adds a route into the IPX RIP routing table.

Parameters

<*networkaddr*>	Destination network address.
<*nextroutnextnode*>	Next router's Network.Node address.
<*metric*>	The number of hops to this route. You can specify a number from 0 – 14.
<*ticks*>	Ticks associated with this route.

Restrictions

- Route entries that you add using the **ipx add route** command override dynamically learned entries, regardless of hop count.
- IPX is not supported in partially meshed WAN networks unless each node has a unique network address.

Example

To add an IPX route to IPX network A1B2C3F5 via router A1B2C3D4.00:E0:63:11:11:11 with a metric of 1 and a tick of 100:

```
xp(config)# ipx add route A1B2C3F5 A1B2C3D4.00:E0:63:11:11:11 1 100
```

ipx add sap

Purpose

Add an IPX SAP entry to the routing table.

Format

ipx add sap <*type*> <*SrvcName*> <*node*> <*socket*> <*metric*> <*interface-network*>

Mode

Configure

Description

The **ipx add sap** command adds an entry for an IPX server to the IPX SAP table.

Parameters

<*type*>	The type of service. Specify the service type using its hexadecimal value.
<*SrvcName*>	Name of the IPX server. You can use any characters in the name except the following: " * . / : ; < = > ? [ ] \ |] **Note:**Lowercase characters are changed to uppercase characters.
<*node*>	*The* IPX network and node address. Specify the address in the following format: <*netaddr*>**.**<*macaddr*>. Example: a1b2c3d4.aa:bb:cc:dd:ee:ff.
<*socket*>	The socket number for this SAP entry. You can specify a Hexadecimal number from 0x0 – 0xFFFF.
<*metric*>	The number of hops to the server. You can specify a number from 1 – 14.
<*interface-network*>	The interface network associated with this SAP entry.

Restrictions

SAP entries that you add using the **ipx add sap** command override dynamically learned entries, regardless of hop count. Moreover, if a dynamic route entry that is associated with the static SAP entry ages out or is deleted, the X-Pedition does not advertise the corresponding static SAP entries for the service until it relearns the route.

ipx find rip

Purpose

Find an IPX address in the routing table.

Format

ipx find rip <*address*>

Mode

Enable

Description

The **ipx find rip** command searches for an IPX address in the routing table.

Parameter

<*address*> The IPX network address of this interface. Specify the IPX address using its hexadecimal value.

Restrictions

None.

Example

To find an IPX network in the route table:

```
xp(config)# ipx find rip A1B2C3F5
```

ipx find sap

Purpose

Find a SAP entry in the routing table.

Format

ipx find sap <*type*>|**all** <*SrvcName*>|**all** <*network*>|**all** <*entrytype*>

Mode

Enable

Description

The **ipx find sap** command searches for a SAP entry in the routing table.

Parameters

<*type*>|**all** The types of service. Specify the service type using its hexadecimal value. Specify **all** for all types of service.

<*SrvcName*>|**all**
Name of the IPX service. You can use any characters in the name except the following: " * . / : ; < = > ? [ ] \ |

Note: Lowercase characters are changed to uppercase characters.

Specify **all** for all IPX services.

<*network*>|**all**
Network on which the service resides. Specify an IPX network address in the following format: <*netaddr.*> Example: a1b2c3d4. Specify **all** for all networks.

<*entrytype*> The types of entry you want to find. Specify one of the following:

all Finds static and dynamic SAP entries.

dynamic Finds only the dynamic SAP entries.

static Finds only the static SAP entries.

Restrictions

None.

Example

To find a SAP entry in the route table:

```
xp(config)# ipx find sap 4 FILESERVER a2b2c3d4 dynamic
```

ipx l3-hash

Purpose

Changes the hashing algorithm used for the L3 IPX lookup table.

Format

ipx l3-hash module <*num*>**|all variant** <*num*>

Mode

Configure

Description

The X-Pedition's L3 Lookup table is organized as a hash table. The hash function reduces the destination and source MAC addresses to 16-bit quantities each. The hashing algorithm generates a uniform distribution within the MAC address space. However, given a particular set of addresses, the distribution may cause addresses to clump together in the table. To minimize the risk of thrashing in the tables, three variations to the basic hashing algorithm are defined. Only one variation is in effect on a line card at any given time. You can use the ipx **l3-hash** command to set which variation is in effect for a line card.

Swizzling shifts the hash value by a certain amount of bits, producing more random distribution across the L3 lookup table.

Auto-hashing periodically queries the L2 or L3 tables for hash bucket overflow on a port. If there are more overflows than a certain threshold level, auto-hashing will automatically change the hash mode for that port. Eventually a 'best' hash mode for the particular traffic will be found, which will provide a more even distribution across the L2 or L3 lookup table.

Parameters

module <*num*>**|all**
Is a slot number on the X-Pedition. Specify any number between 1 and 16. The hashing algorithm change affects all ports on the line card in the slot. The **all** option causes the hashing algorithm to change on all ports on all slits.

variant <*num*>
Causes a variation to the basic hashing algorithm to be made. Valid variant numbers are: 0-3, 4-7 (swizzled), and 8 (auto-hashed). If you specify 0, the default hashing algorithm is used.

Restrictions

None.

Example

To change the default hashing algorithm used for the L3 lookup table on all ports on slot 7:

```
xp(config)# ipx l3-hash module 7 variant 1
```

ipx set interface

Purpose

Sets the IPX interface parameters.

Format

ipx set interface ifname <*string*>**| ipg** <*num*> **| ripintvl** <*num*> **| sapintvl** <*num*>

Mode

Configure

Description

The **ipx set interface** command sets the IPX interface parameters such as interface name, inter-packet gap, broadcast interval for RIP, and broadcast interval for SAP.

Parameter

ifname <*string*> Specify the interface name.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Note:

ipg <*num*> Specify the Inter Packet Gap (in milliseconds). Specify any number between 30 and 180.

ripintvl <*num*> Specify the broadcast interval for RIP (in seconds). Specify any number between 60 and 300.

sapintvl <*num*> Specify the broadcast interval for SAP (in seconds). Specify any number between 60 and 300.

Restrictions

None.

ipx set rip

Purpose

Sets the RIP socket buffer size in bytes or the number of packets per iteration.

Format

ipx set rip buffers <*buffer-size*> **| packets-per-iteration** <*num*>

Mode

Configure

Description

The **ipx set rip buffers** command sets the RIP socket buffer size or the number of rip packets processed per iteration.

Parameters

buffers <*buffer-size*>	Specify the socket buffer size in bytes.
packets-per-iteration <*num*>	Specify the number of rip packets to be processed per iteration.

Restrictions

None.

ipx set ripreq

Purpose

Sets the buffer size or number of packets per iteration for rip requests.

Format

ipx set ripreq buffers <*buffer-size*> |**packets-per-iteration** <*num*>

Mode

Configure

Description

The **ipx set ripreq** command sets the rip buffer size or the number of packets per iteration.

Parameters

buffers <*buffer-size*>	Size of the buffer in bytes.
packets-per-iteration <*num*>	The number of rip request packets processed per iteration.

Restrictions

None.

ipx set sap

Purpose

Sets various SAP socket parameters.

Format

ipx set sap buffers <*buffer-size*> **|packets-per-iteration** <*num*> **|topn |delay** <*num*>

Mode

Configure

Description

The **ipx set sap** command sets various SAP socket parameters.

Parameter

buffers <*buffer-size*>	Specify the buffer size in bytes.
packets-per-iteration <*num*>	The number of SAP packets processed per iteration.
topn	Send only the nearest N services for a general reply.
delay <*num*>	Delay multiple to be used for general SAP requests.

Restrictions

None.

ipx set sapgns

Purpose

Sets parameters for sap get nearest server packets.

Format

ipx set sapgns buffers <*buffer-size*>**|round-robin |packets-per-iteration** <*num*>

Mode

Configure

Description

The **ipx set sapgns** command sets the following parameters for get nearest server packets:

- sets buffer size
- sets a round-robin scheme for finding servers

Parameter

buffers <*buffer-size*>	Specify the buffer size in bytes.
round-robin	Sets a round-robin scheme for finding the nearest server.
packets-per-iteration <*num*>	The number of SAP Get Nearest Server packets processed per iteration.

Restrictions

None.

ipx set type20 propagation

Purpose

Controls the propagation of type 20 packets.

Format

ipx set type20 propagation on

Mode

Configure

Description

The **ipx set type20 propagation** command controls the propagation of type 20 packets.

Parameter

None.

Restrictions

None.

ipx set port

Purpose

Configures an IPX port for forwarding mode.

Format

ipx set port forwarding-mode destination-based

Mode

Configure

Description

The **ipx set port forwarding-mode destination-based** command sets up an IPX port to forward traffic based on the packet destination network, node, and socket.

Parameter

None.

Restrictions

None.

ipx show buffers

Purpose

Display the RIP and SAP socket buffer sizes.

Format

ipx show buffers

Mode

Enable

Description

The **ipx show buffers** command displays the RIP and SAP socket buffer sizes.

Parameters

Restrictions

None.

ipx show hash-variant

Purpose

Display IPX hash variant per module.

Format

ipx show hash-variant <*num*>|**all**

Mode

Enable

Description

The **ipx show hash-variant** command displays hash variant information. There are a total of 16 modules using the hash variant feature (1-16).

Enabling hash variant causes a variation to the basic hashing algorithm. This variation will prevent clustering of hash values and will provide a more even distribution across the L3 lookup table. Valid variant numbers are: 0-3, 4-7 (swizzled), and 8 (auto-hashed). The default hashing algorithm is 0.

Swizzling shifts the hash value by a certain amount of bits, causing a more random distribution across the L3 lookup table. Auto-hashing allows the X-Pedition to auto-select a hashing algorithm optimized for ‘best case’ L3 table distribution.

Parameters

<*num*>|**all**	Specifies the module. Specify any number between 1-16. Specify **all** to display hash variant information for all modules.

Restrictions

None.

Example

To display IPX hash variant information on all 16 modules:

```
xp# ipx show hash-variant all
IPX Module                    Hash Variant
-----------------------------------------------------
Module 2                      variant-3
Module 3                      variant-0
Module 4                      variant-0
Module 5                      variant-0
Module 6                      variant-5
Module 7                      variant-0
Module 8                      variant-0
Module 9                      variant-0
Module 10                     variant-0
Module 11                     variant-2
Module 12                     variant-2
Module 13                     variant-0
Module 14                     variant-0
Module 15                     variant-1
```

ipx show interfaces

Purpose

Display the configuration of IPX interfaces.

Format

ipx show interfaces <*interface*>|**all [brief]**

Mode

Enable

Description

The **ipx show interfaces** command displays the configuration of an IPX interface. If you issue the command without specifying an interface name then the configuration of all IPX interfaces is displayed.

Parameters

<*interface*>|**all** Name of the IPX interface (for example, xp14) or all interfaces.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

brief Shows a brief summary of the interface in tabular form.

Restrictions

If you specify an interface name, the name must belong to an existing IPX interface.

Example

To display the configuration of all IPX interfaces:

```
xp# ipx show interfaces all
xp12: flags=9863<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,LINK0,MULTICAST>
      VLAN: _VLAN-1
      Ports: et.1.7
      IPX: A1B2C3D4.00:E0:63:11:11:11
xp14: flags=9863<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,LINK0,MULTICAST>
      VLAN: _VLAN-2
      Ports: et.1.2
      IPX: ABCD1234.00:E0:63:11:11:11
```

ipx show rib destination

Purpose

Show IPX RIP table output sorted by destination.

Format

ipx show rib destination

Mode

User

Description

The **ipx show rib destination** command displays IPX RIP table output sorted by destination.

Parameters

None.

Restrictions

None.

ipx show servers

Purpose

Show IPX server information.

Format

ipx show servers hops|net|name|type

Mode

User

Description

The **ipx show servers** command displays IPX server information sorted by any or all of the optional arguments. Sorting is done based on the order of optional arguments given.

Parameters

hops	Shows the number of hops that the service is away.
name	Shows the Sap service name.
net	Shows the interface type over which the service arrived.
type	Shows the Sap service type.

Restrictions

None.

ipx show summary

Purpose

Displays summary of the IPX RIP/SAP tables.

Format

ipx show summary

Mode

User

Description

The **ipx show summary** command displays a summary of the IPX RIP/SAP tables.

Parameters

None.

Restrictions

None.

ipx show routes

Purpose

Displays information for IPX routes.

Format

ipx show routes

Mode

User

Description

The **ipx show routes** command displays information for all IPX routes.

Parameters

None.

Restrictions

None.

ipx show packets-per-iteration

Purpose

Display the number of IPX control packets processed per iteration.

Format

ipx show packets-per-iteration

Mode

User

Description

The **ipx show packets-per-user** command displays the number of IPX control packets processed per iteration for rip, rip request, SAP, and SAP GNS.

Parameters

None.

Restrictions

None.

ipx show stack-queues

Purpose

Displays information for IPX stack queues.

Format

ipx show stack-queues

Mode

User

Description

The **ipx show stack-queues** command displays size and drop information for the IPX stack queues.

Parameters

None.

Restrictions

None.

Chapter 35

l2-tables Commands

The **l2-tables** commands let you display various L2 tables related to MAC addresses.

Command Summary

Table 29 lists the **l2-tables** commands. The sections following the table describe the command syntax.

Table 29. l2-tables commands

l2-tables show all-flows [vlan <*VLAN-num*> **[source-mac** <*MACaddr*>**]] [undecoded]**
l2-tables show all-macs [verbose [undecoded]] **[vlan** <*VLAN-num*>**] [source] [destination] [multicast]**
l2-tables show bridge-management
l2-tables show igmp-mcast-registrations [vlan <*VLAN-num*>**]**
l2-tables show mac <*MACaddr*> **vlan** <*VLAN-num*>
l2-tables show mac-table-stats
l2-tables show port-macs port <*port-list*>**|all-ports** **[[vlan** <*VLAN-num*>**] [source] [destination] [multicast] [undecoded] [no-stats]** **[verbose] [decode-smarttrunks]]**
l2-tables show vlan-igmp-status vlan <*VLAN-num*>
l2-tables show system-macs

l2-tables show all-flows

Purpose

Show all L2 flows (for ports in flow-bridging mode).

Format

l2-tables show all-flows [vlan <*VLAN-num*> **[source-mac** <*MACaddr*>**]] [undecoded]**

Mode

User or Enable

Description

The **l2-tables show all-flows** command shows all the L2 flows learned by the X-Pedition. The X-Pedition learns flows on ports that are operating in flow-bridging mode.

Parameters

vlan <*VLAN-num*>
The VLAN number (1-4095) associated with the flows. There are two special VLANs on the X-Pedition—the Default VLAN (ID=1) and the VLAN with ID=4095 (i.e., the "Blackhole VLAN").

The Default VLAN contains all ports not in use by other VLANs. When you add ports to or remove them from a VLAN (with an ID other than 1), the X-Pedition removes the ports from or adds them to the Default VLAN. The VLAN with the ID of 4095 serves as a repository for incoming frames with no destination. The following restrictions apply to these VLANs:

- You may not add ports explicitly to either of these VLANs.
- You cannot associate Layer 3 interfaces with the "Blackhole VLAN" (ID = 4095).
- You cannot associate IPX interfaces with the "Default VLAN" (ID = 1).

In order to pass all IBM protocol types, you must configure a unique VLAN for both the SNA and the Bridged protocols. Additional information can be found online.

source-mac <*MACaddr*>
The source MAC address of the flows. Specify the MAC address in either of the following formats:

xx:xx:xx:xx:xx:xx
xxxxxx:xxxxxx

undecoded

Prevents the **X-Pedition** from displaying the vendor names with the MAC addresses. Instead, the OUI of each MAC address is displayed "as is," in hexadecimal format. If you do not use this option, the **X-Pedition** decodes the OUI and displays the vendor name.

Restrictions

None.

l2-tables show all-macs

Purpose

Show all MAC addresses currently in the L2 tables.

Format

l2-tables show all-macs [verbose [undecoded]]
[vlan <*VLAN-num*>**] [source] [destination] [multicast]**

Mode

User or Enable

Description

The **l2-tables show all-macs** command shows how many MAC addresses the X-Pedition has in its L2 tables. You can format the displayed information based on VLAN, source MAC address, destination MAC address or multicast. If you enter the verbose option, the command also shows the individual MAC addresses.

Parameters

vlan <*VLAN-num*>	Displays only MAC addresses in the specified VLAN.
source	Displays only source addresses.
destination	Displays only destination addresses.
multicast	Displays only multicast and broadcast addresses.
verbose	Shows detailed information for each MAC address entry.
undecoded	Prevents the X-Pedition from displaying the vendor names with the MAC addresses. Instead, the OUI of each MAC address is displayed “as is,” in hexadecimal format. If you do not use this option, the X-Pedition decodes the OUI and displays the vendor name.

Restrictions

None.

l2-tables show bridge-management

Purpose

Show information about all MAC addresses registered by the system.

Format

l2-tables show bridge-management

Mode

User or Enable

Description

The **l2-tables show bridge-management** command shows MAC addresses that have been inserted into the L2 tables for management purposes. Generally, these entries are configured so that a port forwards a frame to the Control Module if the management MAC matches the frame's destination MAC.

An example of a bridge-management MAC is Spanning Tree's bridge group address (0180C2:000000), which is registered in the L2 tables of X-Pedition ports on which the Spanning Tree Protocol (STP) is enabled.

Parameters

None.

Restrictions

None.

l2-tables show igmp-mcast-registrations

Purpose

Show information about multicast MAC addresses registered by IGMP.

Format

l2-tables show igmp-mcast-registrations [vlan <*VLAN-num*>**]**

Mode

User or Enable

Description

The **l2-tables show igmp-mcast-registrations** command shows the multicast MAC addresses that IGMP has registered with the L2 tables. The X-Pedition forwards the multicast MAC addresses only to the ports that IGMP specifies.

Parameters

vlan <*VLAN-num*>	Displays only the multicast MAC addresses registered for the specified VLAN.

Restrictions

None.

l2-tables show mac

Purpose

Show information about a particular MAC address.

Format

l2-tables show mac <*MACaddr*> **vlan** <*VLAN-num*>

Mode

User or Enable

Description

The **l2-tables show mac** command shows the port number on which the specified MAC address resides.

Parameters

<*MACaddr*> Is a MAC address. You can specify the address in either of the following formats:

xx:xx:xx:xx:xx:xx
xxxxxx:xxxxxx

vlan <*VLAN-num*> Displays the MAC address for this VLAN.

Restrictions

None.

l2-tables show mac-table-stats

Purpose

Show statistics for the MAC addresses in the MAC address tables.

Format

l2-tables show mac-table-stats

Mode

User or Enable

Description

The **l2-tables show mac-table-stats** command shows statistics for the master MAC address table in the Control Module and the MAC address tables on the individual ports.

Parameters

None.

Restrictions

None.

l2-tables show port-macs

Purpose

Show information about MACs residing in a port's L2 table.

Format

l2-tables show port-macs port <*port-list*>**|all-ports**
[[vlan <*VLAN-num*>**] [source] [destination] [multicast] [undecoded] [no-stats] [verbose] [decode-smarttrunks]]**

Mode

User or Enable

Description

The **l2-tables show port-macs** command shows the information about the learned MAC addresses in individual L2 MAC address tables. Each port has its own MAC address table. The information includes the number of source MAc addresses and the number of destination MAC addresses in the table. If you enter the **verbose** option, the MAC addresses also are displayed.

Parameters

port <*port-list*>**|all-ports**
Specifies the port(s) for which you want to display MAC address information. You can specify a single port or a comma-separated list of ports. If you use the **all-ports** keyword, MAC address information is displayed for all ports.

vlan <*VLAN-num*>
Specifies the type of MAC address for which you want to show statistics.

source
Displays statistics for only source addresses.

destination
Displays statistics for only destination addresses.

multicast
Displays statistics for only multicast and broadcast addresses.

undecoded
Displays the MAC addresses in hexadecimal format rather than undecoded format. Undecoded format does not show the vendor name in place of the first three hexadecimal digits (example: Enterasys:33:44:55). The default is undecoded (example: 00:11:22:33:44:55).

no-stats
Lists the MAC addresses without displaying any statistics.

verbose
Shows detailed statistics for each MAC address entry.

decode-smarttrunks
Shows l2 table information for SmartTRUNK ports.

Restrictions

None.

Example

```
xp(l2-tables-show)# port-macs port et.1.2 decode-smarttrunks


L2 table information for port et.1.2
-----------------------------------------
Number of source MAC addresses: 0
Number of destination MAC addresses: 0
Number of management-configured MAC addresses: 3
Port table capacity: 5888
Port table demand deletion upper & lower thresholds: 95% - 85%
Number of times table usage has reached upper threshold: 0
Number of times buckets have become full: 0
Number of duplicate learning frames: 0
Number of times LG port got out-of-sync: 0
Number of requests to learn a frame on an invalid VLAN: 0
Number of frames received from this switch (possible loop): 0
Aging is enabled
Addresses will be aged-out after 300 seconds
```

l2-tables show vlan-igmp-status

Purpose

Show whether IGMP is on or off on a VLAN.

Format

l2-tables show vlan-igmp-status vlan <*VLAN-num*>

Mode

Enable

Description

The **l2-tables show vlan-igmp-status** command shows the multicast MAC addresses that IGMP has registered with the L2 tables. This command also shows the ports to which the multicast MAC addresses are forwarded.

Note: For IGMP forwarding to occur for a multicast MAC address, IGMP must be enabled on the VLAN with which the MAC address is associated.

Parameters

vlan <*VLAN-num*>

The VLAN number (1-4095) associated with the flows. There are two special VLANs on the X-Pedition—the Default VLAN (ID=1) and the VLAN with ID=4095 (i.e., the “Blackhole VLAN”).

The Default VLAN contains all ports not in use by other VLANs. When you add ports to or remove them from a VLAN (with an ID other than 1), the X-Pedition removes the ports from or adds them to the Default VLAN. The VLAN with the ID of 4095 serves as a repository for incoming frames with no destination. The following restrictions apply to these VLANs:

- You may not add ports explicitly to either of these VLANs.
- You cannot associate Layer 3 interfaces with the “Blackhole VLAN” (ID = 4095).
- You cannot associate IPX interfaces with the “Default VLAN” (ID = 1).

In order to pass all IBM protocol types, you must configure a unique VLAN for both the SNA and the Bridged protocols. Additional information can be found online.

Restrictions

None.

l2-tables show system-macs

Purpose

To display information about specific MACs registered by the system.

Format

l2-tables show system-macs

Mode

Enable

Description

The **l2-tables show system-macs** command displays information about MACs auto-registered by the system (e.g., VRRP MAC addresses).

Parameters

None.

Restrictions

None.

Example

```
xp(l2-tables-show)# system-macs

Name:              VRRP Virtual-MAC
-------            -------------------------
Direction:         Source
Restriction:       force-to-go
VLAN:              30
Source MAC:        00005E:000102
In-List ports:     et.1.(1-2)
```

Chapter 36

load-balance Commands

The **load-balance** commands allow you to distribute session load across a pool of servers. These commands provide a way to load balance network traffic to multiple servers.

Command Summary

Table 30 lists the **load-balance** commands. The sections following the table describe the command syntax.

Table 30. load-balance commands

load-balance add host-to-group <*ipaddr/range*> **group-name** <*group name*> **port** <*port number*> [**weight** <*weight*>]
load-balance add host-to-vip-range <*range*> **vip-range-name** <*range name*> **port** <*port number*> [**weight** <*weight*>]
load-balance allow access-to-servers client-ip <*ipaddr/range*> **group-name** <*group name*>
load-balance create group-name <*group name*> **virtual-ip** <*ipaddr*> **virtual-port** <*port number*> **protocol tcp|udp [persistence-level vpn| tcp|ssl| sticky]**
load-balance create vip-range-name <*range name*> **vip-range** <*range*> **virtual-port** <*port number*> **protocol tcp|udp [persistence-level vpn| tcp|ssl| sticky]**
load-balance set aging-for-src-maps <*string*> **aging-time** <*num*>
load-balance set client-proxy-subnet <*group name*> **subnet** <*num*>
load-balance set ftp-control-port <*port number*>
load-balance set group-options <*string*> **[ping-int** <*num*>**] [ping-tries** <*num*>**] [app-int** <*num*>**] [app-tries** <*num*>**] [acv-command** <*string*> **acv-reply** <*string*>**] [acv-quit** <*string*>**] [read-till-index** <*num*>**] [check-port** <*port number*>**]**

Table 30. load-balance commands (Continued)

load-balance set group-conn-thresh <*string*> **limit** <*num*>
load-balance set hash-variant <*value*>
load-balance set policy-for-group <*group name*> **policy** <*policy*>
load-balance set server-status server-ip <*ipaddr/range*> **server-port** <*port number*> **group-name** <*group name*> **status up**|**down**
load-balance set server-options <*ipaddr*> **[port** <*num*>**] [ping-int** <*num*>**] [ping-tries** <*num*>**] [app-int** <*num*>**] [app-tries** <*num*>**] [acv-command** <*string*> **acv-reply** <*string*>**] [acv-quit** <*string*>**] [read-till-index** <*num*>**] [check-port** <*port number*>**]**
load-balance set vpn-dest-port <*num*>
load-balance show acv-options [group-name <*string*>**] [destination-host-ip** <*ipaddr*>**] [destination-host-port** <*num*>**]**
load-balance show hash-stats
load-balance show source-mappings client-ip <*ipaddr*> **virtual-ip** <*ipaddr*> **virtual-port** <*port number*> **destination-host-ip** <*ipaddr*>
load-balance show statistics group-name <*group name*> **virtual-ip** <*ipaddr*> **virtual-port** <*port number*>
load-balance show virtual-hosts group-name <*group name*> **virtual-ip** <*ipaddr*> **virtual-port** <*port number*>

load-balance add host-to-group

Purpose

Adds a server to a previously-created group of load balancing servers.

Format

load-balance add host-to-group <*ipaddr/range*> **group-name** <*group name*> **port** <*port number*> [**weight** <*weight*>]

Mode

Configure

Description

The **load-balance add host-to-group** command lets you add a server to a server group that was previously-created with the **load-balance create group-name** command.

Parameters

host-to-group <*ipaddr/range*>

The IP address of the server being added to the group, in the form a.b.c.d or a range of IP addresses in the form 10.10.1.1-10.10.1.3.

Note: Do not use an IP address for load-balancing that is already configured for VRRP.

group-name <*group name*>

The name of the group of load balancing servers.

port <*port number*>

The port number to be used for load balancing communications for the server being added. Specify a number between 1 and 65535.

weight <*weight*>

This parameter is only valid if you specify the weighted round robin policy for this group of load balancing servers. (The **load-balance set policy-for-group** command specifies the policy for distributing workload to the servers.) The weight determines how many sessions are assigned to this server during its turn in the weighted round robin selection. Specify a number between 1 and 65535. The default value is 1.

Restrictions

Do not use an IP address for load-balancing that is already configured for VRRP.

Examples

To add a server 10.10.13.2 to the server group ‘service2’:

```
xp(config)# load-balance add host-to-group 10.10.13.2 group-name service2 port 80
```

To add servers 10.10.13.3, 10.10.13.4, and 10.10.13.5 to the server group ‘service2’:

```
xp(config)# load-balance add host-to-group 10.10.13.3-10.10.13.5 group-name service2 port 80
```

The following is an example of specifying the weighted round robin policy for distributing the workload on the server group ‘service2.’ To add servers 10.10.13.3, 10.10.13.4, and 10.10.13.5 to the server group ‘service2,’ a weight must be assigned to each server in the group:

```
xp(config)# load-balance set policy-for-group service2 policy weighted-round-robin
xp(config)# load-balance add host-to-group 10.10.13.3 group-name service2 port 80 weight 10
xp(config)# load-balance add host-to-group 10.10.13.4 group-name service2 port 80 weight 100
xp(config)# load-balance add host-to-group 10.10.13.5 group-name service2 port 80 weight 1000
```

load-balance add host-to-vip-range

Purpose

Adds a range of servers to a range of virtual IP addresses that were created with the **load-balance create vip-range-name** command.

Format

load-balance add host-to-vip-range <*range*> **vip-range-name** <*range name*> **port** <*port number*> [**weight** <*weight*>]

Mode

Configure

Description

The **load-balance add host-to-vip-range** command lets you add a range of servers to a range of virtual IP addresses that were previously created with the **load-balance create vip-range-name** command. This command adds the first server address in the range to the first virtual IP address, the second server address to the second virtual IP address, and so on. Therefore, the number of servers in the specified range must *equal* the number of virtual IP addresses; if you specified 15 virtual IP addresses with the l**oad-balance create vip-range-name** command, then you must specify a range of 15 IP addresses in the **load-balance add host-to-vip-range** command.

Parameters

host-to-vip-range <*range*>

The IP range of the servers being added to the range, in the form 10.10.1.1-10.10.1.3. The number of servers in the range must be the same as the number of virtual IP addresses that were previously-created.

Note: Do not use an IP address for load-balancing that is already configured for VRRP.

vip-range-name <*range name*>

The name of the range of load balancing servers.

Note: Do not use an IP address for load-balancing that is already configured for VRRP.

port <*port number*>

The port number to be used for load balancing communications for the server being added. Specify a number between 1 and 65535.

weight <*weight*>

This parameter is only valid if you specify the weighted round robin policy for this group of load balancing servers. (The **load-balance set policy-for-group** command specifies the policy for distributing workload to the servers.) The weight determines how many

sessions are assigned to this server during its turn in the weighted round robin selection. Specify a number between 1 and 65535. The default value is 1.

Restrictions

None.

Examples

The following command creates the server groups ‘service1’ through ‘service15’ with virtual IP addresses 207.135.89.1 through 207.135.89.15:

```
xp(config)# load-balance create vip-range-name service vip-range 207.135.89.1-207.135.89.15 virtual-port
80 protocol tcp
```

To add servers 10.10.13.1-10.10.13.15 to the server groups ‘service1’ through ‘service15’:

```
xp(config)# load-balance add host-to-vip-range 10.10.13.1-10.10.13.15 vip-range-name service port 80
```

load-balance allow access-to-servers

Purpose

Allows specified hosts to access the load balancing servers without address translation.

Format

load-balance allow access-to-servers client-ip <*ipaddr/range*> **group-name** <*group name*>

Mode

Configure

Description

Load balancing causes both source and destination addresses to be translated on the X-Pedition. It may be undesirable in some cases for a source address to be translated; for example, when data is to be updated on each individual server. The **load-balance allow access-to-servers** command lets you specify the hosts which are allowed to access a group of load balancing servers without address translation.

Note that a host that is allowed to access a group of load balancing servers without address translation *cannot* use the virtual IP address and port to access servers in the group.

Parameters

client-ip <*ipaddr/range*>
The IP address of the host that is to be granted direct access, in the form a.b.c.d or a range of IP addresses in the form 10.10.1.1-10.10.1.3.

Note: Do not use an IP address for load-balancing that is already configured for VRRP.

group-name <*group name*>
The name of the group of load balancing servers.

Restrictions

None.

Examples

To allow the host 10.23.4.8 to directly access the server group ‘service2’:

```
xp(config)# load-balance allow access-to-servers client-ip 10.23.4.8 group-name service2
```

load-balance create group-name

Purpose

Creates a server group for load balancing.

Format

load-balance create group-name <*group name*> **virtual-ip** <*ipaddr*> **virtual-port** <*port number*> **protocol tcp|udp [persistence-level vpn|tcp|ssl| sticky]**

Mode

Configure

Description

The **load-balance create group-name** command lets you create a load balancing server group and specify a unique "virtual" IP address and port number that is used by a client to access any server in the group. You must also specify the protocol (for example, TCP for HTTP and FTP sessions) to be used by the load balancing servers. After you create the group with this command, use the **load-balance add host** command to add specific server systems to the group.

Note: If you want to create many groups, each with a virtual IP address, use the **load-balance create vip-range-name** command. Do not use an IP address for load-balancing that is already configured for VRRP.

Parameters

group-name <*group name*>

The name of this group of load balancing servers.

virtual-ip <*ipaddr*>

The address in the form a.b.c.d that will be used as the IP address for this group.

Note: Do not use an IP address for load-balancing that is already configured for VRRP.

virtual-port <*port number*>

The port number to be used for this group. Specify a number between 1 and 65535.

Note: You cannot specify port number 20, as it is the FTP data port. If you create a group on the FTP control port for FTP, an implicit group will be created on port number 20.

protocol tcp|udp

The protocol used by this group of load balancing servers.

persistence-level vpn|tcp|ssl|sticky

The level of persistence to use for the bindings or connections, either **vpn**, **tcp** (TCP), **ssl** (secure socket layer), or **sticky**. **tcp** is the default if the **persistence-level** parameter is not specified. **Sticky** connections allow a client to connect to the same real server as in previous connections.

Restrictions

The X-Pedition allows users to create two load balance groups that use the same address and port as long as the groups use the same protocol (i.e., UDP or TCP). If a user attempts to create groups with different protocols, the following error messages will appear:

```
%CLI-E-FAILED, Execution failed for “load-balance create group-name abc2 virtual-ip 10.10.10.1 virtual-port 12
protocol tcp”
%LOADBAL-E-VIPPORTUSED, Virtual IP 10.10.10.1 and Virtual Port 12 combination is already used
```

Note: Do not use an IP address for load-balancing that is already configured for VRRP.

Examples

To configure the server group ‘service2’:

```
xp(config)# load-balance create group-name service2 virtual-ip 10.10.100.100 virtual-port 80 protocol tcp
```

load-balance create vip-range-name

Purpose

Creates a group of servers for load balancing.

Format

load-balance create vip-range-name <*range name*> **vip-range** <*range*> **virtual-port** <*port number*> **protocol tcp|udp [persistence-level vpn|tcp|ssl| sticky]**

Mode

Configure

Description

The **load-balance create vip-range-name** command lets you specify a range of "virtual" IP addresses and a port number that is used by a client to access a server in the virtual IP address range. You must also specify the protocol (for example, TCP for HTTP and FTP sessions) to be used by the load balancing servers.

This command *implicitly* creates separate server groups for each virtual IP address in the specified range. The <*range name*> you specify becomes the base group name. Thus, the command **load-balance create vip-range-name myrange vip-range 207.135.89.1-207.135.89.15 virtual-port 80 protocol tcp** creates the groups 'myrange1' with virtual IP address 207.135.89.1, 'myrange2' with virtual IP address 207.135.89.2, etc. This command allows you to create *multiple* server groups, each with unique virtual IP addresses, whereas the **load-balance create group-name** command allows you to only create a *single* group with a *single* virtual IP address.

After you create groups with this command, you can use the **load-balance add host-to-group** command to identify specific server systems in each group. Or, you can use the **load-balance add host-to-vip-range** command to add a range of server IP addresses to each group.

Parameters

vip-range-name <*range name*>

The base group name for this range of load balancing servers.

vip-range <*range*>

The range of virtual IP addresses to be created.

virtual-port <*port number*>

The port number to be used for this virtual IP range. Specify a number between 1 and 65535.

Note: You cannot specify port number 20, as it is the FTP data port.

protocol tcp|udp

The protocol used by this virtual IP range.

persistence-level vpn|tcp|ssl|sticky

The level of persistence to use for the bindings, either vpn, **tcp** (TCP), or **ssl** (secure socket layer). **tcp** is the default if the **persistence-level** parameter is not specified. **Sticky** connections allow a client to connect to the same real server as in previous connections.

Restrictions

None.

Examples

To configure the server groups ‘service1’ through ‘service15’:

```
xp(config)# load-balance create vip-range-name service vip-range 207.135.89.1-207.135.89.15 virtual-port
80 protocol tcp
```

load-balance set aging-for-src-maps

Purpose

Set the aging time for the mappings of a group.

Format

load-balance set aging-for-src-maps <*string*> **aging-time** <*num*>

Mode

Configure

Description

The **load-balance set aging-for-src-maps** command sets the aging time for server group mapping. Once the aging time has expired, mapping from a client to a selected server within the group is cleared. This allows the user to better configure timeout values to specific server groups instead of using a general timeout value for all groups.

Parameters

aging-for-src-maps <*string*>
Specifies the name of the server group.

aging-time <*num*>
Specifies the aging time in minutes. Specify a number between 1 and 4320. The default values depend on which persistence level is selected for a group. Persistence levels vpn and tcp has a default value of 3 minutes. Persistence levels ssl and sticky has a default value of 120 minutes.

Restrictions

None.

Example

To set the aging time to 120 minutes for the server group ‘group1’:

```
xp(config)# load-balance set aging-for-src-maps group1 aging-time 120
```

load-balance set client-proxy-subnet

Purpose

Set the subnet for client address range mapping.

Format

load-balance set client-proxy-subnet <*group name*> **subnet** <*num*>

Mode

Configure

Description

The **load-balance set client-proxy-subnet** command sets the subnet used for mapping clients to a specific server group.

Parameters

client-proxy-subnet <*group name*>
Specifies the name of the server group.

subnet <*num*>
Specifies the subnet. Specify a number between 1 and 31.

Restrictions

None.

Example

To set the subnet number to 10 for the server group ‘group1’:

```
xp(config)# load-balance set client-proxy-subnet group1 subnet 10
```

load-balance set ftp-control-port

Purpose

Specifies the port for FTP control.

Format

load-balance set ftp-control-port <*port number*>

Mode

Configure

Description

File Transfer Protocol (FTP) packets require special handling with load balancing, because IP address information is contained within the FTP packet data. You can use the **load-balance set ftp-control-port** command to specify the port number that is used for FTP control. The default is port 21.

Parameters

ftp-control-port <*port number*>
Specifies the port number used for FTP control. Specify a value between 1 and 65535.

Restrictions

None.

Example

To set the FTP control port to 5000:

```
xp(config)# load-balance set ftp-control-port 5000
```

load-balance set group-options

Purpose

Sets options for a virtual group.

Format

load-balance set group-options <*string*> **[ping-int** <*num*>**] [ping-tries** <*num*>**] [app-int** <*num*>**] [app-tries** <*num*>**] [acv-command** <*string*> **acv-reply** <*string*>**] [acv-quit** <*string*>**] [read-till-index** <*num*>**] [check-port** <*port number*>**]**

Mode

Configure

Description

The **load-balance set group-options** command allows you to set various parameters for checking server content of a load balancing server group. This group must already be created with the **load-balance create group-name** command.

Parameters

group-options <*string*>
The name of the group of load balancing servers.

ping-int
Use this parameter to set the ping interval (seconds) for servers in this group. Specify any value between 5 and 3600. The default value is 5.

ping-tries
Use this parameter to set the number of ping retries before marking the server down. Specify any value between 4 and 255. The default value is 4.

app-int
Use this parameter to set the interval (seconds) between application checks. Specify any value between 5 and 3600. The default value is 15.

app-tries
Use this parameter to set the number of retries before marking the application down. Specify any value between 4 and 255. The default value is 4.

acv-command
The command or series of commands to pass to the server to request the desired file or response. For example, an HTTP command might be: “GET /index.html HTTP/1.1\r\nHost: <server IP>\r\n\r\n”

Note: You must explicitly add any carriage returns (\r) or line feeds (\n) to the end of any command you pass to the server.

acv-reply
The response expected from the server.

acv-quit
The command that must be supplied to terminate the connection with the server (e.g., QUIT for SMTP, BYE for FTP—not required for HTTP). Please note that the CLI will add a Line Feed (\n) to this command before sending it to the server.

read-till-index
The number of bytes of the response to read (from 2 to 255) before starting an acv-reply.

check-port
Use this parameter to set an alternate port for application checks. Specify a number between 1 and 65535.

Restrictions

None.

Example

To set the load-balancing group-options for the server group ‘service2’ to ping every 5 seconds:

```
xp(config)# load-balance set group-options service2 ping-int 5
```

load-balance set group-conn-thresh

Purpose

Sets the connection threshold for each server in this group.

Format

load-balance set group-conn-thresh <*string*> **limit** <*num*>

Mode

Configure

Description

The **load-balance set group-options** command allows you to set a limit on how many connections will be supported for a load balancing server group. This number will be the maximum number of connections allowed for each server in the group. This group must already be created with the **load-balance create group-name** command.

Parameters

group-conn-thresh <*string*>
The name of the group of load balancing servers.

limit <*num*>
Specifies the number of connections that are supported by the server group. Specify any number between 1 and 65535.

Restrictions

None.

Example

To set the maximum number of connections to 50000 connections for server group ‘service2’:

```
xp(config)# load-balance set group-conn-thresh service2 limit 50000
```

load-balance set hash-variant

Purpose

Sets the hash variant for calculating the load-balancing mappings index.

Format

load-balance set hash-variant <*value*>

Mode

Configure

Description

The **load-balance set hash-variant** command sets the hash variant that is used to calculate the load-balancing mappings index. You will only need to set this variant if the **load-balance show hash-stats** command output shows extremely uneven distribution of hash table entries.

Parameters

hash-variant <*value*>
Specifies the hash variant. Specify 0, 1, or 2. The default value is 0.

Restrictions

None.

Example

To set the hash variant to 1:

```
xp(config)# load-balance set hash-variant 1
```

load-balance set mappings-age-timer

Purpose

Specifies the timeout for sessions between hosts and load-balancing servers.

Format

load-balance set mappings-age-timer <*timer*>

Mode

Configure

Description

A mapping between a host (source) and a load-balancing server (destination) times out after a period of non-use. The **load-balance set mappings-age-timer** command allows you to set the timeout for the mappings. The default is 3 minutes.

Parameters

mappings-age-timer <timer>
The number of minutes before a source-destination mapping times out. Specify a value between 3-4320.

Restrictions

None.

Example

To set the timeout for load-balancing mappings to 720 minutes (12 hours):

```
xp(config)# load-balance set mappings-age-timer 720
```

load-balance set policy-for-group

Purpose

Specifies the policy for distributing workload on load-balancing servers.

Format

load-balance set policy-for-group <*group name*> **policy** <*policy*>

Mode

Configure

Description

The **load-balance set policy-for-group** command allows you to specify how the X-Pedition selects the server that will service a new session. The default policy for distributing workload among the load balancing servers is "round-robin," where the X-Pedition selects the server on a rotating basis.

Parameters

policy-for-group <*group name*>
The name of this group of load balancing servers.

policy <*policy*>
One of the following keywords:

round-robin
The servers are selected sequentially (round-robin), without regard to the load on individual servers. This is the default policy.

weighted-round-robin
This policy is a variation of the round-robin policy. The X-Pedition still selects servers in turn, but during its turn, each server takes on a number of session connections according to its assigned weight. For example, if 'server1' is assigned a weight of 1000 and 'server2' is assigned a weight of 10, then server1 will be assigned 1000 sessions during its turn and server2 will be assigned 10 sessions during its turn. If you specify this policy, then you should assign different weights to each server in the group with the **load-balance add host-to-group** or the **load-balance add host-to-vip-range** command.

least-loaded
The server with the fewest number of sessions bound to it is selected to service the new session.

Restrictions

None.

Example

To set the load-balancing policy for the server group ‘service2’ to ‘weighted round robin’:

```
xp(config)# load-balance set policy-for-group service2 policy weighted-round-robin
```

load-balance set server-status

Purpose

Sets the status of a load balancing server.

Format

load-balance set server-status server-ip <*ipaddr/range*> **server-port** <*port number*> **group-name** <*group name*> **status up|down**

Mode

Enable

Description

The **load-balance set server-status** command allows you to set the status of a load balancing server. When the status of a server is set to "down," no *new* sessions are directed to that server. Current sessions on the server are not affected. This command can be used when server content needs to be updated or to bring one or more backup servers online during peak usage times.

Parameters

server-ip <*ipaddr/range*>
IP address of the server whose status is to be set.

Note: Do not use an IP address for load-balancing that is already configured for VRRP.

server-port <*port number*>
Port number of the server whose status is to be set.

group-name <*group name*>
Group name to which this server belongs.

status up|down
Sets the server status to up or down. Setting a server's status to down will cause new sessions *not* to be directed to the server.

Restrictions

Do not use an IP address for load-balancing that is already configured for VRRP.

Example

To set the status for the server 10.10.1.2 to ‘down’:

```
xp# load-balance set server-status server-ip 10.10.1.2 server-port 80 group-name service2 status
down
```

load-balance set server-options

Purpose

Sets options for a destination server.

Format

load-balance set server-options <*string*> **[port** <*num*>**] [ping-int** <*num*>**] [ping-tries** <*num*>**] [app-int** <*num*>**] [app-tries** <*num*>**] [acv-command** <*string*> **acv-reply** <*string*>**] [acv-quit** <*string*>**] [read-till-index** <*num*>**] [check-port** <*port number*>**]**

Mode

Configure

Description

The **load-balance set server-options** command allows you to set various parameters for a load balancing destination server.

Parameters

server-options <*string*>
The name of the destination server.

port <*num*>
Use this parameter to select the port running the application on the destination server.

ping-int <*num*>
Use this parameter to set the ping interval (seconds) for servers in this group. Specify any value between 5 and 3600. The default value is 5.

ping-tries <*num*>
Use this parameter to set the number of ping retries before marking the server down. Specify any value between 4 and 255. The default value is 4.

app-int <*num*>
Use this parameter to set the interval (seconds) between application checks. Specify any value between 5 and 3600. The default value is 15.

app-tries <*num*>
Use this parameter to set the number of retries before marking the application down. Specify any value between 4 and 255. The default value is 4.

acv-command <*string*>
The command or series of commands to pass to the server to request the desired file or

response. For example, an HTTP command might be: “GET /index.html HTTP/1.1\r\nHost: <server IP>\r\n\r\n”

Note: You must explicitly add any carriage returns (\r) or line feeds (\n) to the end of any command you pass to the server.

acv-reply <*string*>
The response expected from the server.

acv-quit <*string*>
The command that must be supplied to terminate the connection with the server (e.g., QUIT for SMTP, BYE for FTP—not required for HTTP). Please note that the CLI will add a Line Feed (\n) to this command before sending it to the server.

read-till-index
The number of bytes of the response to read (from 2 to 255) before starting an acv-reply.

check-port <*port number*>
Use this parameter to set an alternate port for application checks. Specify a number between 1 and 65535.

Restrictions

None.

Example

To set the load-balancing server-options for the destination server ‘server2’ to ping every 5 seconds:

```
xp(config)# load-balance set server-options server2 ping-int 5
```

load-balance set vpn-dest-port

Purpose

Sets the destination port for VPNs.

Format

load-balance set vpn-dest-port <*num*>

Mode

Configure

Description

The **load-balance set vpn-dest-port** command allows you to set the destination port number for load balanced VPNs.

Parameters

vpn-dest-port <*num*>
Specifies the destination port number. Specify any number between 1 and 65535. Default is 500.

Restrictions

Do not specify port 20, since this is the number designated for the FTP data port.

Example

To set the destination port to port 5000:

```
xp(config)# load-balance set vpn-dest-port 5000
```

load-balance show acv-options

Purpose

Displays load balance application content verification (acv) options.

Format

load-balance show acv-options [group-name <*string*>**] [destination-host-ip** <*ipaddr*>**] [destination-host-port** <*num*>**]**

Mode

Enable

Description

The **load-balance show acv-options** command allows you to display load balancing acv options.

Parameters

group-name <*string*>
Use this parameter to show acv-options of the servers belonging to this group.

destination-host-ip <*ipaddr*>
Use this parameter to show acv-options of the servers that are a part of the group with this Virtual IP.

destination-host-port <*num*>
Use this parameter to show acv-options of servers that are a part of the group with this Virtual port. Specify any number between 1 and 65535.

Restrictions

None.

load-balance show hash-stats

Purpose

Displays load balancing hashing statistics.

Format

load-balance show hash-stats

Mode

Enable

Description

The **load-balance show hash-stats** command allows you to display load balancing hash statistics.

Parameters

None.

Restrictions

None.

Example

To display hash statistics:

```
xp# load-balance show hash-stats
Total Mappings: 4502

Top 10 Hash Depths:

+---------+----------------+------------------------+
| Index |  Hash Depth  | Hash Depth Occurrence |
+---------+----------------+------------------------+
|1      |0             |11882                   |
|2      |1             |4226                    |
|3      |2             |138                     |
+---------+----------------+------------------------+

Top 10 Hash Depth Occurrences:

+---------+------------------------+----------------+
| Index |  Hash Depth Occurrence |  Hash Depth  |
+---------+------------------------+----------------+
|1      |11882                   |0             |
|2      |4226                    |1             |
|3      |138                     |2             |
+---------+------------------------+----------------+
```

load-balance show source-mappings

Purpose

Displays load balancing source-destination bindings.

Format

load-balance show source-mappings client-ip <*ipaddr*> **virtual-ip** <*ipaddr*> **virtual-port** <*port number*> **destination-host-ip** <*ipaddr*>

Mode

Enable

Description

The **load-balance show source-mappings** command allows you to display load balancing source-destination bindings.

Parameters

client-ip <*ipaddr*>
IP address of client whose mappings are to be shown.

virtual-ip <*ipaddr*>
Virtual IP address whose mappings are to be shown.

virtual-port <*port number*>
Virtual port number whose mappings are to be shown.

destination-host-ip <*ipaddr*>
IP address of the destination server whose mappings are to be shown.

Restrictions

None.

Example

To display source-destination bindings:

```
xp# load-balance show source-mappings

Current Mappings:

FC: Flow Count
AC: Age Count
SPort: Source Port
VPort: Virtual Port
DPort: Destination Port

+---------------+-----+---------------+-----+---------------+-----+----+----+
| Source Address |Sport| Virtual IP |VPort| Dst. Address |DPort| FC | AC |
+---------------+-----+---------------+-----+---------------+-----+----+----+
|70.1.0.71 |1024 |50.1.1.18 |80 |52.1.1.73 |80 |2 |0 |
|70.1.0.71 |1025 |50.1.1.17 |80 |52.1.1.71 |80 |2 |0 |
|70.1.0.72 |1026 |50.1.1.17 |80 |52.1.1.72 |80 |2 |0 |
|70.1.0.72 |1027 |50.1.1.18 |80 |52.1.1.74 |80 |2 |0 |

4 source mapping(s) displayed.
```

load-balance show statistics

Purpose

Displays load balancing statistics.

Format

load-balance show statistics group-name <*group name*> **virtual-ip** <*ipaddr*> **virtual-port** <*port number*>

Mode

Enable

Description

The **load-balance show statistics** command allows you to display load balancing statistics.

Parameters

group-name <*group name*>
Name of the group whose statistics are to be shown.

virtual-ip <*ipaddr*>
Virtual IP address whose statistics are to be shown.

virtual-port <*port number*>
Virtual port number whose statistics are to be shown.

Restrictions

None.

Example

To display load balance statistics:

```
xp# load-balance show statistics

Load Balancing Packets Dropped:
      No Such Virtual-IP Packet drop count: 73
      TTL expired Packet drop count: 0

Load Balance Group Statistics:

      Group Name: telnet Virtual-IP: 50.1.1.17 Virtual-Port: 23
            No destination selected Packet drop count            : 0
            Memory Allocation error Packet drop count            : 0
            No forward route found Packet drop count             : 0
            Number of Packets forwarded                          : 23437
            Channel not Load Balancing compliant Packet drop count : 0
            No hosts in group Packet drop count                  : 0
            Client in Access List Packet drop count              : 2

      Group Name: http Virtual-IP: 50.1.1.17 Virtual-Port: 80
            No destination selected Packet drop count            : 2
            Memory Allocation error Packet drop count            : 0
            No forward route found Packet drop count             : 0
            Number of Packets forwarded                          : 34429
            Channel not Load Balancing compliant Packet drop count : 0
            No hosts in group Packet drop count                  : 0
            Client in Access List Packet drop count              : 1

Statistics of 2 groups shown.
```

load-balance show virtual-hosts

Purpose

Displays hosts in a load balancing group.

Format

load-balance show virtual-hosts group-name <*group name*> **virtual-ip** <*ipaddr*> **virtual-port** <*port number*>

Mode

Enable

Description

The **load-balance show virtual-hosts** command allows you to display the hosts in a load balancing group.

Parameters

group-name <*group name*>
The load balancing group that is to be shown.

virtual-ip <*ipaddr*>
IP address of the group that is to be shown.

virtual-port <*port number*>
Port number of the group that is to be shown.

Restrictions

None.

Example

To display load balance groups:

```
xp# load-balance show virtual-hosts

Load Balanced Groups:

Flow Mode Count: 0

OS: Operational state of server
AS: Admin state of server

+---------------+---------------+------+-------------+----------+------------+
|   Group Name   |   Virtual IP   | Port | Hosts Added | Hosts Up | Next Index |
+---------------+---------------+------+-------------+----------+------------+
|telnet         |50.1.1.17      |23    |2            |2         |0           |
+---------------+---------------+------+-------------+----------+------------+
+-------+---------------+------+----------------+----+----+------------+
| Index |    Host IP    | Port |  Client Count  | OS | AS | Load Count |
+-------+---------------+------+----------------+----+----+------------+
|0      |52.1.1.73      |23    |0               |Up  |Up  |0           |
|1      |52.1.1.74      |23    |0               |Up  |Up  |0           |
+-------+---------------+------+----------------+----+----+------------+


+---------------+---------------+------+-------------+----------+------------+
|   Group Name   |   Virtual IP   | Port | Hosts Added | Hosts Up | Next Index |
+---------------+---------------+------+-------------+----------+------------+
|http           |50.1.1.17      |80    |2            |2         |0           |
+---------------+---------------+------+-------------+----------+------------+
+-------+---------------+------+----------------+----+----+------------+
| Index |    Host IP    | Port |  Client Count  | OS | AS | Load Count |
+-------+---------------+------+----------------+----+----+------------+
|0      |52.1.1.71      |80    |0               |Up  |Up  |0           |
|1      |52.1.1.72      |80    |0               |Up  |Up  |0           |
+-------+---------------+------+----------------+----+----+------------+
```

Chapter 37

logout Command

The **logout** command ends the CLI session.

Format

logout

Mode

All modes

Description

The **logout** command ends your CLI session. If you have uncommitted changes in the scratchpad, a message warns you that the changes are not saved and gives you an opportunity to cancel the logout and save the changes.

Parameters

None.

Restrictions

None.

Chapter 38

multicast Commands

The **multicast** commands let you display information about IP multicast interfaces.

Command Summary

Table 31 lists the **multicast** commands. The sections following the table describe the command syntax.

Table 31. multicast commands

multicast show interface [<*ipAddr*>|<*hostname*>**]**
multicast show mfc
multicast show mroutes [child <*IPaddr*>**] [group** <*IPaddr*>**] [parent** <*IPaddr*>**]**
multicast show sg-counts
multicast show vif

multicast show interface

Purpose

Display information about IP multicast interfaces.

Format

multicast show interface [<*ipAddr*>|<*hostname*>**]**

Mode

Enable

Description

The **multicast show interface** command displays interfaces that are running IGMP or DVMRP.

Note: This command is a superset of the **dvmrp show interface** and **igmp show interface** commands.

Parameters

<*ipAddr*>|<*hostname*> IP address or hostname of the interface.

Restrictions

None.

Examples

To display IP multicast information about interface 10.50.89.90:

```
xp# multicast show interface 10.50.89.90
```

The following example shows a larger listing.

```
xp# multicast show interface

Address: 172.1.1.10 Subnet: 172.1.1/24 Met: 1 Thr: 1
Name: mls15 State: Up Querier Leaf Igmp Dvmrp

Address: 207.135.89.64 Subnet: 207.135.89.0/25 Met: 1 Thr: 1
Name: company State: Up Querier Leaf Igmp Dvmrp
Groups: 224.0.1.12
224.1.127.255
224.0.1.24
224.2.127.253
224.2.127.254

Address: 10.135.89.10 Subnet: 10.135.89.0/25 Met: 1 Thr: 1
Name: test State: Up Querier Igmp Dvmrp
Peer: 10.135.89.67 Flags: 0xe Version: 3.255

Address: 190.1.0.1 Subnet: 190.1/16 Met: 1 Thr: 1
Name: rip State: Dis

Address: 207.135.122.11 Subnet: 207.135.122.8/29 Met: 1 Thr: 1
Name: mbone State: Up Igmp Dvmrp
Peer: 207.135.122.10 Flags: 0xe Version: 3.255
Groups: 224.0.1.11
224.0.1.12
224.2.127.254
239.255.255.255
224.2.127.253

Address: 10.40.1.10 Subnet: 10.40.1/24 Met: 1 Thr: 1
Name: downstream State: Up Dvmrp
Peer: 10.40.1.1 Flags: 0xf Version: 3.255

Address: 10.100.1.1 Subnet: 10.100.1/24 Met: 1 Thr: 1
Name: dan State: Dn Dvmrp
```

multicast show mfc

Purpose

Displays the multicast forwarding cache information, including the exit port (s) for each (S,G) flow.

Format

multicast show mfc

Mode

Enable

Description

PIM uses the Multicast Forwarding Cache to store forwarding information for each (S,G) flow. The **multicast show mfc** command displays a table containing this information.

Parameters

None.

Example

To display the multicast forwarding cache information, enter the following:

```
xp# multicast show mfc

Source Address       Group Address        Incoming I/f     Outgoing I/f     Exit Ports
-------------------- -------------------- ---------------- ---------------- ------------
10.32.32.5           227.1.1.64           server2          to_rp            et.3.4
                                                           XP1              et.3.1
```

multicast show mroutes

Purpose

Display the IP multicast routing table.

Format

multicast show mroutes [child <*IPaddr*>**] [group** <*ipaddr*>**] [parent** <*IPaddr*>**]**

Mode

Enable

Description

The **multicast show mroutes** command displays the IP multicast routing table entry for the specified multicast group address.

This command lists all the multicast distribution trees, showing the parent interface (from where the traffic is coming), and the children distribution interfaces (to which the traffic is being forwarded). It would also show any cache information available either in hardware forwarding mechanism or in the main processor (for software based forwarding).

Note: The cache information can be timed out when not enough traffic is present, but multicast routes can still be present. Cache information is presented in number of flows (Layer 4 sessions). Multicast routes stay at least for 5 minutes, while the hardware forwarding mechanism can time out a flow faster.
Any pruning information if present is also shown.

The search can always be narrowed by looking at a particular group, and/or looking at a particular parent interface, and/or looking at a particular child interface.
Multicast routes are not the same as DVMRP routes.

Parameters

child <*ipaddr*>	Address of a child interface.
group <*ipaddr*>	Address of a multicast group.
parent <*ipaddr*>	Address of a parent interface.

Restrictions

None.

Examples

To display the IP multicast route entry for the group 225.0.0.10:

```
xp# multicast show mroutes group 225.0.0.10
```

Here is a fuller example of the output from this command.

```
xp# multicast show mroutes
Network: 130.207.8/24 Group: 224.2.1.1 Age: 99s
Parent: mbone Child: test
downstream
Source: 130.207.8.82 Pkts: 383 Flows: 1

Network: 131.120.63/24 Group: 224.2.1.1 Age: 63s
Parent: mbone Pruned Child: test Pruned
downstream Pruned
Source: 131.120.63.33 Pkts: 0 Flows: 0

Network: 147.6.65.0/25 Group: 224.2.2.1 Age: 48s
Parent: mbone Pruned Child: test Pruned
downstream Pruned
Source: 147.6.65.38 Pkts: 0 Flows: 0
```

multicast show sg-counts

Purpose

Displays the packet and byte counts for each multicast forwarding cache entry.

Format

multicast show sg-counts

Mode

Enable.

Description

PIM uses the Multicast Forwarding Cache to store forwarding information for each (S,G) flow. The **multicast show sg-counts** command displays a table byte counts and packet counts.

Parameters

None.

Example

To display table byte and packet counts for multicast forwarding cache entries, enter the following:

```
xp# multicast show sg-counts

Source Address     Group Address      Packet Cnt   Byte Count
-------------------------------------------------------------
10.32.32.5         227.1.1.64         26920        41322130
```

multicast show vif

Purpose

Displays the virtual interface information for all multicast capable interfaces.

Format

multicast show vif

Mode

Enable

Description

Displays information about the virtual interface configurations used in multicast.

Parameters

None.

Example

To display the virtual interface configuration information, enter the following:

```
xp# multicast show vif

F ->      0x01   -  Vif is tunnel end-point
          0x02   -  Tunnel is using IP source routing
          0x04   -  Vif is used for register encap/decap
          0x10   -  Vif owner is DVMRP
          0x20   -  Vif owner is PIM

Vif     Interface       F       Local Addr          Remote Addr         Portmask
 ---    -----------     -       ---------------     ----------------    ------------
   0    register_vif    24      127.0.0.3           127.0.0.0
   1    XP1             20      172.10.1.117        0.0.0.0
   2    to_rp           20      172.10.1.113        0.0.0.0
   3    server2         20      10.32.32.1          0.0.0.0
```

Note: The “F” above represents flags.

Chapter 39

nat Commands

The **nat** commands allow you to define Network Address Translation (NAT) bindings for local (inside) and global (outside) network addresses.

Command Summary

Table 32 lists the **nat** commands. The sections following the table describe the command syntax.

Table 32. nat commands

nat clear-err-stats out-of-globals| port-mode
nat create dynamic local-acl-pool <*local-acl*> **global-pool** <*ip-addr/ip-addr-range/ip-addr-list*> **[matches-interface** <*interface*>**] [enable-ip-overload]**
nat create static protocol ip|tcp|udp local-ip <*local-ip-addr/address range*> **global-ip** <*global-ip-addr/address range*> **[local-port** <*tcp/udp-local-port*>**|any] [global-port** <*tcp/udp-global-port*>**|any]**
nat flush-dynamic-binding all| pool-specified [**local-acl-pool** <*local-acl*>] [**global-pool** <*ip-addr/ip-addr-range*>**]| type-specified| owner-specified [dns| ftp-control| ftp-data]**
nat set dns-name-extension-error on|off
nat set dns-session-timeout <*num*>
nat set dynamic-binding-timeout <*minutes*>**|disable**
nat set ftp-control-port <*port number*>
nat set ftp-session-timeout <*minutes*>

Table 32. nat commands (Continued)

nat set interface <*name*> **inside**|**outside**
nat set secure-plus on| **off**
nat show [translations] [timeouts] [statistics]

nat clear-err-stats

Purpose

Clears NAT error statistics.

Format

nat clear-err-stats out-of-globals| port-mode

Mode

Enable

Description

The **nat clear-err-stats** command allows you to clear specific NAT error statistics such as out-of-globals messages in the case of dynamic bindings and port misconfiguration.

Parameters

out-of-globals	Clears error statistics during dynamic binding in the case where there are no more global IP addresses in the global address pool.
port-mode	Clears error statistics that occur because of port misconfigurations. Such cases are where the port is set to either destination-based forwarding or host-flow based forwarding.

Restrictions

None

Example

To clear all out-of-global error statistics:

```
xp(config)# nat clear-err-stats out-of-globals
```

nat create dynamic

Purpose

Defines local and global IP address pools for dynamic address binding.

Format

nat create dynamic local-acl-pool <*local-acl*> **global-pool** <*ip-addr/ip-addr-range/ip-addr-list*> **[matches-interface** <*interface*>**] [enable-ip-overload]**

Mode

Configure

Description

The **nat create dynamic** command lets you specify the local-acl pool and global IP address pool that are to be used for dynamic address binding. With dynamic address translation, IP address bindings last only until the data flow ages out or the dynamic binding is manually deleted. Global IP addresses defined for dynamic translation are reassigned whenever they become free. The local address pool for dynamic bindings are defined via an ACL profile, while the global address pool must be specified as a single IP address, an address range, an IP address and mask, or an IP list. You can also specify multiple global pools for the same local-acl pool, if you have more than one connection to the Internet on different interfaces.

Parameters

local-acl-pool <*local-acl*>

The ACL that corresponds to the local IP address pool. The ACL may contain either **permit** or **deny** keywords. Note that only the source IP address information in the ACL is used; other ACL parameters are ignored. Used in the case of address translation from an inside private to an outside public network.

global-pool <*ip-addr/ip-addr-range/ip-addr-list*>

The global address pool, defined in one of the following ways:

A single IP address in the form a.b.c.d

An IP address range in the form 10.10.1.1-10.10.1.50

IP address and mask in the form 1.2.0.0/255.255.0.0 or 1.2.3.0/16

A list of IP addresses, separated by spaces and enclosed in quotation marks.

Used in the case of address translation from an inside private to an outside public network.

Note: Do not specify more than 64K global addresses.

matches-interface <*interface*>
Specifies the interface to use for multiple global pools. Used in the case of address translation from an inside private to an outside public network.

enable-ip-overload
Enables Port Address Translation (PAT) if no global addresses are available from the pool. This allows many local addresses to be bound to a single global address using port numbers 1024 through 4999 (port numbers are not configurable). With PAT, multiple IP addresses can map to a single IP address with multiple numbers. Used in the case of address translation from an inside private to an outside public network.

Note: Protocols like ICMP do not work with the **enable-ip-overload** option. Thus, the **ping** command will not work if this option is used.

Restrictions

None.

Examples

To configure address pools for dynamic address bindings, first configure the ACL that corresponds to the local IP address pool. In the following example, the ACL ‘lcl’ corresponds to IP addresses from 10.1.1.1 to 10.1.1.254:

```
xp(config)# acl lcl permit ip 10.1.1.0/24
```

Then, specify this ACL for the local IP address pool for dynamic address bindings with global addresses 136.1.1.1 to 136.1.1.254:

```
xp(config)# nat create dynamic local-acl-pool lcl global-pool 136.1.1.0/24
```

The following examples show the use of Port Address Translation, where the global pool consists of only two specified IP addresses. In the following example, the ACL ‘lcl’ corresponds to IP addresses from 10.1.1.1 to 10.1.1.254:

```
xp(config)# acl lcl permit ip 10.1.1.0/24
```

Then, specify this ACL for the local IP address pool for dynamic address bindings with global addresses 136.1.1.1 and 136.1.1.2 with Port Address Translation enabled:

```
xp(config)# nat create dynamic local-acl-pool lcl global-pool 136.1.1.1-136.1.1.2 enable-ip-overload
```

Port numbers 1024 through 4999 can be used for global addresses 136.1.1.1 and 136.1.1.2, so you can have a maximum of about 4000 bindings per global address.

nat create static

Purpose

Defines one-to-one binding between a local address and global address.

Format

nat create static protocol ip|tcp|udp local-ip <*local-ip-addr/address range*> **global-ip** <*global-ip-addr/address range*> **[local-port** <*tcp/udp-local-port*>**|any] [global-port** <*tcp/udp-global-port*>**|any]**

Mode

Configure

Description

The **nat create static** command lets you define fixed address translation from the local network to the global network. The binding of the local to the global address does not expire until this command is negated. If the protocol used is TCP or UDP, you can also specify port address translation (PAT).

Parameters

ip|tcp|udp
Specifies either only IP address translation, IP and TCP port address translation, or IP and UDP port address translation.

local-ip <*local-ip-addr/address range*>

Either a single IP address, in the form a.b.c.d, or an address range, in the form 10.10.1.1-10.10.1.50.

global-ip <*global-ip-addr/address range*>

Either a single IP address, in the form a.b.c.d, or an address range, in the form 10.10.1.1-10.10.1.50.

local-port <*tcp/udp-local-port*>**|any**

The local TCP or UDP port number. Specify a number between 1-65535, or **any** for no port translation. This parameter is only valid if you specified **tcp** or **udp**.

Note: The number of IP addresses in the local range should be equal to the number of IP addresses in the global range.

global-port <*tcp/udp-global-port*>|**any**

The global TCP or UDP port number. Specify a number between 1-65535, or **any** for no port translation. This parameter is only valid if you specified **tcp** or udp.

Restrictions

None.

Examples

To configure a static binding of a local and a global IP address:

```
xp(config)# nat create static protocol ip local-ip 10.1.1.13 global-ip 136.1.1.13
```

To configure a static binding of local and global IP address ranges:

```
xp(config)# nat create static protocol ip local-ip 10.1.1.1-10.1.1.50 global-ip 136.1.1.1-136.1.1.50
```

To configure a static binding of local and global IP and UDP port addresses:

```
xp(config)# nat create static local-ip 10.1.1.13 global-ip 136.1.1.13 local-port 18 global-port 36
protocol udp
```

nat flush-dynamic-binding

Purpose

Deletes dynamic NAT bindings.

Format

nat flush-dynamic-binding all| pool-specified [**local-acl-pool** <*local-acl*>] [**global-pool** <*ip-addr/ip-addr-range/ ip-addr-list*>]| **type-specified| owner-specified [dns| ftp-control| ftp-data]**

Mode

Enable

Description

The **nat flush-dynamic-binding** command deletes dynamic address bindings. You can delete the dynamic address bindings for specific address pools or delete all dynamic bindings.

Parameters

all
Deletes all NAT dynamic bindings.

pool-specified
Deletes NAT dynamic bindings based on local and global acl pools.

local-acl-pool <*local-acl*>
The ACL that corresponds to the local IP address pool.

global-pool <*ip-addr/ip-addr-range*>
The global address pool, defined in one of the following ways:

A single IP address in the form a.b.c.d

An IP address range in the form 10.10.1.1-10.10.1.50

IP address and mask in the form 1.2.0.0/255.255.0.0 or 1.2.3.0/16

type-specified
Deletes NAT dynamic bindings based on the type of dynamic binding.

owner-specified
Deletes NAT dynamic bindings based on the type of application utilizing the bindings.

dns
Deletes NAT dynamic bindings created by DNS (domain name server).

ftp-control
Deletes NAT dynamic bindings created by FTP control connection.

ftp-data
Deletes NAT dynamic bindings created by FTP data connection.

Restrictions

None.

Examples

To delete dynamic address bindings for the local address pool that corresponds to the ACL ‘lcl’ and the global address pool that corresponds to 136.1.1.1-136.1.1.254:

```
xp# nat flush-dynamic-binding pool-specified local-acl-pool lcl global-pool 136.1.1.0/24
```

To delete all dynamic address bindings:

```
xp# nat flush-dynamic-binding all
```

nat set dns-name-extension-error

Purpose

Enable or disable the error message associated with DNS name extensions.

Format

nat set dns-name-extension-error on|off

Mode

Configure.

Description

The **nat set dns-name-extension-error** command allows you to enable or disable the router's ability to display the DNS name extension error message.

Parameters

on|off Select **on** or **off** to enable or disable this error message. If you negate a command that specifies the option to be off (i.e., **nat set dns-name-extension-error off**), the command is enabled automatically.

Restrictions

None.

Example

To *enable* the router's ability to display the DNS name extension error message, enter the following:

```
xp(config)# nat set dns-name-extension-error on
```

To *disable* the router's ability to display the DNS name extension error message, enter the following:

```
xp(config)# nat set dns-name-extension-error off
```

nat set dns-session-timeout

Purpose

Specifies the timeout for the DNS session.

Format

nat set dns-session-timeout <*num*>

Mode

Configure.

Description

The **nat set dns-session-timeout** command sets the timeout for DNS application-specific sessions.

The default DNS session timeout is **30** minutes.

Parameters

<*num*>	The timeout for the DNS session, in minutes. Specify a value between 3-2880. Default is 30 minutes.

Restrictions

None.

Example

To set the DNS session timeout to 60 minutes:

```
xp(config)# nat set dns-session-timeout 60
```

nat set dynamic-binding-timeout

Purpose

Sets the timeout for dynamic NAT binding.

Format

nat set dynamic-binding-timeout <*minutes*>|**disable**

Mode

Configure

Description

Dynamic address bindings time out after a period of non-use. The **nat set dynamic-binding-timeout** command lets you set the timeout for dynamic address bindings. The default is 1440 minutes (24 hours).

Parameters

<*minutes*>	The number of minutes before an dynamic address binding times out. Specify a value between 3-2880.
disable	Disables timeout of dynamic address bindings.

Restrictions

None

Example

To set the timeout for dynamic address bindings to 3 minutes:

```
xp(config)# nat set dynamic-binding-timeout 3
```

To disable timeout of dynamic address bindings:

```
xp(config)# nat set dynamic-binding-timeout disable
```

nat set ftp-control-port

Purpose

Specifies the port for FTP control.

Format

nat set ftp-control-port <*port number*>

Mode

Configure

Description

File Transfer Protocol (FTP) packets require special handling with NAT, because IP address information is contained within the FTP packet data. You can use the **nat set ftp-control-port** command to specify the port number that is used for FTP control.

The default port for FTP control is port **21**.

The X-Pedition's current ACL/NAT implementation does not make provisions for running standard or PASV FTP sessions across a translated interface when only ports 20 (FTP data port) and 21 (FTP control port) are open for communication. Because FTP will use other higher-numbered ports to establish TCP sessions, FTP sessions established across a NAT-translated interface may hang if these other TCP ports are not open for communication. In order to allow FTP to establish a TCP session on higher-numbered ports, the NAT-associated ACL must be set up to allow incoming traffic from any port. When running this configuration, it is suggested that NAT secure-plus is enabled (**nat set secure-plus on**) in order to increase security and prevent private address leaks. For more information, please reference RFC 1579 (“Firewall-Friendly FTP”).

Parameters

<*port number*>
Specifies the port number used for FTP control. Specify a value between 1 and 65535.

Restrictions

None.

Example

To set the FTP control port to 100:

```
xp(config)# nat set ftp-control-port 100
```

nat set ftp-session-timeout

Purpose

Specifies the timeout for the FTP session.

Format

nat set ftp-session-timeout <*minutes*>

Mode

Configure

Description

The **nat set ftp-session-timeout** command sets the timeout for the FTP session.

The default FTP session timeout is **30** minutes.

Parameters

<*minutes*> The timeout for the FTP session. Specify a value between 3-2880.

Restrictions

None.

Example

To set the FTP session timeout to 60 minutes:

```
xp(config)# nat set ftp-session-timeout 60
```

nat set interface

Purpose

Defines an interface as inside or outside for NAT address translation.

Format

nat set interface <*name*> **inside|outside**

Mode

Configure

Description

The **nat set interface** command allows you to define an interface as inside or outside. When NAT is enabled using the **nat create static** or **nat create dynamic** command, address translation is applied only to packets that arrive on these interfaces.

Parameters

<*name*>
Is the name of the interface to which address translation will apply.

Note: The X-Pedition will display interface names up to 32 characters in length.

inside|outside
Specifies the interface(s) as inside or outside.

Restrictions

None.

Examples

To create the interface ‘10-net’ and define it as an inside interface for NAT:

```
xp(config)# interface create ip 10-net address-netmask 10.1.1.1/24 port et.2.1
xp(config)# nat set interface 10-net inside
```

To create the interface ‘192-net’ and define it as an outside interface for NAT:

```
xp(config)# interface create ip 192-net address-netmask 192.50.20.1/24 port et.2.2
xp(config)# nat set interface 192-net outside
```

nat set secure-plus

Purpose

Block IP addresses defined as *inside* addresses from ever appearing on an *outside* interface.

Format

nat set secure-plus on|off

Mode

Configure

Description

The **nat set secure-plus** command forces all flows from the *inside* network or the *outside* network to go through network address translation. Packets that would otherwise bypass NAT and transmit untranslated are dropped.

Parameters

on|off
Specify on to enable secure-plus feature. Specify off to disable secure-plus feature.

Restrictions

None.

nat show

Purpose

Displays NAT information.

Format

nat show [translations <*type*>**] [timeouts] [statistics]**

Mode

Enable

Description

The **nat show** command allows you to display NAT address translations, timeouts, and statistics.

Parameters

translations <*type*>
Displays NAT translations. Specify one of the following keywords:

all
Shows all translations.

type static|dynamic|overloaded-dynamic
Shows static, dynamic, or IP overloaded dynamic translations.

owner dns|ftp-control|ftp-data
Shows dynamic translation created by dns, overloaded dynamic ftp control connection translations, or overloaded dynamic ftp data connection translations.

local-filter-in <*local-ip-addr*>
Shows translations of the specified local IP address. The IP address must be in the form a.b.c.d.

global-filter-in <*global-ip-addr*>
Shows translations of the specified global IP address. The IP address must be in the form a.b.c.d.

timeouts
Displays the current set of timeouts.

statistics
Displays NAT statistics.

verbose
Displays NAT translations in greater detail.

Restrictions

None.

Examples

To display active NAT translations:

```
xp# nat show translations all

Proto Local/Inside          Global/Outside IP     Type           No. of flows
----- --------------------- --------------------- -------------- ------------
TCP   15.15.15.15:1896      100.1.1.1:1026        Dyn. ovr.      2
TCP   15.15.15.15:1897      100.1.1.1:1028        Dyn. ovr.      0
TCP   15.15.15.15:1894      100.1.1.1:1024        Dyn. ovr.      2
TCP   15.15.15.15:1895      100.1.1.1:1025        Dyn. ovr.      2
TCP   15.15.15.15:1892      100.1.1.1:1027        Dyn. ovr.      0
IP    10.10.10.10:*         200.1.1.1:*           Dynamic        20
IP    4.4.4.4:*             202.1.1.1:*           Static         789
```

If there are many active NAT translations, you can filter the display by specifying **local-filter-in**, **global-filter-in**, or **type** parameters for the **nat show translations** command.

To display NAT timeouts:

```
xp# nat show timeouts

All values in minutes
Flow    FTP Sess.  DNS Sess.  Dyn. Sess.
------  ---------  ---------  ----------
2       30         30         1440
```

To display NAT statistics:

```
NAT current status
------------------
active

NAT secure-plus status
----------------------
inactive

Interface Information
---------------------
No. of Interfaces: 1
Interface: 20net, configured as nat: outside

STATIC Binding Information
--------------------------
No. of Static Bindings: 1

DYNAMIC Binding Information
---------------------------
No. of Dynamic Bindings: None

Local Acl pool  Max. globals  Globals used  Max. ports  Ports Used  Out of globals/ports
--------------  ------------  ------------  ----------  ----------  ----
local                1             0          3975          0        0
```

Chapter 40

negate Command

The **negate** command negates a command in the scratchpad or the active configuration.

Format

negate <*cmd-number*> **[scratchpad|active-config]**

Mode

Configure

Description

The **negate** command allows you to negate one or more commands by specifying the command number of the commands you want to negate. The command number for each command can be found using the Configure mode **show** command. You can negate commands from the active running system or non-committed commands from the scratchpad. By default, if you do not specify **active-config** or **scratchpad**, the command to negate is assumed to be in the **active-config**.

Parameters

<*cmd-number*>	The number of the command(s) you want to negate. Use the **show** command to display the command numbers.
active-config	Negate the specified command from the active running system.
scratchpad	Negate the specified non-committed command from the scratchpad.

Restrictions

The specified command number must represent a command that exists.

Examples

To negate command 23 from the active configuration:

```
xp# negate 23
```

To negate commands 3, 5, 6 and 7 from the scratchpad:

```
xp# negate 3,5-7 scratchpad
```

Chapter 41

netflow Commands

NetFlow data characterizes the movement of IP traffic on a network. NetFlow allows you to collect information about packets sent through the network and to use this data for detailed traffic analysis, network planning, network monitoring, usage-based billing, and for use by third party mediation vendors, network management tools, and billing companies. Combined with a network data analyzer, the NetFlow data you collect can help enterprise engineers, capacity planners, marketing groups, and network management better understand network traffic patterns and isolate areas in need of improvement. NetFlow traffic describes source and destination addresses, autonomous system numbers, port addresses, time of day, number of packets, total bytes, and type of service.

Note: In order to run NetFlow, you must *enable* SNMP.

Note: **Do not** run NetFlow and RMON Professional simultaneously.

Command Summary

Table 33 lists the **NetFlow** commands. The sections following the table describe the command syntax.

Table 33. NetFlow commands

netflow clear statistics
netflow enable
netflow set engine id <*engine id*> **type** <*engine type*>
netflow set flow-destination-port <*port-number*>
netflow set interval <*minutes*>
netflow set memory <*size*>
netflow set memory-threshold <*number*>
netflow set ports <*port list*> | **all-ports**
netflow set priority <*number*> **|low |medium |high**

netflow set collector <*collector_IPaddr*> **[flow-destination-port** <*number*>**]**
netflow show configuration| collector| statistics| status| historical max memory bytes| historical max memory time| all

netflow clear statistics

Purpose

Clear many of the NetFlow statistics kept by the X-Pedition. *Optional*.

Format

netflow clear statistics

Mode

Enable.

Description

The **netflow clear statistics** command will clear the NetFlow session statistics kept by the X-Pedition (available through *netflow show* on page 645). Clearing statistics will not affect the netflow process or the data packets sent from the XP to the collector.

Parameters

None.

Restrictions

NetFlow cannot monitor traffic exits in a multicast environment. If you want to collect statistics on traffic moving through your system, you must monitor the input port(s).

.

netflow enable

Purpose

Enable NetFlow agent. *Required*.

Format

netflow enable

Mode

Configure.

Description

The netflow enable command starts the NetFlow agent.

Parameters

None.

Restrictions

- You must configure at least one NetFlow collector before you can execute this command successfully. See *netflow set collector* on page 644.
- You must configure **netflow set ports** before you can monitor any ports.
- **Do not** run NetFlow and RMON simultaneously.
- In order to run NetFlow, you must *enable* SNMP.

netflow set engine

Purpose

Allows you to modify the engine identification and type sent with a NetFlow packet header.

Format

netflow set engine id <*engine id*> **type** <*engine type*>

Mode

Configure.

Description

The **netflow set engine** command allows the modification of the engine identification and type sent with a NetFlow packet header, affecting those NetFlow collectors that require specific engine values.

Parameters

<*engine id*>	The NetFlow engine ID (0-255 inclusive). By default, this value is 0.
<*engine type*>	The engine type (0-255 inclusive). By default, this value is 0.

Restrictions

None.

Example

To set the engine id to 4 and the engine type to 15, enter the following:

```
xp# netflow set engine id 4 type 15
```

netflow set flow-destination-port

Purpose

Sets the IP address of the default flow destination port of a specific NetFlow collector. *Optional*.

Format

netflow set flow-destination-port <*port-number*>

Mode

Configure.

Description

The **netflow set flow-destination-port** command sets the default flow destination port through which the X-Pedition will send its data. The default netflow flow-destination port number is 2055.

Note: If you change the default value for the flow-destination port, the new value must correspond with the port ID on the collector you will use.

Parameters

<*port-number*>
The port through which to send the NetFlow packets.

Restrictions

Hardware restrictions do not allow NetFlow to report a destination port for ICMP flows—the destination port is reported as 0.

netflow set interval

Purpose

Changes the default interval parameter for NetFlow. *Optional*.

Format

netflow set interval <*minutes*>

Mode

Configure.

Description

The **netflow set interval** command allows you to set the time interval at which all currently monitored flows will report updated information to the collector. NetFlow staggers the packet reporting throughout the entire period of the interval to reduce congestion and prevent packet loss. As flows *expire*, they will report to the collector regardless of the time interval specified.

Parameters

Interval <*minutes*>
The time in minutes (1-1440) for an interval. The default interval is 30 minutes.

Restrictions

None.

netflow set memory

Purpose

Changes the default memory settings for NetFlow. *Optional*.

Format

netflow set memory <*size*>

Mode

Configure.

Description

The **netflow set memory** command allows you to set the maximum amount of memory used by the NetFlow agent. This limits the maximum number of flows and buffered datagrams supported by the NetFlow agent.

Parameters

memory <*size*>
The amount of memory you will use for NetFlow operations (from 100k to 80% of the total allocated memory). By default, this value is 450k.

Restrictions

None.

netflow set memory-threshold

Purpose

Set the upper memory limit for NetFlow.

Format

netflow set memory-threshold <*number*>

Mode

Configure.

Description

The **netflow set memory** command allows you to set the upper memory limit to determine how much memory NetFlow may allocate to the X-Pedition (this includes memory allocated by any X-Pedition component).

Parameters

<*number*>
The percent (1-100) of memory netflow will allocate. The default value is 85%.

Restrictions

None.

Example

To set the upper memory limit to 75%, enter the following:

```
xp# netflow set memory-threshold 75
```

netflow set ports

Purpose

Configure the ports that will participate in the NetFlow accounting. *Required.*

Format

netflow set ports <*port list*>|**all-ports**

Mode

Configure.

Description

The **netflow set ports** command is used to identify which ports the NetFlow agent will monitor.

Parameters

ports <*port list*>
Specifies the ports participating in the NetFlow feature.

all-ports
Enables all ports.

Restrictions

None.

Example

```
xp# netflow set ports et.3.1-4
```

netflow set priority

Purpose

Changes the default priority levels of NetFlow tasks. *Optional*.

Format

netflow set priority <*number*> **| low | medium | high**

Mode

Configure.

Description

The **netflow set priority** command allows you to set the operational priority of NetFlow tasks. A higher priority indicates that the NetFlow tasks will have a chance to run more often. If you specify a numerical value, the lower the value of the number, the higher its task priority. Negate this command to set the NetFlow tasks priorities to their default values (between low and medium). This command is not commonly used.

Parameters

Priority <*number*>
Specify this parameter to set the task priority to a numerical value (from 50 to 250). The lower the value, the higher its priority.

low
Set the task priority to low.

medium
Set the task priority to medium.

high
Set the task priority to high.

Restrictions

None.

netflow set collector

Purpose

Sets the IP address of a NetFlow collector and allows you to configure NetFlow-related parameters for the collector. *Required*.

Format

netflow set collector <*collector_IPaddr*> **[flow-destination-port** <*number*>**]**

Mode

Configure.

Description

The **netflow set collector** command allows you to set NetFlow-related parameters—even override the default settings—on a *collector-by-collector* basis only (by using the IP address of a specific NetFlow collector). Although a collector may service multiple routers, you may not enable multiple collectors for the same router.

Parameters

collector <*collector_IPaddr*>
The IP address of a specific NetFlow collector. You can define only one NetFlow collector per **netflow set collector** command.

Note: Because NetFlow packets are UDP packets, packet delivery is a best effort delivery—the system will not attempt retries if delivery fails.

flow-destination-port <*number*>
The destination port number. The default NetFlow destination port number is 2055.

Note: If you change the default value for the flow-destination port, the new value must correspond with the port ID on the collector you will use.

Restrictions

Hardware restrictions do not allow NetFlow to report a destination port for ICMP flows—the destination port is reported as 0.

netflow show

Purpose

Displays all pertinent NetFlow agent data including configuration, collector, statistics, and status.

Format

netflow show configuration| collector| statistics| status| historical max memory bytes| historical max memory time| all

Mode

Enable.

Description

The **netflow show** command allows you to show the configuration, collector, statistics, and status, of the NetFlow agent.

Parameters

configuration

Show configured values:

- Whole flow table interval
- Ports included in NetFlow
- Maximum memory

collector

Show collector information:

- IP Address
- UDP port number

statistics

Show NetFlow statistics:

- Current flow count
- Number of reported flow updates
- Number of sent PDUs
- Memory consumed dropped PDUs

Note: NetFlow cannot monitor traffic exits in a multicast environment. If you want to collect statistics on traffic moving through your system, you must monitor the input port(s).

status

Show NetFlow status:

- Current NetFlow sequence number

State of the NetFlow agent (on/off)
Agent start time

status

Show NetFlow status:
Current NetFlow sequence number
State of the NetFlow agent (on/off)
Agent start time

historical max memory bytes

The maximum amount of memory in bytes which has been used in the past. This value may be more than is currently in use. This can be used to identify when memory usage is approaching the maximum configured for use by the NetFlow client. This value resets when you clear NetFlow statistics.

historical max memory time

Date and time when "Historical memory bytes" peaked.

all Show all of the above.

Restrictions

Hardware restrictions do not allow NetFlow to report a destination port for ICMP flows—the destination port is reported as 0.

Example

For detailed information about the contents of each field, please see the *Enterasys X-Pedition User Reference Manual*:

```
xp# netflow show all
 NetFlow Status:
   NetFlow is ENABLED
   NetFlow Started at              : 2002-08-01 10:11:08

 Netflow Default Configuration:
   NetFlow Version                 : 5
   NetFlow Engine ID               : 0
   NetFlow Engine Type             : 0
   Active Flows Polling Interval   : 1
   Threshold % Heap                : 85%
   Default Port                    : 2055
   NetFlow Task Priority           : 230

 Netflow Statistics

 Intervals
    Time of Last Reporting Interval:                  2002-08-01 10:23:58
    Time of Next Reporting Interval:                  2002-08-01 10:24:58

(Continued on next page....)
```

Continued from previous page:

```
Memory (Global)
  Total Memory Available for use:                 112298008
  Total Memory already in use:                     8531248
  Percent of total memory already in use:             6.71%
  Total Memory remaining for use:                 103766760

Memory (Netflow)
  Maximum amount of memory Netflow is allowed to use:    450000
  Memory in use by Netflow:                             40044
  Percent of total memory that is being used by Netflow:     0.04%
  Percent of memory allocated to Netflow that is in use:     8.90%

  Peak memory used by Netflow:                          40124
  Time of Peak memory usage:                2002-08-01 10:23:44
  Percent of Netflow memory used during the peak:            8.92%
  Percent of total memory used by Netflow during the peak:   0.04%
  Number of times Netflow failed to get requested memory:       0

Counters
  Current number of flows:                                  0
  Number of times netflow has sent reports:                 1
  Number of packets used to send reports:                   1
  Number of flows created in Netflow:                       2
  Number of flows deleted in Netflow:                       2
  Number of flows pending delete:                           0
  Number of flows not reported by Netflow (discarded):      0
  Number of reported records (flows):                       2
Ports Enabled for NetFlow:
            Port              Tracked       Tracked
 ifIndex    Name               In Flows      Out Flows     Monitored
 ------------------------------------------------------------------------------------------------
   0025    et.4.1             000000001     000000001     ON
   0026    et.4.2             000000001     000000001     ON
 Unknown Ports Flows :                      000000000

 Total Flows Count   :      000000002     000000002
 ------------------------------------------------------------------------------------------------
Number of Ports Being Monitored :2

NetFlow Collector
 IP Address:          10.136.136.210
 Accounting Port:     9999
```

Chapter 42

no Command

The **no** command removes a configuration command from the active configuration of the running system.

Format

no <*command-to-negate*>

Mode

Configure

Description

The **no** command allows you to negate a previously executed command. Following the keyword **no**, one can specify the command to negate in its entirety or use the wildcard character (*) to negate a group of commands. In addition to the **no** command, one can also use the **negate** command to negate a group of commands using the command number.

Parameters

<*command*> The CLI command you want to negate. You do not have to enter the entire command. You can use the wildcard character, *, to negate matching commands. For example, if you specify “no acl 100 *” then all commands starting with the words “acl 100” will be negated.

Restrictions

The command to negate must already be in the active configuration. You cannot negate a command that hasn’t been entered.

Examples

To negate the specified **arp add** command, enter the following. By negating this command, the system removes the ARP entry for *nfs2* from the ARP table.

```
xp# no arp add nfs2 macaddr 080020:13a09f exit-port et.3.1
```

To negate all commands starting with the word “acl”:

```
xp# no acl *
```

Chapter 43

ntp Commands

The **ntp** commands configure and display the characteristics of the NTP (Network Time Protocol) client.

Command Summary

Table 34 lists the **ntp** commands. The sections following the table describe the command syntax.

Table 34. ntp commands

ntp set server <*nameiplist*> **[interval** <*minutes*>**] [source** <*ipaddr*>**] [version** <*num*>**]**
ntp show all
ntp synchronize server <*host*>

ntp set server

Purpose

Specifies the NTP server against which the X-Pedition is to synchronize its clock.

Format

ntp set server <*nameiplist*> **[interval** <*minutes*>**] [source** <*ipaddr*>**] [version** <*num*>**]**

Mode

Configure

Description

The **ntp set server** command instructs the X-Pedition's NTP client to periodically synchronize its clock. By default, the X-Pedition specifies an NTPv3 client that sends a synchronization packet to the server every 60 minutes. This means the X-Pedition will attempt to set its own clock against the server once every hour. The synchronization interval as well as the NTP version number can be changed. To ensure that NTP has the correct time, you need to specify the time zone, as well. You can set the time zone by using the **system set timezone** command. When specifying daylight saving time, you'll need to use the **system set daylight-saving** command.

Note: If you configured the **ntp set server** command in the startup file and the X-Pedition does not receive a valid response from the configured NTP servers after startup, the X-Pedition will send an NTP request to each NTP server every minute until: (a) the X-Pedition receives a valid response from an NTP server; (b) the X-Pedition reaches the configured NTP query interval; or (c) you reconfigure the "ntp set server" command. During this time, the X-Pedition will not display any NTP related message. After it meets one of the above conditions, the X-Pedition returns to the normal NTP mode—this mode sends requests at configured query intervals and displays NTP messages.

Parameters

server <*nameiplist*> Specifies a list of host names or/and ip addresses of NTP servers. The maximum number of servers is 3. Specify each ip address in dotted-decimal-notion. If more than one servers are configured, they need to be separated by space and are surrounded with a set of quotes.

interval <*minutes*> Specifies how often (in minutes) the X-Pedition should synchronize with the server. The default synchronization interval is 60 minutes. Valid interval is between 1 minute to 10080 minutes (7 days).

source <*ipaddr*> Specifies the source IP address to be used by the X-Pedition for sending the NTP packet. The IP address must belong to one of the interfaces on the X-Pedition.

version <*num*> Specifies the NTP version number of the packet. The default version number is 3 (NTPv3). Valid value is 1-3.

Restrictions

None.

Examples

To send NTP packets to the NTP server 10.13.1.1 with default parameters:

```
xp(config)# ntp set server 10.13.1.1
```

To synchronize with a NTP server every 15 minutes with a specific source IP address:

```
xp(config)# ntp set server 10.13.1.1 interval 15 source 10.15.3.3
```

To configure three NTP servers with default parameters:

```
xp(config)# ntp set server “ntpserver1.org ntpserver2.org 10.17.5.5"
```

ntp show all

Purpose

Display NTP information about the X-Pedition.

Format

ntp show all

Mode

Enable

Description

The **ntp show all** command displays various NTP information about the XP. This information may include the last time a successful synchronization was made, the synchronization interval, the NTP version number, the NTP server list, and so on.

Parameters

None.

Restrictions

None.

Example

```
xp# ntp show all
NTP status:
    Synchronization interval: 60 mins
    Version: NTPv3
    Servers:
        ntpserver1.org
        ntpserver2.org
        10.17.5.5
    Last successful contact: 2001-09-04 16:46:40
```

ntp synchronize server

Purpose

Manually force the X-Pedition to immediately synchronize with a NTP server.

Format

ntp synchronize server <*host*>

Mode

Enable

Description

The **ntp synchronize server** command forces the X-Pedition to immediately synchronize its clock with the NTP server. Unlike the Configuration mode **ntp set server** command, this Enable mode command does not send periodic synchronization packets to the server. Instead, each time this command is executed, the X-Pedition synchronizes itself with the server. To have the X-Pedition synchronizes itself periodically, use the **ntp set server** command.

Parameters

<*host*>	Specifies the hostname or the IP address of the NTP server.

Restrictions

None.

Examples

To synchronize the X-Pedition against the NTP server 10.13.1.1:

```
xp(config)# ntp synchronize server 10.13.1.1
%NTP-I-TIMESYNC, Time synchronized to Mon Jan 22 23:11:28 2001
```

Chapter 44

ospf Commands

The **ospf** commands let you display and set parameters for the Open Shortest Path First (OSPF) routing protocol.

Command Summary

Table 35 lists the **ospf** commands. The sections following the table describe the command syntax.

Table 35. ospf commands

ospf add interface <*interfacename-or-IPaddr*> **to-area** <*area-addr*>**|backbone [type broadcast|non-broadcast|point-to-multipoint]**
ospf add nbma-neighbor <*IPaddr*> **to-interface** <*interfacename-or-IPaddr*> **[eligible]**
ospf add network | summary-range <*IPaddr/mask*> **to-area** <*area-addr*> **[restrict] [host-net]**
ospf add pmp-neighbor <*IPaddr*> **to-interface** <*interfacename-or-IPaddr*>
ospf add stub-host <*IPaddr*> **to-area [**<*area-addr*>**| backbone] [cost** <*num*>**]**
ospf add virtual-link <*number-or-string*> **neighbor** <*IPaddr*> **transit-area** <*area-num*>
ospf create area <*area-num*> **[backbone]**
ospf create-monitor destination <*hostname-or-IPaddr*>
ospf log router-lsas on|off | on detail

Table 35. ospf commands (Continued)

ospf monitor statistics| errors| next-hop-list| interfaces| neighbors | version **[destination** <*hostname-or-IPaddr*>**] [auth-key** <*string*>**]** **ospf monitor lsdb [display-retransmit-list] [destination** <*hostname-or-IPaddr*>**]** **[auth-key** <*string*>**] [area]** **ospf monitor routes [type all| asbrs-in-area| area-border-routers|** **asbrs-other-areas| networks-in-area| networks-other-areas | as-routes]** **[destination** <*hostname-or-IPaddr*>**] [auth-key** <*string*>**]** **ospf monitor lsa area-id** <*IPaddr*> **type router-links| network-links|** **summary-networks| summary-asbr| as-external ls-id** <*IPaddr*> **adv-rtr** <*IPaddr*> **[destination** <*hostname-or-IPaddr*>**] [auth-key** <*string*>**]** **ospf monitor as-external-db [display-retransmit-list] [destination** <*hostname-or-IPaddr*>**]** **[auth-key** <*string*>**]**
ospf set area <*area-num*> **[stub] [no-summary] [stub-cost** <*num*>**] [authentication-method** **none| simple| md5]**
ospf set ase-defaults {[preference <*num*>**] |[cost** <*num*>**] |** **[type** <*num*>**]| [inherit-metric]}**
ospf set export-interval <*num*>
ospf set export-limit <*num*>
ospf set interface <*name-or-IPaddr*>**|all [state disable|enable] [cost** <*num*>**] [no-multicast]** **[retransmit-interval** <*num*>**] [transit-delay** <*num*>**] [priority** <*num*>**]** **[hello-interval** <*num*>**] [router-dead-interval** <*num*>**] [poll-interval** <*num*>**]** **[key-chain** <*num-or-string*>**] [authentication-method none| simple| md5]**
ospf set virtual-link <*number-or-string*> **[state disable|enable] [cost** <*num*>**] [no-multicast]** **[retransmit-interval** <*num*>**] [transit-delay** <*num*>**] [priority** <*num*>**]** **[hello-interval** <*num*>**] [router-dead-interval** <*num*>**] [poll-interval** <*num*>**]** **[key-chain** <*num-or-string*>**] [authentication-method none| simple| md5]**
ospf show <*option-list*>
ospf start| stop
ospf trace [lsa-builds| lsa-transmit| lsa-receive spf| debug| packets {detail| send| receive}| **hello {detail| send| receive}| dd {detail| send| receive}| request {detail| send| receive}|** **update {detail| send| receive}| ack {detail| send| receive}| local-options [all| general| state|** **normal| policy| task| timer| route| none]]**

ospf add interface

Purpose

Associates an interface with an OSPF area.

Format

ospf add interface <*interfacename-or-IPaddr*> **to-area** <*area-addr*>**|backbone**
[type broadcast|non-broadcast|point-to-multipoint]

Mode

Configure

Parameters

<*interfacename-or-IPaddr*>
An interface name or an IP address.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

to-area <*area-addr*>**|backbone**
OSPF Area with which this interface is to be associated.

type
Specifies whether the interface is broadcast, non-broadcast, or point-to-multipoint. Specify one of the following:

- **broadcast** (default)
- **non-broadcast**
- **point-to-multipoint**

Restrictions

None.

ospf add nbma-neighbor

Purpose

Specifies an OSPF NBMA Neighbor.

Format

ospf add nbma-neighbor <*IPaddr*> **to-interface** <*interfacename-or-IPaddr*> **[eligible]**

Mode

Configure

Parameters

nbma-neighbor <*IPaddr*>
The nbma neighbor you will add.

to-interface <*interfacename-or-IPaddr*>
Adds the neighbor to the specified OSPF interface.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

eligible
Specifies whether an OSPF NBMA Neighbor is eligible for becoming a designated router.

Restrictions

None.

ospf add network | summary-range

Note: The **OSPF add network** command may use the same syntax as other vendors. Please review the documentation for the proper use of this command.

Purpose

Configures summary-ranges on Area Border Routers (ABRs). This allows you to reduce the amount of routing information propagated between areas.

On the X-Pedition, summary-ranges are created using the **ospf add summary-range** command—the networks specified using this command describe the scope of an area. Intra-area Link State Advertisements (LSAs) that fall within the specified ranges are not advertised into other areas as inter-area routes. Instead, the specified ranges/networks are advertised as summary network LSAs. If you specify the **restrict** option, the summary network LSAs are not advertised. Each intra-area LSA that does not fall into any range is advertised as an OSPF Type-3 or 4 LSA.

Format

ospf add network | summary-range <*IPaddr/mask*> **to-area** <*area-addr*> **[restrict] [host-net]**

Mode

Configure

Parameters

<*IPaddr/mask*>
IP Address and network mask value representing the summary-range. Example: 16.122.0.0/255.255.0.0 or 16.122.0.0/16.

to-area <*area-addr*>
OSPF Area with which this summary-range is to be associated.

restrict
If the restrict option is specified for a network/summary-range, then that network is not advertised in Summary network LSAs.

host-net
Specifies that the network is an OSPF Host Network.

Restrictions

Although this does not apply to most changes to OSPF and other routing-based entries in the configuration file, the following actions force the OSPF Link State Databases (LSDB) to re-initialize:

- Adding a network to or removing one from an area.

- Changing an area's type.
- Adding a summary range to or removing one from an Area Border Router.

Example

In the following example, two summary ranges are created:

```
ospf add summary-range 207.135.16.0/24 to-area 207.135.0.0
ospf add summary-range 207.135.17.0/24 to-area 207.135.0.0 restrict
```

Intra-area Link State Advertisements (LSAs) that fall within the range 207.135.16.0/24 are not advertised into other areas as inter-area routes. Instead, the specified range 207.135.16.0/24 is advertised as summary network LSA.

Because the summary range 207.135.17.0/24 has the restrict option associated with it, intra-area link state advertisements (LSAs) that fall within it are not advertised as summary network LSA. Using this mechanism, one can have "hidden networks" within an area, which are not advertised to other areas.

ospf add pmp-neighbor

Purpose

Specifies an OSPF Point-to-Multipoint Neighbor.

Format

ospf add pmp-neighbor <*IPaddr*> **to-interface** <*interfacename-or-IPaddr*>

Mode

Configure

Description

The **ospf add pmp-neighbor** configures a Point-to-Multipoint neighbor router on an interface. A Point-to-Multipoint connectivity is used when the network does not provide full connectivity to all routers in the network. As in the case of NBMA (non-broadcast multiple access) networks, a list of neighboring routers reachable over a PMP network should be configured so that the router can discover its neighbors.

Parameters

pmp-neighbor <*IPaddr*>
Specifies the point-to-multipoint neighbor.

to-interface <*interfacename-or-IPaddr*>
Adds the neighbor to the specified OSPF interface.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

None.

Example

To add a point-to-multipoint neighbor with IP address 134.141.179.141 to the OSPF interface 134.141.179.152:

```
xp(config)# ospf add pmp-neighbor 134.141.179.141 to-interface 134.141.179.152
```

ospf add stub-host

Purpose

Adds a stub-host to an OSPF area.

Format

ospf add stub-host <*IPaddr*> **to-area [**<*area-addr*>**| backbone] [cost** <*num*>**]**

Mode

Configure

Parameters

to-area <*area-addr*>**|backbone**
OSPF Area to which you are adding a stub host.

cost <*num*>
The cost that should be advertised for this directly attached stub host. Specify a number from 0 – 65535.

Restrictions

None.

ospf add virtual-link

Purpose

Creates an OSPF Virtual Link.

Format

ospf add virtual-link <*number-or-string*> **neighbor** <*IPaddr*>
transit-area <*area-num*>

Mode

Configure

Parameters

<*number-or-string*>
A number or character string identifying the virtual link.

neighbor <*IPaddr*>
The IP address of an OSPF virtual link neighbor.

transit-area <*area-num*>
The Area ID of the transit area.

Restrictions

None.

ospf create area

Purpose

Create an OSPF area.

Format

ospf create area <*area-num*>|**backbone**

Mode

Configure

Parameters

<*area-num*>	The Area ID. Area IDs are formatted like IP addresses: <*num*>.<*num*>.<*num*>.<*num*>.
backbone	Specifies that the Area you are adding is the backbone area.

Restrictions

None.

ospf create-monitor

Purpose

Create an OSPF monitor destination.

Format

ospf create-monitor destination <*hostname-or-IPaddr*>

Mode

Enable

Parameters

destination <*hostname-or-IPaddr*>
Specifies the destination whose OSPF activity is to be monitored.

Restrictions

None.

ospf log router-lsas

Purpose

Logs Router LSAs from incoming link-state update packets to the console and Syslog server.

Format

ospf log router-lsas on|off | on detail

Mode

Enable.

Parameters

off Turns logging off (default).

on Turns logging on. Gives the user basic information that states which neighbor sent the update and which router advertised each LSA contained within the update packet

on detail
Logs detailed information from the Router LSA.

Restrictions

None.

Examples

To view basic log information, enter the following command from Enable mode:

```
xp# ospf log router-lsas on
2002-07-25 12:09:40 %OSPF-I-UPDATE, Update received from 50.50.50.23 on interface to23.
2002-07-25 12:09:40 %OSPF-I-ADVRTR, Advertising Router: 15.15.15.15
```

To display detailed log information contained within the Router LSA, enter the following from Enable mode:

```
xp# ospf log router-lsas on detail
2002-07-25 12:09:40 %OSPF-I-UPDATE, Update received from 50.50.50.23 on interface to23.
2002-07-25 12:09:40 %OSPF-I-ADVRTR, Advertising Router: 15.15.15.15
2002-07-25 12:09:40 %OSPF-I-LINK, type: TRANS NET Link Id: 40.40.40.23 LinkData: 40.40.40.15
2002-07-25 12:09:40 %OSPF-I-LINK, type: TRANS NET Link Id: 30.30.30.16 LinkData: 30.30.30.15
2002-07-25 12:09:40 %OSPF-I-LINK, type: STUB Link Id: 80.80.80.0 LinkData: 255.255.255.0
2002-07-25 12:09:40 %OSPF-I-LINK, type: STUB Link Id: 15.15.15.15 LinkData: 255.255.255.255
```

ospf monitor

Purpose

Monitor OSPF.

Format

ospf monitor statistics| errors| next-hop-list| interfaces| neighbors | version [destination <*hostname-or-IPaddr*>**] [auth-key** <*string*>**]**

ospf monitor lsdb [display-retransmit-list] [destination <*hostname-or-IPaddr*>**] [auth-key** <*string*>**] [area]**

ospf monitor routes [type all| asbrs-in-area| area-border-routers| asbrs-other-areas| networks-in-area| networks-other-areas | as-routes] [destination <*hostname-or-IPaddr*>**] [auth-key** <*string*>**]**

ospf monitor lsa area-id <*IPaddr*> **type router-links| network-links| summary-networks| summary-asbr| as-external ls-id** <*IPaddr*> **adv-rtr** <*IPaddr*> **[destination** <*hostname-or-IPaddr*>**] [auth-key** <*string*>**]**

ospf monitor as-external-db [display-retransmit-list] [destination <*hostname-or-IPaddr*>**] [auth-key** <*string*>**]**

Mode

Enable

Parameters

destination <*hostname-or-IPaddr*>
Monitors the specified OSPF destination. Default is the router on which the command is executed.

auth-key <*string*>
Specifies the authorization key for the OSPF destination. This option is not needed if the OSPF destination does not require a key or if an authorization was specified using the **ospf monitor create-destination** command.

statistics
Shows input/output statistics for monitor request, hello, data base description, link-state request, link-state update, and link-state ack packets. Area statistics are provided, which describe the total number of routing neighbors and number of active OSPF interfaces. Routing table statistics are summarized and reported as the number of intra-area routes, inter-area routes, and AS external data base entries.

errors

Shows the various error conditions which can occur between OSPF routing neighbors and the number of occurrences for each.

DD: extern option mismatch	Indicates a disagreement between this router and a neighbor about whether or not they belong to a stub area. The packet will be ignored.
DD: neighbor state low	A Database Description packet was received from a neighbor considered to be in state *Down* or *Attempt*. The packet will be ignored—neighbors must be in a state greater than Attempt for Database Description packets to be valid.
DD: router id confusion	A Database Description packet received during the exchange start procedure from a neighboring router claims to have the same router ID as this router. The packet will be ignored and the attempt to start database exchange will fail.
DD: unknown LSA type	Five types of LSA are defined: Router, Network, Network Summary, AS Boundary Summary, and AS External—two other types, Group Membership and type 7 AS External, are recognized but are currently ignored. This error indicates that the router received a Database Description packet that contains a summary for an LSA that is not one of these types. The advertisement will be ignored.
HELLO: dead timer mismatch	All routers connected to a common logical network must agree on the Hello timer. Any Hello packet with a different Hello interval than the one configured on the receiving interface will be ignored.
HELLO: extern option mismatch	This error indicates a disagreement between routers as to whether or not the area is a stub. The Hello packet will be ignored.
HELLO: hello timer mismatch	All routers connected to a common logical network must agree on the Hello timer. Any Hello packets with a different Hello interval than that configured on the receiving interface will be ignored.
HELLO: NBMA neighbor unknown	A Hello packet was received from a non-configured NBMA neighbor. The packet will be ignored.
HELLO: netmask mismatch	A Hello packet was received on an interface configured with a different network mask. All routers attached to a common logical network must agree on the network mask used for the network. The packet will be ignored.
HELLO: router id confusion	A Hello packet was received from a neighboring router claiming to have the same router ID as this router. The packet will be ignored.
HELLO: virtual neighbor unknown	A Hello packet was received by a virtual interface that is not yet “up.” Virtual interfaces are brought up as a result of the shortest path first calculation. The packet will be ignored.

LS ACK: bad ack	An acknowledgment was received for an LSA that does not match the one present in the router's link state database. The acknowledgment is ignored.
LS ACK: duplicate ack	A duplicate acknowledgment was received. The acknowledgment is ignored.
LS ACK: neighbor state low	A neighbor must be in at least *Exchange* state before it can send LS acknowledgments. If this is not the case, the acknowledgment ignored.
LS ACK: Unknown LSA type	Five types of LSA are defined: Router, Network, Network Summary, AS Boundary Summary, and AS External—two other types, Group Membership and type 7 AS External, are recognized but currently ignored. This error indicates that an acknowledgment for some other type of link state advertisement was received. The acknowledgment is ignored.
LS REQ: bad request	A request for an LSA was received for an unrecognized LSA type or an LSA that is not in the router's link state database.
LS REQ: empty request	An empty request for link state information was received from a neighbor. The packet is ignored.
LS_REQ: neighbor state low	A neighbor must be in state *Exchange*, *Loading*, or *Full* to send Link State Request packets. If not, the packet is ignored.
LS UPD: LSA checksum bad	Any LSA with bad checksums is considered corrupted and is ignored. Since the flooding procedure is reliable, the LSA will be sent again by the originating router.
LS UPD: received less recent LSA	The router has a more recent copy of an LSA than it received. This shows that some part of the OSPF network is not updating its link state database with new information—the advertisement is ignored.
LS UPD: unknown LSA type	Five types of LSA are defined: Router, Network, Network Summary, AS Boundary Summary, and AS External—two other types, Group Membership and type 7 AS External, are recognized but currently ignored. This error indicates that the router received some other type of link state advertisement. The advertisement is ignored.
OSPF: area mismatch	A protocol packet was received that, from its format, should be from a virtual link. However, because this router is not an area border router, the packet is invalid and will be ignored.
OSPF: bad area id	A protocol packet received from another router on the same logical network was configured to belong to a different area. All routers attached to a common logical network must agree on the area to which that network belongs. The packet will be ignored and routing information will not be exchanged with this router.
OSPF: bad authentication key	A protocol packet was received with an incorrect authentication key. The packet will be ignored.

OSPF: bad authentication type	All routers belonging to a common area must agree on the authentication type (currently none, simple, and MD5 passwords). A protocol packet was received with an authentication type that differs from the one configured for this router. The packet will be ignored.
OSPF: bad checksum	A corrupted OSPF protocol packet was received. The packet will be ignored.
OSPF: bad packet type	An OSPF packet was received that was not one of the following: Hello, Database Description, Link State Request, Link State Update, or Link State Acknowledgment. The packet will be ignored.
OSPF: bad version	A protocol packet from a router running a version of OSPF other than version 2 was received. The packet will be ignored and routing information will not be exchanged with this router.
OSPF: bad virtual link	A packet has been received which, from its format, should be from a virtual link. However, this router does not have a virtual interface corresponding to this packet. The packet is therefore deemed to be invalid and will be ignored.
OSPF: packet size > ip length	A packet was received that claims that the length of the OSPF packet is larger than the total size of the IP packet. The packet will be ignored.
OSPF: packet too small	The length of all OSPF routing protocol packets is indicated in the header. A protocol packet was received whose length does not match this value. The packet will be ignored.
OSPF: transmit error	An error occurred when attempting to transmit an OSPF packet over an interface.
OSPF: unknown neighbor	A protocol packet other than a Hello packet was received from an unknown neighbor. The packet will be ignored.

next-hop-list
Shows information about all valid next hops mostly derived from the SPF calculation.

interfaces
Shows information about all interfaces configured for OSPF. Information reported includes the area, interface IP address, interface type, interface state, cost, priority, and the IP address of the Designated Router and Backup Designated Router for the network.

neighbors
Shows information about all OSPF routing neighbors. Information reported includes the area, local interface address, router ID, neighbor IP address, state, and mode.

version
Show information about all OSPF routing versions.

lsdb
Displays the link-state database (except for ASEs). This table describes the routers and

networks making up the AS. If the display-retransmit-list option is specified, the retransmit list of neighbors held by this lsdb structure will also be printed.

display-retransmit-list – Displays the retransmit list from the link state database.

area – Displays the area for which lsdb is to be displayed.

routes

Displays the OSPF routing table. This table reports the AS border routes, area border routes, summary AS border routes, networks, summary networks and AS external networks currently managed via OSPF.

type all
Shows all OSPF routes.

type asbrs-in-area
Shows routes to AS boundary routers in this area.

type area-border-routers
Shows routes to area border routers for this area.

type asbrs-other-areas
Shows summary routes to AS boundary routers in other areas.

type networks-in-area
Shows routes to networks in this area.

type networks-other-areas
Shows routes to networks in other areas.

type as-routes
Shows AS routes to non-OSPF networks.

lsa

Displays the link state advertisement. Area_Id is the OSPF area for which the query is directed. Adv_Rtr is the router -id of the router which originated this link state advertisement. Type specifies the type of advertisement to request:

area-id <*IPaddr*>
Specifies the OSPF area.

type router-links
Requests router link advertisements that describe the collected states of the router interfaces. ls-id is set to the originating router's router-id.

type network-links
Requests network link advertisements that describe the set of routers attached to the network. ls-id is set to the IP interface address of the designated router for the network.

type summary-networks
Request summary-link advertisements describing routes to networks. ls-id is set to the IP address of the destination network.

type summary-asbr
Requests summary-link advertisements describing routes to AS boundary routers. ls-id is set to the AS boundary router's router-id.

type as-external
Requests AS external link state advertisements. ls-id is set to the IP address of the destination network.

ls-id *<IPaddr>*
Species the ls-id for the type of link-state advertisement requested

adv-rtr *<IPaddr>*
Requests the router ID of the originating router.

as-external-db
Display the AS external data base entries. This table reports the advertising router, forwarding address, age, length, sequence number, type, and metric for each AS external route. If the display-retransmit-list option is specified, the retransmit list of neighbors held by this lsdb structure will also be printed.

Restrictions

None.

Examples

The following are examples of **ospf monitor** commands.

ospf monitor statistics

```
xp# ospf monitor statistics

IO stats
Input  Output  Type
8       0  Monitor request
1322    1314  Hello
716     721  DB Description
39     728  Link-State Req
3037    3355  Link-State Update
1317     354  Link-State Ack
ASE: 1903 checksum sum 3BB0F22

LSAs originated: 1915   received: 17
Router: 5  ASE: 1910

Area 0.0.0.0:
Neighbors: 3    Interfaces: 3
Spf: 3  Checksum sum 6CB41
DB: rtr: 5 net: 5 sumasb: 0 sumnet: 2

Routing Table:
Intra Area: 5   Inter Area: 4   ASE: 1
```

ospf monitor errors

```
xp# ospf monitor errors

Packets Received:
10: Monitor request                1342: Hello
716: DB Description                  39: Link-State Req
3212: Link-State Update              1536: Link-State Ack

Packets Sent:
0: Monitor response               1335: Hello
721: DB Description                728: Link-State Req
3907: Link-State Update             359: Link-State Ack

Errors:
0: IP: bad destination              0: IP: bad protocol
0: IP: received my own packet       0: OSPF: bad packet type
0: OSPF: bad version                0: OSPF: bad checksum
0: OSPF: bad area id                0: OSPF: area mismatch
0: OSPF: bad virtual link           0: OSPF: bad authentication type
0: OSPF: bad authentication key     0: OSPF: packet too small
0: OSPF: packet size > ip length    1: OSPF: transmit error
0: OSPF: interface down             0: OSPF: unknown neighbor
0: HELLO: netmask mismatch          0: HELLO: hello timer mismatch
```

```
0: HELLO: dead timer mismatch          0: HELLO: extern option mismatch
0: HELLO: router id confusion          0: HELLO: virtual neighbor
unknown
0: HELLO: NBMA neighbor unknown        0: DD: neighbor state low
0: DD: router id confusion             0: DD: extern option mismatch
0: DD: unknown LSA type                1: LS ACK: neighbor state low
0: LS ACK: bad ack                  1140: LS ACK: duplicate ack
0: LS ACK: Unknown LSA type            0: LS REQ: neighbor state low
0: LS REQ: empty request               0: LS REQ: bad request
8: LS UPD: neighbor state low          0: LS UPD: newer self-gen LSA
0: LS UPD: LSA checksum bad          131: LS UPD: received less recent
LSA
0: LS UPD: unknown LSA type            2: Interface: Not configured for
OSPF
0: Interface: Invalid type             0: Interface: Mcast disabled.
0: Interface: Invalid state            0: Interface: Address not found
1: No vlinks and src is non local
```

ospf monitor next-hop-list

```
xp# ospf monitor next-hop-list

Next hops:

Address          Type      Refcount  Interface
-----------------------------------------------------------------
10.12.1.1        Neighbor        6  10.12.1.2       to-c4500
10.12.1.2        Direct          1  10.12.1.2       to-c4500
150.1.0.1        Direct          1  150.1.0.1       to-aval-eth5
172.23.1.5       Direct          3  172.23.1.5      to-xp6
172.23.1.6       Neighbor        5  172.23.1.5      to-xp6
172.23.1.21      Direct          3  172.23.1.21     to-xp1
172.23.1.22      Neighbor       19  172.23.1.21     to-xp1
172.23.1.25      Direct          3  172.23.1.25     lo
222.1.1.1        Direct          1  222.1.1.1       to-linux1
```

ospf monitor interfaces

```
xp# ospf monitor interfaces
>sent to 127.0.0.1

Source <<127.0.0.1      >>

Area: 0.0.0.0
IP Address      Type  State    Cost Pri DR             BDR
------------------------------------------------------------------------------
172.23.1.5      Bcast BackupDR 2    2   172.23.1.6      172.23.1.5
10.12.1.2       Bcast BackupDR 1    2   10.12.1.1       10.12.1.2
172.23.1.21     Bcast BackupDR 1    2   172.23.1.22     172.23.1.21
done
```

ospf monitor neighbors

```
xp# ospf monitor neighbors
> sent to 127.0.0.1

Source <<127.0.0.1      >>

Interface: 172.23.1.5      Area: 0.0.0.0
Router Id       Nbr IP Addr     State      Mode   Prio
-------------------------------------------------------
0.0.0.6         172.23.1.6      Full       Slave  1

Interface: 10.12.1.2       Area: 0.0.0.0
Router Id       Nbr IP Addr     State      Mode   Prio
-------------------------------------------------------
172.23.1.14     10.12.1.1       Full       Slave  1

Interface: 172.23.1.21     Area: 0.0.0.0
Router Id       Nbr IP Addr     State      Mode   Prio
-------------------------------------------------------
0.0.0.1         172.23.1.22     Full       Master 1
done
```

ospf monitor routes

```
xp# ospf monitor routes
> sent to 127.0.0.1

Source <<127.0.0.1      >>
AS Border Routes:
Router          Cost AdvRouter      NextHop(s)
-------------------------------------------------------
Area 0.0.0.0:
0.0.0.6          2 0.0.0.6        172.23.1.6
172.23.1.22
0.0.0.4          0 0.0.0.4
0.0.0.1          1 0.0.0.1        172.23.1.22

Total AS Border routes: 3

Area Border Routes:
Router          Cost AdvRouter      NextHop(s)
-------------------------------------------------------
Area 0.0.0.0:
0.0.0.3          2 0.0.0.3        172.23.1.22
0.0.0.1          1 0.0.0.1        172.23.1.22

Total Area Border Routes: 2

Summary AS Border Routes:
Router          Cost AdvRouter      NextHop(s)
-------------------------------------------------------

Networks:
Destination       Area          Cost Type NextHop        AdvRouter
-------------------------------------------------------------------------------

172.23.1.4/30     0.0.0.0          2 Net  172.23.1.5      0.0.0.6
10.12.1.0/30      0.0.0.0          1 Net  10.12.1.1       172.23.1.14
172.23.1.20/30    0.0.0.0          1 Net  172.23.1.21     0.0.0.1
172.23.1.25       0.0.0.0          0 Stub 172.23.1.25     0.0.0.4
172.23.1.8/30     0.0.0.0          2 Net  172.23.1.22     0.0.0.1
10.12.1.4/30      0.0.0.0          2 Net  172.23.1.22     172.23.1.14
172.23.1.14       0.0.0.0          2 Stub 10.12.1.1       172.23.1.14
172.23.1.26       0.0.0.0          3 Stub 172.23.1.6      0.0.0.6
172.23.1.22
16              0.0.0.0          2 SNet 172.23.1.22     0.0.0.1
ASEs:
Destination       Cost E      Tag NextHop        AdvRouter
-------------------------------------------------------------------------------
15.1              1   1 c0000000 172.23.1.22     0.0.0.1
Total nets: 9
Intra Area: 5   Inter Area: 4   ASE: 1
done
```

ospf monitor lsdb

```
xp# ospf monitor lsdb

LS Data Base:
Area: 0.0.0.0
Type LinkState ID    AdvRouter       Age Len Sequence Metric Where
------------------------------------------------------------------------
Rtr  10.1.1.1        10.1.1.1         21 36  8000000c      0 SpfTree
Rtr  172.26.0.7      172.26.0.7       417 36  8000000c      0 SpfTree
Net  130.1.1.2       172.26.0.7       417 32  80000007      0 SpfTree
SNet 150.20          172.26.0.7        41 28  80000007     20 Uninitialized
SNet 140.1.1         10.1.1.1         382 28  80000024     2 Inter List
SNet 140.1.2         10.1.1.1        1727 28  80000007     2 Inter List
SNet 140.1.3         10.1.1.1        1727 28  80000007     2 Inter List
SNet 140.1.4         10.1.1.1        1727 28  80000006     3 Inter List
SNet 140.1.5         10.1.1.1         222 28  800000ce     3 Inter List

Area: 140.1
Type LinkState ID    AdvRouter       Age Len Sequence Metric Where
------------------------------------------------------------------------
Stub 140.1.3         10.1.1.1          2 24  0            0 SpfTree
Rtr  10.1.1.1        10.1.1.1          2 60  8000063b     0 SpfTree
Rtr  140.1.1.2       140.1.1.2         4 48  80000659     0 Clist
Rtr  140.1.5.1       140.1.5.1       814 60  80000009     0 Clist
Rtr  64.86.127.1     64.86.127.1     1491 48  8000000b     0 Clist
Net  140.1.1.1       10.1.1.1        1821 32  80000004     0 Uninitialized
Net  140.1.1.2       140.1.1.2       1184 32  80000009     0 SpfTree
Net  140.1.2.1       10.1.1.1          21 32  80000008     0 SpfTree
Net  140.1.4.2       64.86.127.1     1491 32  80000007     0 SpfTree
Net  140.1.5.2       140.1.1.2       1191 32  80000007     0 SpfTree
SNet 150.20          10.1.1.1        1727 28  80000005    40 Inter List
SNet 130.1.1         10.1.1.1        1727 28  80000005    20 Inter List
```

LSDB Field Definitions

Field	Description
Area	The OSPF area to which this links state database is associated.
Type	The type of link state advertisement. There are 6 types supported on the X-Pedition: • Stub Network (Found in Router LSAs) • Router LSA • Network LSA • Network (type 3) Summary LSA • ASBR* (type 4) Summary LSA • AS External LSA *ASBR: Autonomous System Border Router

Field	Description
Link State ID	The link state ID is determined by the type of LSA: • Router LSA: Originating Router's router id. • Network LSA: IP Interface address of the Designated Router. • Network Summary LSA: IP address of the destination network. • ASBR Summary LSA: ASBR's router id. • AS External LSA: Destination network's IP address.
AdvRouter	The router ID of the originating router who advertised this link state.
Age	Age (in seconds) of the link state advertisement.
Len	Length of the link state advertisement in bytes.
Sequence	Sequence number associated with this link state advertisement.
Metric	Metric used to calculate the shortest path to a destination.
Where	"Where" determines where the link state advertisement is located during the SPF (Shortest Path First) calculation. There are 9 distinct locations. In order to understand these location, the user must know how OSPF and SPF calculation work: • Uninitialized: The LSA is not being used. • CList: The LSA is on the candidate list. • SpfTree: Used on the SPF tree that is built during the SPF calculation. • SumAsb List: Reachable ASBR from the attached area. • SumNet List: Reachable network from the attached area. • Inter List: LSA is on the inter-area list imported from the backbone. • ASE List: On the ASE list. • ASE Infinity and Sum Infinity: Infinity is used to prematurely age an LSA. Therefore, the LSA cannot participate in the SPF calculation.

ospf monitor as-external-db

```
xp# ospf monitor as-external-db

AS External Data Base:
Destination      AdvRouter      Forward Addr     Age Len Sequence T Metric
-------------------------------------------------------------------------
130.58.225       0.0.0.4        0.0.0.0          201 36  80000001 21
130.58.174       0.0.0.4        0.0.0.0          201 36  80000001 21
130.56.235       0.0.0.4        0.0.0.0          236 36  80000001 21
130.56.184       0.0.0.4        0.0.0.0          236 36  80000001 21
130.54.245       0.0.0.4        0.0.0.0          238 36  80000001 21
130.54.194       0.0.0.4        0.0.0.0          239 36  80000001 21
130.52.255       0.0.0.4        0.0.0.0          241 36  80000001 21
130.52.204       0.0.0.4        0.0.0.0          241 36  80000001 21
130.51.9         0.0.0.4        0.0.0.0          211 36  80000001 21
130.50.214       0.0.0.4        0.0.0.0          211 36  80000001 21
130.49.19        0.0.0.4        0.0.0.0          213 36  80000001 21
130.48.224       0.0.0.4        0.0.0.0          214 36  80000001 21
130.47.29        0.0.0.4        0.0.0.0          216 36  80000001 21
130.46.234       0.0.0.4        0.0.0.0          248 36  80000001 21
130.45.39        0.0.0.4        0.0.0.0          251 36  80000001 21
130.44.244       0.0.0.4        0.0.0.0          251 36  80000001 21
130.43.49        0.0.0.4        0.0.0.0          253 36  80000001 21
130.42.254       0.0.0.4        0.0.0.0          221 36  80000001 21
130.41.59        0.0.0.4        0.0.0.0          256 36  80000001 21
130.41.8         0.0.0.4        0.0.0.0          256 36  80000001 21
130.39.69        0.0.0.4        0.0.0.0          258 36  80000001 21
130.39.18        0.0.0.4        0.0.0.0          258 36  80000001 21
130.37.79        0.0.0.4        0.0.0.0          261 36  80000001 21
130.37.28        0.0.0.4        0.0.0.0          261 36  80000001 21
130.35.89        0.0.0.4        0.0.0.0          263 36  80000001 21
130.35.38        0.0.0.4        0.0.0.0          263 36  80000001 21
130.33.99        0.0.0.4        0.0.0.0          267 36  80000001 21
130.33.48        0.0.0.4        0.0.0.0          267 36  80000001 21
130.31.109       0.0.0.4        0.0.0.0          272 36  80000001 21
130.31.58        0.0.0.4        0.0.0.0          272 36  80000001 21
130.29.119       0.0.0.4        0.0.0.0          277 36  80000001 21
130.29.68        0.0.0.4        0.0.0.0          277 36  80000001 21
130.27.129       0.0.0.4        0.0.0.0          282 36  80000001 21
130.27.78        0.0.0.4        0.0.0.0          282 36  80000001 21
130.25.139       0.0.0.4        0.0.0.0          287 36  80000001 21
130.25.88        0.0.0.4        0.0.0.0          287 36  80000001 21
130.23.149       0.0.0.4        0.0.0.0          292 36  80000001 21
130.23.98        0.0.0.4        0.0.0.0          292 36  80000001 21
130.21.159       0.0.0.4        0.0.0.0          297 36  80000001 21
```

ospf set area

Purpose

Sets the parameters for an OSPF area.

Format

ospf set area <*area-num*> **[stub] [no-summary] [stub-cost** <*num*>**] [authentication-method none| simple| md5]**

Mode

Configure

Parameters

<*area-num*>
The Area ID.

stub
Makes this Area a stub area.

no-summary
Specifies that this is a fully stubby area.

stub-cost <*num*>
Specifies the cost to be used to inject a default route into the area. Specify a number from 0-65535.

authentication-method none|simple|md5
Specifies the authentication method used within the area. Specify one of the following:

none Does not use authentication.

simple Uses a simple string (password) up to 16 characters in length for authentication. If you chose this authentication method, you should also specify a key-chain identifier using the key-chain option.

md5 Uses the MD5 algorithm to create a crypto-checksum of an OSPF packet and an authentication key of up to 16 characters.

Restrictions

Although this does not apply to most changes to OSPF and other routing-based entries in the configuration file, the following actions force the OSPF Link State Databases (LSDB) to re-initialize:

- Adding a network to or removing one from an area.

- Changing an area's type.
- Adding a summary range to or removing one from an Area Border Router.

ospf set ase-defaults

Purpose

Sets the defaults used when importing OSPF ASE routes into the routing table and exporting routes from the routing table into OSPF ASEs.

Format

ospf set ase-defaults {[preference <*num*>**] |[cost** <*num*>**] |**
[type <*num*>**]| [inherit-metric]}**

Mode

Configure

Parameters

preference <*num*>
Specifies the preference of OSPF ASE routes. Specify a number between 0 and 255.

cost <*num*>
Specifies the cost used when exporting non-OSPF route into OSPF as an ASE. Specify a number from 0 – 65535.

type <*num*>
Specifies the ASE type (1 or 2) for routes exported from the routing table into OSPF—the default is type **2**. You can change the default using the **type** option, or override the type in OSPF export policies.

inherit-metric
Allows an OSPF ASE route to inherit the metric of the external route when no metric is specified on the export. A metric specified with the export command takes precedence. The cost specified in the default is used if you do not specify **inherit-metric**.

Restrictions

None.

ospf set export-interval

Purpose

Specifies the interval at which ASE LSAs will be generated and flooded into OSPF. The default is once per second.

Format

ospf set export-interval <*num*>

Mode

Configure

Parameters

<*num*> The interval in seconds. Specify a number equal to or greater than 1. The default is 1 (once per second).

Restrictions

None.

ospf set export-limit

Purpose

Specifies how many ASEs will be generated and flooded in each batch.

Format

ospf set export-limit <*num*>

Mode

Configure

Parameters

<*num*> The export limit. Specify a number equal to or greater than 1 (the default is 250).

Restrictions

None.

ospf set interface

Purpose

Sets parameters for an OSPF interface.

Format

ospf set interface <*name-or-IPaddr*>**|all [state disable|enable] [cost** <*num*>**] [no-multicast] [retransmit-interval** <*num*>**] [transit-delay** <*num*>**] [priority** <*num*>**] [hello-interval** <*num*>**] [router-dead-interval** <*num*>**] [poll-interval** <*num*>**] [key-chain** <*num-or-string*>**] [authentication-method none| simple| md5]**

Mode

Configure

Parameters

<*name-or-IPaddr*>**|all**
The OSPF interface for which you are setting OSPF parameters.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

state disable|enable
Enables or disables OSPF on the interface.

cost <*num*>
The cost associated with this interface. Specify a number from 0 – 65535.

The total cost to get to a destination is calculated by adding up the cost of all interfaces that a packet must cross to reach a destination.

The default cost of an OSPF interface is calculated using its bandwidth. A VLAN that is attached to an interface could have several ports of differing speeds. The bandwidth of an interface is represented by the highest bandwidth port that is part of the associated VLAN. The cost of an OSPF interface is inversely proportional to this bandwidth. The cost is calculated using the following formula:

Cost = 2000000000 / *speed (in bps)*

The following is a table of the port types and the OSPF default cost associated with each type:

Port Media Type	Speed	OSPF Default Cost
Ethernet 1000	1000 Mbps	2
Ethernet 10/100	100 Mbps	20
Ethernet 10/100	10 Mbps	200
WAN (T1)	1.5 Mbps	1333
WAN (T3)	45 Mbps	44

no-multicast
Instructs the X-Pedition not to send multicast packets to neighbors on point-to-point interfaces.

retransmit-interval <*num*>
The number of seconds between link state advertisement retransmissions for adjacencies belonging to this interface. Specify a number equal to or greater than 1. The default is 5.

transit-delay <*num*>
The estimated number of seconds required to transmit a link state update over this interface. Transit delay takes into account transmission and propagation delays and must be greater than 0. Specify a number equal to or greater than 1. The default is 1.

priority <*num*>
A number between 0 and 255 specifying the priority for becoming the designated router on this interface. When two routers attached to a network both attempt to become the designated router, the one with the higher priority wins. A router whose router priority is set to 0 is ineligible to become designated router. Specify a number from 0 – 255. The default is 1.

hello-interval <*num*>
The length of time, in seconds, between hello packets that the router sends on this interface. Specify a number from 0 – 255. The default is 10 for broadcast interfaces and 30 for point-to-point and other non-broadcast interfaces.

router-dead-interval <*num*>
The number of seconds not hearing a router's Hello packets before the router's neighbors will declare it down. Specify a number from 0 – 255. The default is 4 times the value of the hello interval.

poll-interval <*num*>
Before adjacency is established with a neighbor, OSPF packets are sent periodically at the specified poll interval. Specify a number equal to or greater than 1. The default value for this option is 120 seconds.

key-chain <*num-or-string*>
The identifier of the key-chain containing the authentication keys.

[authentication-method none| simple| md5]
none No authentication method associated with this interface.

simple The authentication method is a simple password in which an authentication key of up to 16 characters is included in the packet. If you choose this authentication method, you should also specify a key-chain identifier using the key-chain option.

md5 Use MD5 to create a crypto-checksum of an OSPF packet and an authentication key of up to 16 characters. If you choose this authentication method, you should also specify a key-chain identifier using the key-chain option.

Restrictions

None.

ospf set virtual-link

Purpose

Sets the parameters for an OSPF virtual link.

Format

ospf set virtual-link <*number-or-string*> **[state disable|enable] [cost** <*num*>**] [no-multicast] [retransmit-interval** <*num*>**] [transit-delay** <*num*>**] [priority** <*num*>**] [hello-interval** <*num*>**] [router-dead-interval** <*num*>**] [poll-interval** <*num*>**] [key-chain** <*num-or-string*>**] [authentication-method none| simple| md5]**

Mode

Configure

Parameters

<*number-or-string*>
The identifier for this virtual link.

state disable|enable
Enables or disables the virtual link.

cost <*num*>
The cost associated with this virtual link. The cost of all interfaces that a packet must cross to reach a destination are added to get the cost to that destination. The default cost of the OSPF interface is 1, but another non-zero value may be specified. Specify a number from 0 – 65535.

no-multicast
Instructs the X-Pedition to not send multicast packets to neighbors on point-to-point virtual links.

retransmit-interval <*num*>
The number of seconds between link state advertisement retransmissions for adjacencies belonging to this virtual link. Specify a number equal to or greater than 1.

transit-delay <*num*>
The estimated number of seconds required to transmit a link state update over this virtual link. Transit delay takes into account transmission and propagation delays and must be greater than 0. Specify a number equal to or greater than 1.

priority <*num*>
A number between 0 and 255 specifying the priority for becoming the designated router on this virtual link. When two routers attached to a network both attempt to become the designated router, the one with the higher priority wins. A router whose router priority is set to 0 is ineligible to become designated router. Specify a number from 0 – 255.

hello-interval <*num*>
The length of time, in seconds, between hello packets that the router sends on this virtual link. Specify a number from 0 – 255. The default is 60 seconds.

router-dead-interval <*num*>
The number of seconds not hearing a router's Hello packets before the router's neighbors will declare it down. Specify a number from 0 – 255. The default value for this parameter is 4 times the value of the **hello-interval** parameter.

poll-interval <*num*>
Before adjacency is established with a neighbor, OSPF packets are sent periodically at the specified poll interval. Specify a number from 0 – 255. The default is 120 seconds.

key-chain <*num-or-string*>
The identifier of the key chain containing the authentication keys.

authentication-method none| simple| md5
Specifies the authentication method used within the area. Specify one of the following:

none Do not associate an authentication method with this interface.

simple The authentication-method is a simple password in which an authentication key of up to 16 characters is included in the packet. If you choose this authentication method, you should also specify a key-chain identifier using the key-chain option.

md5 Uses the MD5 algorithm to create a crypto-checksum of an OSPF packet and an authentication key of up to 16 characters. If you choose this authentication method, you should also specify a key-chain identifier using the key-chain option.

Restrictions

None.

ospf show

Purpose

Show OSPF information.

Format

ospf show <*option-list*>

Mode

Enable

Parameters

<*option-list*>
Specifies the OSPF information you want to display. Specify one or more of the following:

all	Displays all OSPF tables.
globals	Displays OSPF globals.
timers	Displays OSPF timers.
areas	Displays OSPF areas.
interfaces	Displays OSPF interfaces.
next-hop-list	Displays valid next hop entries.
import-policies	Displays OSPF import policies.
export-policies	Displays OSPF export policies.
statistics	Displays OSPF statistics. Specify the following:

Interface Enter the interface address for the statistics you want to display. If you do not enter a specific interface address, the X-Pedition will collect statistics for all interfaces.

To-terminal Outputs the information to the terminal—the default.

To-file Saves output to the file **/cfg/gated.dmp**.

Note: The statistics for the ports specified will reinitialize if you add a previously removed interface to OSPF, add a previously removed area to OSPF (this reinitializes the statistics for that area's interfaces), create a previously removed interface, add a previously removed **ospf start** command, remove a previously added **ospf stop** command, or if you change the port status from “down” to “up.” To reset *all* port statistics, restart OSPF.

errors Displays OSPF errors. Specify the following:

Interface Enter an interface address for which errors are displayed. If you do not enter a specific interface address, the X-Pedition will collect errors for all interfaces.

To-terminal Outputs the information to the terminal—the default.

To-file Saves output to the file **/cfg/gated.dmp**.

Note: The statistics for the ports specified will reinitialize if you add a previously removed interface to OSPF, add a previously removed area to OSPF (this reinitializes the statistics for that area's interfaces), create a previously removed interface, add a previously removed "ospf start" command, remove a previously added "ospf stop" command, or if you change the port status from "down" to "up." To reset *all* port statistics, restart OSPF.

virtual-links Displays OSPF virtual links.

summary-asb Displays OSPF border routes.

as-external-ldsb Displays OSPF Autonomous System External Link State Advertisements.

exported-routes Show routes queued up to be redistributed into OSPF.

lsa Displays the Link State Advertisement. This option requires the **Area-id**, **Adv_Rtr**, and **Type**.

area-id <*IPaddr*> Specifies the OSPF area for which the query is directed.

type router-links Request the router link advertisements that describe the collected states of the router interfaces. Ls-id is set to the originating router's router-id.

type network-links Requests network link advertisements that describe the set of routers attached to the network. Ls-id is set to the IP interface address of the designated router for the network.

type summary-networks Requests summary-link advertisements describing routes to networks. Ls-id is set to the IP address of the destination network.

type summary-asbr Requests summary-link advertisements describing routes to AS boundary routers. Ls-id is set to the AS boundary router's router-id.

type as-external Requests AS external link state advertisements. Ls-id is set to the IP address of the destination network.

ls-id <*IPaddr*> Specifies the ls-id for the type of link-state advertisement requested.

adv-rtr <*IPaddr*> Requests the router ID of the router that originated this link state advertisement.

Note: **ospf show all** can be used with the following display options:

to-file <*ipaddr*>	Saves output in the file **/int-flash/cfg/gated.dmp**
to terminal	Displays output to the console (the default).

ospf start|stop

Purpose

Start or stop the OSPF protocol. OSPF is disabled by default on the X-Pedition.

Format

ospf start|stop

Mode

Configure

Parameters

start Starts OSPF.

stop Stops OSPF.

Restrictions

None.

ospf trace

Purpose

Trace OSPF.

Format

ospf trace [lsa-builds| lsa-transmit| lsa-receive spf| debug| packets {detail| send| receive}| hello {detail| send| receive}| dd {detail| send| receive}| request {detail| send| receive}| update {detail| send| receive}| ack {detail| send| receive}| local-options [all| general| state| normal| policy| task| timer| route| none]]

Mode

Enable and Configure.

Parameters

lsa-builds
Traces Link State Advertisement creation.

lsa-transmit
Traces Link State Advertisement (LSA) transmission.

lsa-receive
Traces Link State Advertisement (LSA) reception.

spf Traces Shortest Path First (SPF) calculations.

debug Traces OSPF at the debugging level of detail.

packets Traces OSPF packets.
- **detail** Show detailed information about packets.
- **send** Show OSPF packets sent by the router.
- **receive** Show OSPF packets received by the router.

hello Traces OSPF HELLO packets used to determine neighbor reachability.
- **detail** Show detailed information about hello packets.
- **send** Show OSPF hello packets sent by the router.
- **receive** Show OSPF hello packets received by the router.

dd Traces OSPF Database Description packets used in synchronizing OSPF databases.
- **detail** Show detailed information about database description packets.
- **send** Show OSPF database description packets sent by the router.
- **receive** Show OSPF database description packets received by the router.

request Traces OSPF Link State Request packets used in synchronizing OSPF databases.
- **detail** Show detailed information about Link State Request packets.

send Show OSPF Link State Request packets sent by the router.
receive Show OSPF Link State Request packets received by the router.

update Traces OSPF Link State Update packets used in synchronizing OSPF databases.
detail Show detailed information about Link State Update packets.
send Show OSPF Link State Update packets sent by the router.
receive Show OSPF Link State Update packets received by the router.

ack Traces OSPF Link State Acknowledgement packets used in synchronizing OSPF databases:
detail Show detailed information about Link State Ack packets.
send Show OSPF Link State Ack packets sent by the router.
receive Show OSPF Link State Ack packets received by the router.

local-options
Sets various trace options for this protocol only. By default, these trace-options are inherited from those specified by the **ip-router global set trace-options** command.
all Turns on all tracing.
general Turns on normal and route tracing.
state Traces state machine transitions.
normal Traces normal OSPF occurrences—abnormal OSPF occurrences are always traced.
policy Traces application of OSPF and user-specified policy to routes being imported and exported.
task Traces system interface and processing.
timer Traces timer usage.
route Traces routing table changes for routes installed.
none All tracing should be turned off.

Restrictions

Users must add the following CLI command(s) in order for tracing to work.

From Enable mode:

```
xp# ip-router set trace-state on
```

From Configure mode:

```
xp(config)# ip-router global set trace-state on
```

Chapter 45

pim Commands

The **pim** (Protocol Independent Multicast) commands allow you to dynamically build a distribution tree for forwarding multicast data on a network. For detailed information about the use of PIM, consult the *Enterasys X-Pedition User Reference Manual*.

Notes

- Because DVMRP and PIM-SM run in separate processes on the X-Pedition, current IGMP functionality may be used only with DVMRP. PIM-SM must use a separate group of commands called “PIM IGMP.”
- The X-Pedition does not allow users to enable DVMRP and PIM-SM simultaneously. If a user attempts to enable DVMRP and PIM-SM at the same time, one of the following messages will appear:

  %CLI-E-NODVMRPFAC, This command cannot be used when PIM-SM has been configured
  %CLI-E-NOPIMFAC, This command cannot be used when IGMP or DVMRP has been configured.

  To switch between PIM-SM and DVMRP you must remove the protocol's start command from the startup configuration and restart the router.
- PIM-SM will not function over SmartTRUNKs or 802.3ad aggregated links.

Command Summary

Table 36 lists the **pim** commands. The sections following the table describe the command syntax.

Table 36. pim commands

pim global start| stop
pim global set defaults [hello-interval <*sec*>**] [hello-holdtime** <*sec*>**] [hello-priority** <*sec*>**] [mrt-stale-mult** <*num*>**] [mrt-period** <*sec*>**] [assert-holdtime** <*sec*>**] [jp-interval** <*sec*>**] [jp-holdtime** <*sec*>**]**
pim global trace packets| hello| register| bootstrap| jp| assert [detail] [send] [receive]
pim global trace local-options
pim igmp start| stop
pim igmp enable interface <*intf*> **[nosend] [query-interval** <*sec*>**] [robustness** <*num*>**]**
pim igmp set [query-interval <*sec*>**] [max-resp-time** <*sec*>**] [robustness** <*num*>**]**
pim igmp trace local-options
pim igmp trace packets| query| report| leave| mtrace [detail] [send] [receive]
pim show active-rps [to-terminal| to-file]
pim show all [to-terminal| to-file]
pim show bsr [to-terminal| to-file]
pim show crp [to-terminal| to-file]
pim show errors [to-terminal| to-file]
pim show igmp-groups [to-terminal| to-file]
pim show igmp-interface all| <*IP_address*> **[to-terminal| to-file]**
pim show interface all| <*IP_address*> **[to-terminal| to-file]**
pim show neighbor all| <*IP_address*> **[to-terminal| to-file]**
pim show periodic-jp [to-terminal| to-file]
pim show route [all] [detail] [source <*IP_address*>**] [group** <*IP_address*>**] [to-terminal| to-file]**
pim show rp-hash [to-terminal| to-file]
pim show rp-set [to-terminal| to-file]
pim show timers [to-terminal| to-file]
pim sparse add crp-group <*IP-addr-netmask*> **to-component** <*name*> **[priority** <*num*>**]**

Table 36. pim commands

pim sparse add interface <*intf*> **to-component** <*name*>
pim sparse add static-rp <*IP-addr*> **group** <*IP-addr*> **to-component** <*name*>
pim sparse create component <*name*>
pim sparse set component [bsr-on| bsr-off| bsr-address <*IP-addr*>**] [bsr-period** <*sec*>**] [bsr-priority** <*num*>**] [bsr-timeout** <*sec*>**] [crp-on| crp-off| crp-address** <*IP-addr*>**] [crp-adv-period** <*sec*>**] [crp-priority** <*num*>**] [crp-holdtime** <*sec*>**] [threshold** <*num*>**] [threshold-dr** <*num*>**] [threshold-rp** <*num*>**] [reg-sup-timeout** <*sec*>**] [probe-period** <*sec*>**] [mrt-spt-mult** <*num*>**]**
pim sparse set interface [boundary] [hello-interval <*sec*>**] [hello-holdtime** <*sec*>**] [hello-priority** <*sec*>**] [assert-holdtime** <*sec*>**] [jp-interval** <*sec*>**] [jp-holdtime** <*sec*>**]**

pim global

Purpose

Start and stop pim protocol processing.

Format

pim global start| stop

Mode

Configure.

Description

The **pim global** command allows you to enable and disable PIM protocol processing. PIM will not operate until you enable it.

Parameters

start	Start pim protocol processing
stop	Stop pim protocol processing

Restrictions

- Because DVMRP and PIM-SM run in separate processes on the X-Pedition, current IGMP functionality may be used only with DVMRP. PIM-SM must use a separate group of commands called “PIM IGMP.”
- The X-Pedition does not allow users to enable DVMRP and PIM simultaneously. If a user attempts to enable DVMRP and PIM at the same time, one of the following messages will appear:

  %CLI-E-NODVMRPFAC, This command cannot be used when PIM has been configured
  %CLI-E-NOPIMFAC, This command cannot be used when IGMP or DVMRP has been configured.

  To switch between PIM and DVMRP you must remove the protocol's start command from the startup configuration and restart the router.
- PIM will not function over Q-Trunks.

Example

To enable PIM protocol processing, enter the following:

```
xp(config)# pim global start
```

pim global set defaults

Purpose

Specify defaults to apply to all PIM packets.

Format

pim global set defaults [hello-interval <*sec***>] [hello-holdtime <***sec***>] [hello-priority <***sec***>] [mrt-stale-mult** <*num*>**] [mrt-period <***sec***>] [assert-holdtime <***sec***>] [jp-interval <***sec***>] [jp-holdtime <***sec***>]**

Mode

Configure.

Description

The **pim global set defaults** command allows you to apply specific limitations and values to all PIM packets and multicast routing tables.

Parameters

[hello-interval <*sec***>]**

The length of time (in seconds) between hello packets that the router sends on its interfaces. By default, the interval is **30** seconds.

[hello-holdtime <*sec***>]**

The length of time (in seconds) that neighbors should wait for hello messages before expiring this router as a neighbor. A value of 65535 specifies that this router should never timeout as a neighbor. By default, the hold time is **105** seconds.

[hello-priority <*sec***>]**

The priority for becoming Designated Router (DR) on a multiaccess network.

[mrt-stale-mult <*num*>**]**

The number of times to examine the multicast routing table (*MRT*) before removing a stale entry. By default, the number of examination times is **14**.

[mrt-period <*sec***>]**

The number of seconds (since the last examination) to wait before examining the MRT for dead (S,G) entries (i.e., entries whose downstream list is null). The default is 15 seconds.

[assert-holdtime <*sec***>]**

The number of seconds between the time an assert is received and the time at which the assert is timed out. The default is 180 seconds.

[jp-interval <*sec*>]

The number of seconds between transmissions of a Join/Prune message. The default is **60** seconds.

[jp-holdtime <*sec*>]

The Join/Prune hold time advertised in PIM Join/Prune messages. Receivers must wait at least this long after receiving a Join/Prune message before deleting the Join/Prune state associated with the advertiser. The recommended value is **3.5 * jp-interval**. The default is 210 seconds.

Restrictions

Applies to all interfaces configured for PIM. To set defaults for a specific interface, see *pim sparse set interface* on page 740.

Example

To increase the duration between hello messages sent from this router to 60 seconds, enter the following:

```
xp(config)# pim global set hello-interval 60
```

pim global trace

Purpose

Trace all PIM, Hello, Register, Register Stop, Bootstrap, Join/Prune, and Assert packets.

Format

pim global trace packets| hello| register| bootstrap| jp| assert [detail] [send] [receive]

Mode

Configure.

Description

The **pim global trace** command allows you to trace PIM packet processing. When the X-Pedition sends or receives a packet, the router displays a message to the console.

Parameters

packets	Trace all PIM packets.
hello	Trace Hello packets.
register	Trace Register and Register Stop packets.
bootstrap	Trace Bootstrap packets.
jp	Trace Join/Prune packets.
assert	Trace Assert packets.
[detail]	Show a detailed trace message instead of a brief one.
[send]	Show only those packets that are sent from the router.
[receive]	Show only those packets received by the router.

Restrictions

Specify each trace target separately. Instead of **pim igmp trace query report detail send**, use **pim igmp trace query detail send** and **pim igmp trace report detail send**.

Example

To trace sent and received hello messages, enter the following

```
xp(config)# pim global trace hello
```

pim global trace local-options

Purpose

Sets various trace options for this protocol only. By default, these trace-options are inherited from those specified by the **ip-router global set trace-options** command.

Format

pim global trace local-options <*option-list*>

Mode

Configure.

Description

The **pim global trace local-options** command allows you to trace PIM-specific messages for those options specified by **ip-router global set trace-options**. See *ip-router global set trace-options* on page 456 for details.

Parameters

<*option-list*>

Specify which trace options you will set:

debug	Enable developer debugging options for trace.
all	Turn on all tracing.
general	Turn on normal and route tracing.
state	Trace state machine transitions in protocols.
normal	Trace normal protocol occurrences—the X-Pedition always traces abnormal occurrences.
policy	Traces the application of policy to imported and exported routes.
task	Traces system interfaces and task processing associated with this protocol or peer.
timer	Traces timer usage by this protocol or peer.
route	Traces routing table changes for routes installed by this protocol or peer.
none	Specifies that all tracing should be turned off for this protocol or peer.

Restrictions

None.

Example

```
xp(config)# pim global trace local-options timer
```

pim igmp

Purpose

Enable (or disable) IGMP protocol processing (for PIM).

Format

pim igmp start| stop

Mode

Configure.

Description

Because DVMRP and PIM-SM run in separate processes on the X-Pedition, current IGMP functionality may be used only with DVMRP. The **pim igmp** command allows you to enable or disable PIM IGMP protocol processing.

Parameters

None.

Restrictions

- The X-Pedition does not allow users to enable DVMRP and PIM simultaneously. If a user attempts to enable DVMRP and PIM at the same time, one of the following messages will appear:

  %CLI-E-NODVMRPFAC, This command cannot be used when PIM has been configured
  %CLI-E-NOPIMFAC, This command cannot be used when IGMP or DVMRP has been configured.

  To switch between PIM and DVMRP you must remove the protocol's start command from the startup configuration and restart the router.

- PIM will not function over Q-Trunks.

Example

After entering other pim-igmp commands, enter the following syntax to start IGMP (for PIM):

```
xp(config)# pim igmp start
```

pim igmp enable interface

Purpose

Enable IGMP processing for PIM on a specific interface.

Format

pim igmp enable interface <*intf*> **[nosend] [query-interval <***sec***>] [robustness** <*num*>**]**

Mode

Configure.

Description

Enable IGMP processing for PIM on a specific interface. The parameters you select will apply only to the interface specified.

Parameters

<*intf*>	The interface you will enable.
nosend	Allows the interface to *receive*—not *send*—IGMP packets.
query-interval <*sec***>**	Specifies the time (in seconds) between Host Membership Queries on this interface if the router is the Designated Querier for the subnet. The default time is 125 seconds.
robustness <*num*>	The maximum number of lost packets to allow (**2** to **65,535**) before ceasing to send additional packets. By default, this value is 2. Each time a port sends out a query and does not receive a response, it counts against the robustness value specified. When the number of lost packets equals the robustness value minus one (robustness - 1), the port will stop sending packets.

Note: *Do not* enter a 1 for the robustness value.

Restrictions

None.

Examples

To enable interface “foo” to *receive*—not *send*—IGMP packets, the following:

```
xp(config)# pim igmp enable interface foo nosend
```

To enable interface “foo2” to operate IGMP with a query-interval of 200 and to be robust to 3 packet losses, use the following syntax

```
xp(config)# pim igmp enable interface foo2 query-interval 200 robustness 4
```

pim igmp set

Purpose

Set router-wide options for IGMP.

Format

pim igmp set [query-interval <*sec***>] [max-resp-time <***sec***>] [robustness** <*num*>**]**

Mode

Configure.

Description

The **pim igmp set** command allows you to set router-wide options for IGMP. These options apply to all interfaces; however, options set through **pim igmp enable interface** take precedence.

Parameters

query-interval <*sec***>**	Specifies the time (in seconds) between Host Membership Queries if the router is the Designated Querier for a subnet. By default, the query interval is 125.
max-resp-time <*sec***>**	The longest interval (in seconds) that a group will remain in the local group database without receiving a Host Membership Report. The default is 2 x (query-interval + 10) or 270 seconds.
robustness <*num*>	The maximum number of lost packets to allow (**2** to **65,535**) before ceasing to send additional packets. By default, this value is 2. Each time a port sends out a query and does not receive a response, it counts against the robustness value specified. When the number of lost packets equals the robustness value minus one (robustness - 1), the port will stop sending packets.

Note: *Do not* enter a 1 for the robustness value.

Restrictions

None.

Example

To set the router wide query-interval to 200, and the packet-loss tolerance to 3, enter the following:

```
xp(config)# pim igmp set query-interval 200 robustness 4
```

pim igmp trace

Purpose

Set IGMP trace options.

Format

pim igmp trace packets| query| report| leave| mtrace [detail] [send] [receive]

Mode

Configure.

Description

The **pim igmp trace** command allows you to trace IGMP packet processing. When the X-Pedition sends or receives a packet, the router displays a message to the console.

Parameters

packets	Trace IGMP packets in general.
query	Trace IGMP Query packets.
report	Trace IGMP Membership Report packets.
leave	Trace IGMP Leave packets.
mtrace	Trace mtrace packets.
detail	Show a detailed trace message instead of a brief one.
send	Show only those packets that are sent from the router.
receive	Show only those packets received by the router.

Restrictions

Specify each trace target separately. Instead of **pim igmp trace query report detail send**, use **pim igmp trace query detail send** and **pim igmp trace report detail send**.

Examples

To start tracing Membership Reports sent by the router, use the following:

```
xp(config)# pim igmp trace report send
```

To trace queries received by the router in detail, enter the following syntax:

```
xp(config)# pim igmp trace query detail receive
```

pim igmp trace local-options

Purpose

Set gated trace-options specific to IGMP.

Format

pim igmp trace local-options

Mode

Configure.

Description

The **pim igmp trace local-options** command allows you to set gated trace options specific to IGMP—these options are inherited from **ip-router global set trace-options** and modified for IGMP.

Parameters

See *ip-router global set trace-options* on page 456.

Restrictions

None.

Example

To trace IGMP timer usage, enter the following:

```
xp(config)# pim igmp trace local-options timer
```

pim show active-rps

Purpose

View active group addresses and the RP that hashes to them.

Format

pim show active-rps [to-terminal| to-file]

Mode

Enable.

Description

The **pim show active-rps** command allows you to display a list of active group addresses (i.e., groups that currently have membership information) and the RP that hashes to that group.

Parameters

to-terminal	Displays the information on screen. If you do not specify how to output the information, the X-Pedition will select the to-terminal option (the default).
to-file	Writes the information to the gated dump file.

Restrictions

None.

Examples

To display the active group addresses and RP hashing information on screen, enter the following:

```
xp(enable)# pim show active-rps to-terminal
```

To write the active group addresses and RP hashing information to the gated dump file, enter the following:

```
xp(enable)# pim show active-rps to-file
```

pim show all

Purpose

Display all PIM-related information.

Format

pim show all [to-terminal| to-file]

Mode

Enable.

Description

The **pim show all** command allows you to display all interfaces, neighbors, routes, bsrs, crps, active-rps, periodic-jps, and errors.

Parameters

to-terminal	Displays the information on screen. If you do not specify how to output the information, the X-Pedition will select the to-terminal option (the default).
to-file	Writes the information to the gated dump file.

Restrictions

None.

Examples

To display all pim information on screen, enter the following:

```
xp(enable)# pim show all to-terminal
```

To write all pim information to the gated dump file, enter the following:

```
xp(enable)# pim show all to-file
```

pim show bsr

Purpose

Display information about the elected BSR and its priority.

Format

pim show bsr [to-terminal| to-file]

Mode

Enable.

Description

The **pim show bsr** command allows you to display the elected BSR and its priority. If the router is a C-BSR, this command will display the advertised address and priority.

Parameters

to-terminal	Displays the information on screen. If you do not specify how to output the information, the X-Pedition will select the to-terminal option (the default).
to-file	Writes the information to the gated dump file.

Restrictions

None.

Examples

To display bsr information on screen, enter the following:

```
xp(enable)# pim show bsr
```

To write bsr information to the gated dump file, enter the following:

```
xp(enable)# pim show bsr to-file
```

pim show crp

Purpose

Display advertised address and priority of a C-RP.

Format

pim show crp [to-terminal| to-file]

Mode

Enable.

Description

If the current router is configured to be a C-RP, the **pim show crp** command displays the router's advertised address and priority.

Parameters

to-terminal	Displays the information on screen. If you do not specify how to output the information, the X-Pedition will select the to-terminal option (the default).
to-file	Writes the information to the gated dump file.

Restrictions

None.

Examples

To display the C-RP advertised address and priority on screen, enter the following:

```
xp(enable)# pim show crp to-terminal
```

To write the C-RP advertised address and priority to the gated dump file, enter the following:

```
xp(enable)# pim show crp to-file
```

pim show errors

Purpose

Display the count of bad PIM messages.

Format

pim show errors [to-terminal| to-file]

Mode

Enable.

Description

Display the count of bad Hello and BSR messages.

Parameters

to-terminal	Displays the information on screen. If you do not specify how to output the information, the X-Pedition will select the to-terminal option (the default).
to-file	Writes the information to the gated dump file.

Restrictions

None.

Example

To display the bad message count on screen, enter the following:

```
xp(enable)# pim show errors to-terminal
```

To write the bad message count to the gated dump file, enter the following:

```
xp(enable)# pim show errors to-file
```

pim show igmp-groups

Purpose

Display IGMP group membership.

Format

pim show igmp-groups [to-terminal| to-file]

Mode

Enable.

Description

Display IGMP group memberships received from subscribers.

Parameters

to-terminal	Displays the information on screen. If you do not specify how to output the information, the X-Pedition will select the to-terminal option (the default).
to-file	Writes the information to the gated dump file.

Restrictions

None.

Example

To display IGMP group membership information on screen, enter the following:

```
xp(enable)# pim show igmp-groups
```

To write IGMP group membership to the gated dump file, enter the following:

```
xp(enable)# pim show igmp-groups to-file
```

pim show igmp-interface

Purpose

Show IGMP interfaces.

Format

pim show igmp-interface all| <*IP_address*> **[to-terminal| to-file]**

Mode

Enable.

Description

Display an interface (or all interfaces) by address configured to use IGMP protocol with PIM, along with configured values for the interface.

Parameters

all	Select the **all** keyword to display all IGMP interfaces.
<*IP_address*>	The IP address of the specific IGMP interface you wish to display.
to-terminal	Displays the information on screen. If you do not specify how to output the information, the X-Pedition will select the to-terminal option (the default).
to-file	Writes the information to the gated dump file.

Restrictions

None.

Example

To display information on screen about a specific IGMP interface, enter the following:

```
xp(enable)# pim show igmp-interface 10.250.66.3 to-terminal
```

To write information about all IGMP interfaces to the gated dump file, enter the following:

```
xp(enable)# pim show igmp-interface all to-file
```

pim show interface

Purpose

Display all interfaces or a specific interface configured to use PIM.

Format

pim show interface all| <*IP_address*> **[to-terminal| to-file]**

Mode

Enable.

Description

Display all interfaces or a specific interface (by address) configured to use PIM protocol, along with the configured values for the interface, the neighbor count, and the elected DR.

Parameters

all	Show all PIM interfaces.
<*IP_address*>	Show a specific PIM interface.
to-terminal	Displays the information on the terminal.
to-file	Writes the information to the gated dump file.

Restrictions

None.

Examples

To display on screen all interfaces configured to use PIM, enter the following:

```
xp(enable)# pim show interface all to-terminal
```

To write the a specific interface configured to use PIM to the gated dump file, enter the following:

```
xp(enable)# pim show interface 10.136.64.5 to-file
```

pim show neighbor

Purpose

Display information about PIM-configured neighbors.

Format

pim show neighbor all| *<IP_address>* **[to-terminal| to-file]**

Mode

Enable.

Description

Display a neighbor (or all neighbors) by address, the interface it belongs to, how long it has been active, when it expires, and its DR priority.

Parameters

all	Show all PIM neighbors.
<IP_address>	Show a specific PIM neighbor.
to-terminal	Displays the information on screen. If you do not specify how to output the information, the X-Pedition will select the to-terminal option (the default).
to-file	Writes the information to the gated dump file.

Restrictions

None.

Example

To display information on screen about a specific neighbor, enter the following:

```
xp(enable)# pim show neighbor 10.136.64.5 to-terminal
```

To write information about all neighbors to the gated dump file, enter the following:

```
xp(enable)# pim show neighbor all to-file
```

pim show periodic-jp

Purpose

Show pending PIM-SM join/prune message information.

Format

pim show periodic-jp [to-terminal| to-file]

Mode

Enable.

Description

Display a list of pending joins and prunes and when the next series will be sent.

Parameters

to-terminal	Displays the information on screen. If you do not specify how to output the information, the X-Pedition will select the to-terminal option (the default).
to-file	Writes the information to the gated dump file.

Restrictions

None.

Example

To display the list of joins and prunes on screen, enter the following:

```
xp(enable)# pim show periodic-jp to-terminal
```

To write the list of joins and prunes to the gated dump file, enter the following:

```
xp(enable)# pim show periodic-jp to-file
```

pim show route

Purpose

Display multicast route information.

Format

pim show route [all] [detail] [source <*IP_address*>**] [group** <*IP_address*>**] [to-terminal| to-file]**

Mode

Enable.

Description

Display multicast route information (with or without detail) for a route. Information includes the inbound interface, outbound interfaces, and flags for that route. Routes may specify a source-group pair (**source** <*IP_address*> **group** <*IP_address*>), all routes for a group (**group** <*IP_address*>), all routes for a source (**source** <*IP_address*>), or all active routes (**all**).

Parameters

[all] List all active PIM group routes and (S,G) routes.

[detail] Provide detailed information about PIM group routes and (S,G) routes.

[source <*IP_address*>**]**
Display information about a specific source. Requires **group** <*IP address*>.

[group <*IP_address*>**]**
Display information about a specific group.

to-terminal Displays the information on screen. If you do not specify how to output the information, the X-Pedition will select the to-terminal option (the default).

to-file Writes the information to the gated dump file.

Restrictions

None.

Example

To display basic multicast route information on screen for a specific group, enter the following:

```
xp(enable)# pim show route group 229.0.65.6 to-terminal
```

To write detailed multicast route information for all routes to the gated dump file, enter the following:

```
xp(enable)# pim show route all detail to-file
```

To display multicast route information on screen for a source-group pair, enter the following:

```
xp(enable)# pim show route source 10.136.64.7 group 229.0.65.6
```

pim show rp-hash

Purpose

Display a group-to-RP mapping.

Format

pim show rp-hash <*IP_address / netmask*> **[to-terminal| to-file]**

Mode

Enable.

Description

Display a group-to-RP mapping for a group and mask.

Parameters

<*IP_address / netmask*>
The IP address and netmask of the group whose information you will view.

to-terminal Displays the information on screen. If you do not specify how to output the information, the X-Pedition will select the to-terminal option (the default).

to-file Writes the information to the gated dump file.

Restrictions

None.

Example

To display all group-to-rp mappings to the terminal, enter the following:

```
xp(enable)# pim show rp-hash 224.0.0.0/4
```

To write all group-to-rp mappings for the range 229.0.65.6 to 255.255.255.0 to the gated dump file, enter the following:

```
xp(enable)# pim show rp-hash 229.0.65.6 to 255.255.255.0 to-file
```

pim show rp-set

Purpose

Show set of PIM-SM RP mappings.

Format

pim show rp-set [to-terminal| to-file]

Mode

Enable.

Description

Display list of group-to-RP mappings received from the BSR.

Parameters

to-terminal	Displays the information on screen. If you do not specify how to output the information, the X-Pedition will select the to-terminal option (the default).
to-file	Writes the information to the gated dump file.

Restrictions

None.

Example

To display a list of received group-to-RP mappings on screen, enter the following:

```
xp(enable)# pim show rp-set to-terminal
```

To write the list of received group-to-RP mappings to the gated dump file, enter the following:

```
xp(enable)# pim show rp-set to-file
```

pim show timers

Purpose

Display PIM protocol timer values.

Format

pim show timers [to-terminal| to-file]

Mode

Enable.

Description

The **pim show timers** command allows you to display values for PIM protocol timers.

Parameters

to-terminal	Displays the information on screen. If you do not specify how to output the information, the X-Pedition will select the to-terminal option (the default).
to-file	Writes the information to the gated dump file.

Restrictions

None.

Examples

To display timer values on screen, enter the following:

```
xp(enable)# pim show timers to-terminal
```

To write timer values to the gated dump file, enter the following:

```
xp(enable)# pim show timers to-file
```

pim sparse add crp-group

Purpose

Specifies a range of multicast addresses for which the router will attempt to become the RP.

Format

pim sparse add crp-group <*IP-addr-netmask*> **to-component** <*name*> **[priority** <*num*>**]**

Mode

Configure.

Description

Specifies a range of multicast addresses for which the router will attempt to become the RP. If the router is not eligible to be a CRP, you may not use this option. If you specify crp-on but do not select a group, the router will be a CRP for 224/4.

Parameters

<*IP-addr-netmask*>	The name of the group address and netmask.
to-component <*name*>	The name of the PIM-SM component.
[priority <*num*>**]**	The priority level assigned to the C-RP.

Restrictions

You must enter the following syntax before using this command.

```
xp(config)# pim sparse set component crp-on/crp-address
```

Example

To configure the router as a crp for groups in the range 228.160.x.x - 228.160/12, use the following syntax:

```
xp(config)# pim sparse add crp-group 228.160/12 to-component nyc3
```

pim sparse add interface

Purpose

Associates an interface with the PIM sparse component.

Format

pim sparse add interface <*intf*> **to-component** <*name*>

Mode

Configure.

Description

The **pim sparse add interface** command allows you to enable PIM protocol processing on a specific interface.

Parameters

interface <*intf*>	The interface you will activate.
to-component <*name*>	The name of the PIM-SM component.

Restrictions

None.

Example

To enable sparse mode on interface *foo* and add it to component *nyc3*, enter the following:

```
xp(config)# pim sparse add interface foo to-component nyc3
```

pim sparse add static-rp

Purpose

Adds a static RP-mapping to the component specified.

Format

pim sparse add static-rp <*IP-addr*> **group** <*IP-addr*> **to-component** <*name*>

Mode

Configure.

Description

The **pim sparse add static-rp** command allows you to add a static RP-mapping to the component specified. The multicast address listed will serve a specific group.

Parameters

<*IP_address*> The IP address of the port you will assign as the RP.

group <*IP_address / netmask*>
The address and netmask of the group that will map to the RP.

to-component <*name*> The name of the PIM-SM component.

Restrictions

The group IP address must be a valid multicast IP address.

Example

To set 172.16.15.3 as an RP for 226.3.3.3/32, enter the following:

```
xp(config)# pim sparse add static-RP 172.16.15.3 group 226.3.3.3/32 to-component nyc3
```

pim sparse create component

Purpose

Creates a PIM component.

Format

pim sparse create component <*name*>

Mode

Configure.

Description

The **pim sparse create component** command allows you to create a PIM component—a grouping that allows you to specify certain settings for one or more interfaces. You must define a sparse component to run PIM-SM.

Parameters

component <*name*> The name assigned to the component.

Restrictions

You may specify only one component on the router.

Example

To run sparse mode and create a component, enter the following:

```
xp(config)# pim sparse create component nyc3
```

pim sparse set component

Purpose

Sets values for a sparse-mode component.

Format

pim sparse set component [bsr-on| bsr-off| bsr-address <*IP-addr*>**] [bsr-period <***sec***>] [bsr-priority** <*num*>**] [bsr-timeout <***sec***>] [crp-on| crp-off| crp-address** <*IP-addr*>**] [crp-adv-period <***sec***>] [crp-priority** <*num*>**] [crp-holdtime <***sec***>] [threshold** <*num*>**] [threshold-dr** <*num*>**] [threshold-rp** <*num*>**] [reg-sup-timeout <***sec***>] [probe-period <***sec***>] [mrt-spt-mult** <*num*>**]**

Mode

Configure.

Description

The **pim sparse set component** command allows you to set sparse-mode-specific values to use on the router.

Parameters

bsr-on Allows router to be a candidate bootstrap router (BSR). You may select this option *only once* on any router.

bsr-off Prevent the router from becoming the BSR (default).

bsr-address <*IP-addr*>
The address to advertise as the BSR—the largest IP configured on the router (unless specifically stated otherwise). This option nullifies bsr-off and automatically selects bsr-on (i.e., if you set this option, you do not need to set bsr-on).

bsr-period <*sec***>**
The length of time (in seconds) between originating bootstrap messages. The default value for this option is 60 seconds.

bsr-priority <*num*>
BSRs with higher priorities are preferred. If two routers share the same priority, the one with the larger advertised address wins. The default priority value is 0.

bsr-timeout <*sec***>**
The length of time (in seconds) that neighbors should wait for bootstrap messages before assuming this router is unreachable. The default is 130 seconds.

crp-on Select this option to allow the router to be a candidate rendezvous point (RP).

crp-off Prevent the router from becoming the RP for any group.

crp-address <*IP-addr*>
The address advertised by the BSR is the largest IP configured on the router (unless specifically stated otherwise). This option nullifies crp-off and automatically selects crp-on (i.e., if you set this option, you do not need to set crp-on.)

crp-adv-period <*sec***>**
The length of time (in seconds) between originating CRP advertisement messages. The default value for this option is 60 seconds.

crp-priority <*num*>
CRPs with lower priorities are preferred. If two routers share the same priority, the one with the larger advertised address wins. Any group that does not have a priority assigned to it will use this priority. If you do not specify any groups, this priority will be used for 224/4.

crp-holdtime <*sec***>**
The length of time (in seconds) the BSR should use to time out CRP-Adv messages. The default is 150 seconds.

threshold <*num*>
The rate of traffic (in bytes/sec) to reach when either the DR (designated router) or the RP (rendezvous point) will switch to a shortest path tree. The default is 0 bytes/sec.

threshold-dr <*num*>
The rate of traffic (in bytes/sec) to reach when the DR (designated router) will switch to a shortest path tree. The default is 0 bytes/sec. When using the default value of zero, the DR will attempt to switch to the SPT when it receives the first packet.

threshold-rp <*num*>
The rate of traffic (in bytes/sec) to reach when the RP (rendezvous point) will switch to a shortest path tree. The default is 0 bytes/sec. When using the default value of zero, the RP will attempt to switch to the SPT when it receives the first packet.

reg-sup-timeout <*sec***>**
The mean number of seconds between receiving a register-stop message and sending registers again. A low value indicates more frequent bursts at the RP. A high value indicates a longer join latency for new receivers. The default is 60 seconds.

Note: You may lower the timeout value if you send null register messages *n* seconds (where *n* is specified in probe-period) before the timer expires. This prevents register bursts.

probe-period <*sec***>**
When you use null register messages, you can specify the number of seconds before the register-suppression timer expires to send a null register message. If a router receives a register-stop message before sending the null register message,

the register-suppression timer will reset and delay sending the null register message.

mrt-spt-mult <*num*>
The number of times to examine the MRT before trying a switch. The MRT is examined every **mrt-period** seconds (set in *pim global set defaults* on page 704).

Restrictions

None.

Examples

To enable a router as a C-BSR, enter the following:

```
xp(config)# pim sparse set component nyc3 bsr-on
```

To set component nyc3 with a specific bsr-address (172.16.3.5), enter the following:

```
xp(config)# pim sparse set component nyc3 bsr-address 172.16.3.5
```

To enable a router as a C-RP, enter the following:

```
xp(config)# pim sparse set component nyc3 crp-on
```

To set the switching threshold for the router functioning as an RP to switch to a shortest path tree once traffic flow reaches 200 bytes/sec, enter the following:

```
xp(config)# pim sparse set component nyc3 threshold-rp 200
```

To set the router functioning as a DR to switch to a shortest path tree immediately, use the following:

```
xp(config)# pim sparse set component nyc3 threshold-dr 0
```

pim sparse set interface

Purpose

Sets interface-specific options for PIM-sparse.

Format

pim sparse set interface [boundary] [hello-interval <*sec*>**] [hello-holdtime** <*sec*>**]**
[hello-priority <*sec*>**] [assert-holdtime** <*sec*>**] [jp-interval** <*sec*>**] [jp-holdtime** <*sec*>**]**

Mode

Configure.

Description

This command sets interface-specific options for PIM-sparse. See *pim global set defaults* on page 704 for definitions of these options.

Parameters

[boundary] Specifies whether the router will be a multicast border router on this interface. BSR messages do not propagate through a boundary interface.

[hello-interval <*sec*>**]**
The length of time (in seconds) between hello packets that the router sends on its interfaces. By default, the interval is **30** seconds.

[hello-holdtime <*sec*>**]**
The length of time (in seconds) that neighbors should wait for hello messages before expiring this router as a neighbor. A value of 65535 specifies that this router should never timeout as a neighbor. By default, the hold time is **105** seconds.

[hello-priority <*sec*>**]**
The priority for becoming Designated Router (DR) on a multiaccess network.

[assert-holdtime <*sec*>**]**
The number of seconds between the time an assert is received and the time at which the assert is timed out. The default is **180** seconds.

[jp-interval <*sec*>**]**
The number of seconds between transmissions of a Join/Prune message. The default is **60** seconds.

[jp-holdtime <*sec*>**]**
The Join/Prune hold time advertised in PIM Join/Prune messages. Receivers must wait at least this long after receiving a Join/Prune message before deleting the

Join/Prune state associated with the advertiser. The recommended value is **3.5 * jp-interval**. The default is 210 seconds.

Restrictions

None.

Example

To increase the duration between hello messages sent from this router to 60 seconds, enter the following:

```
xp(config)# pim sparse set hello-interval 60
```

Chapter 46

ping Command

The **ping** command tests connection between the X-Pedition and an IP host.

Format

ping <*hostname-or-IPaddr*> **packets** <*num*> **size** <*num*> **wait** <*num*> **[flood] [dontroute]**

Mode

User or Enable

Description

The **ping** command test connection between the X-Pedition and an IP host. The ping command sends ICMP echo packets to the host you specify.

- If the packets reach the host, the host sends a ping response to the X-Pedition and the CLI displays messages stating that the host can be reached.
- If the host does not respond, the X-Pedition assumes the host cannot be reached from the X-Pedition and the CLI display messages stating that the host did not reply.

Parameters

<*hostname-or-IPaddr*>
The host name or IP address you want to ping.

packets <*num*>
The number of ping packets you want to send. The default is 1.

size <*num*>
The packet size. For Ethernet, specify a number from 0 – 1364.

wait <*num*>

The number of seconds the X-Pedition will wait for a positive response from the host before assuming that the host has not responded. The default is 1.

flood

Causes the X-Pedition to send a new ping request as soon as a ping reply is received. If you do not specify the **flood** option, the X-Pedition waits to send a new request. The amount of time the X-Pedition waits is specified by the **wait** option.

dontroute

Restricts the ping to locally attached hosts.

Restrictions

If you enter this command from the User mode, the only parameter you can use is <*hostname-or-IPaddr*>. To use any of the other parameters, you must be in Enable mode.

Chapter 47

port Commands

The **port** commands set and display the following parameters:

- Port state (enabled or disabled)
- Bridging status (flow-based or address-based)
- Port operating mode (half duplex or full duplex)
- Port speed for the 10/100 ports (10-Mbps or 100-Mbps)
- Port mirroring (used for analyzing network traffic)
- Port shut down if broadcast threshold is reached

Command Summary

Table 37 lists the **port** commands. The sections following the table describe the command syntax.

Table 37. port commands

port auto-negotiate enable <*port-list*>|**disable** <*port-list*>|**restart** <*port-list*>
port bmon <*port-list*> [**redirect| unlimited-redirect**] [**rate** <*number*>] [**duration** <*number*>] [**expire** <*number*>] [**packets-limited all|broadcast**]
port description <*port-list*> <*desc*>
port disable <*port-list*>
port enable 8021p port <*port-list*>
port flow-bridging <*port-list*>|**all-ports**
port enable forced-return-flows port <*port-list*> |**all-ports**

Table 37. port commands (Continued)

port set [<*port-list*>**|all-ports] [duplex full|half] [speed 10Mbps|100Mbps|**<*number*>**] [auto-negotiation on|off] [auto-negotiation-speed 10Mbps|100Mbps|10_100Mbps] [auto-negotiation-duplex half|full|both] [auto-negotiation-flowctl off| asymmetric| symmetric| both] [hash-mode m0|m1|m2|m3|m-auto] [rx-hashmode m0|m1|m2|m3|m4|m5|m6|m7|m8|m9] [wan-encapsulation frame-relay| ppp] [ifg** <*number*>**] [input-encapsulation forced-ethernet_ii] [link-timer** <*number*>**] [clock** <*clock-source*>**] [transmit-clock-source local|loop] [framing cbit-parity|m23|esf|g832|g751] [mtu** <*number*>**] [mc-vlan-encap** <*number*>**]**
port show 8021p <*port-list*>**|all-ports**
port show autonegotiation <*port-list*>**|all-ports**
port show autonegotiation-capabilities <*port-list*>**|all-ports**
port show bmon [config][detail][port <*port list*>**][stats]**
port show bridging-status <*port-list*>**|all-ports**
port show description <*port-list*>**|all-ports**
port show MAU <*port-list*>**|all-ports**
port show MAU-statistics <*port-list*>**|all-ports**
port show port-status <*port-list*>**|all-ports| all-smarttrunks**
port show stp-info <*port-list*>**|all-ports [rstp]**
port show pvst-info <*port-list*>**|all-ports spanning-tree** <*string*> **[rstp]**
port show vlan-info <*port-list*>**|all-ports**
port show mirroring-status <*port list*>**| all-ports| all-acls**
port show hash-mode <*port-list*> **|all-ports**
port show mc-vlan-encap <*port-list*> **|all-ports**
port show serial-link-info <*port-list*> **|all-ports**

port auto-negotiate

Purpose

Enables, disables, and/or restarts auto-negotiation on a port.

Format

port auto-negotiate enable <*port-list*>|**disable** <*port-list*>|**restart** <*port-list*>

Mode

Enable

Description

The **port auto-negotiate** command allows you to enable auto-negotiation on a port, disable auto-negotiation on a port, and/or restart auto-negotiation on a port. Auto-negotiation is a process whereby both ports on a connection resolve the best line speed, duplex mode and flow control scheme to communicate with each other.

Parameters

enable <*port-list*>
Enables auto-negotiation on the port or set of ports.

disable <*port-list*>
Disables auto-negotiation on the port or set of ports.

restart <*port-list*>
Restarts auto-negotiation on the port or set of ports.

Restrictions

None.

Example

To enable auto-negotiation on port et.2.1:

```
xp# port auto-negotiate enable et.2.1
```

port bmon

Purpose

Monitor broadcast traffic on a port. This command is useful in cases where excess broadcast traffic heading toward the control module may degrade performance and you need to temporarily shut down the port.

Format

port bmon <*port-list*> [**rate** <*number*>] [**duration** <*number*>] [**expire** <*number*>] [**packet-limited all|broadcast**] [**redirect** <*port*>] [**unlimited-redirect** <*port*>]

Mode

Configure

Description

The **port bmon** command allows you to monitor the broadcast traffic on one or more ports and shut down a port if its broadcast traffic reaches and sustains a certain rate limit for a specific length of time. The **port bmon** command can also *redirect* the traffic from a channel with a monitored port to another port instead of shutting the monitored port down. The unlearned traffic for the monitor port's channel (all ports on that line card) is redirected to the target port and learned traffic flows continue to forward as they were learned. When configuring **port bmon**, you must specify a port to which you will redirect traffic if you are using the redirection feature—otherwise, the command will shut down the port by default.

With the **port bmon** command, you can define monitoring thresholds on a port or set of ports. If those thresholds are met or exceeded, the port(s) will shut down for a user-specified period. This will reduce the risk that the control module may become overloaded by traffic and crash.

Parameters

port <*port-list*>
Specifies the ports that you are monitoring for broadcasts.

rate <*number*>
The rate limit, in Kpkts per second, which will trigger a port shut down if the rate is sustained for the specified duration. Values can be from 1-1000. The default value is 10.

duration <*number*>
The number of seconds that the specified rate limit is sustained, after which the port will be shut down. Values can be from 1-3600. The default value is 1.

expire <*number*>
The number of seconds that the port will be shut down or redirected if the rate threshold is reached. Values can be from 60-36000. The default value is 300.

packets-limited all| broadcast
Specifies the type of packets to monitor for shutdown or redirect. Specify **all** to monitor all packets. Specify **broadcast** to only monitor broadcast packets. The default value is **all**.

redirect <*port*>
Specifies the port to which you will redirect traffic. The redirect option routes all unlearned traffic through another port after the current port reaches capacity. Redirect sends traffic for the number of seconds defined as the *expire* value, but waits the number of seconds defined as the *duration* before redirecting traffic. This option cannot be used with the unlimited-redirect option.

unlimited-redirect <*port*>
Specifies the port to which you will redirect traffic. The unlimited-redirect option routes all unlearned traffic through another port after the current port reaches capacity. Unlimited-redirect sends traffic *indefinitely*, but waits the number of seconds defined as the *duration* before redirecting traffic. This option cannot be used with the redirect or expire options.

Restrictions

None.

Examples

To monitor traffic on port et.1.3 and shut it down for 5 minutes if the rate of 10,000 packets per second is sustained for 1 second:

```
xp(config)# port bmon et.1.3 packets-limited all
```

To monitor traffic on port et.1.3 and shut it down for 3 minutes if the rate of 25,000 packets per second is sustained for 5 seconds:

```
xp(config)# port bmon et.1.3 rate 25 duration 5 expire 180 packets-limited all
```

To configure a 360-second expiration on port et1.3 whenever a rate of 100,000 broadcast packets per second is sustained for 1 second:

```
xp(config)# port bmon et.1.3 rate 100 expire 360 packets-limited broadcast
```

How to use redirection:

This will redirect all unlearned traffic on port et.3.2's channel to port et.4.3 for 60 seconds, one second after the 1kPkts/sec traffic limit is reached:

```
xp(config)# port bmon et.3.2 redirect et.4.3 rate 1 expire 60 packets-limited all
```

This will redirect all unlearned traffic on port et.2.2's channel to port et.3.1 indefinitely, one second after the 5kPkt/sec broadcast traffic limit is reached:

```
xp(config)# port bmon et2.2 unlimited-redirect et.3.1 rate 5
```

This will redirect all unlearned traffic on port et.1.3's channel to port et.2.3 for 100 seconds, five seconds after the 2kPkt/sec limit has been reached:

```
xp(config)# port bmon et.1.3 redirect et.2.3 expire 100 rate 2 duration 5
```

port description

Purpose

Defines a user description for a port.

Format

port description <*port-list*> <*desc*>

Mode

Configure

Description

The **port description** command allows you to define a character string description for a port. This is useful for management purposes.

Parameters

description <*port-list*>
Specifies the port(s). Valid for Ethernet and WAN ports only.

description <*desc*>
Specifies the character string used for the description of the port. This must be 125 characters or less.

Restrictions

This command is valid for Ethernet and WAN only.

Example

To set port et.2.1 with the description ‘vlan1-2’:

```
xp(config)# port description et.2.1 vlan1-2
```

port disable

Purpose

Disable a port.

Format

port disable <*port-list*>

Mode

Configure

Description

The **port disable** command disables the specified ports. Disabled ports do not send or receive any traffic. You might want to disable unused ports to prevent network users from inadvertently or unscrupulously connecting to unoccupied but enabled ports on the X-Pedition.

Parameters

port <*port-list*> Specifies the ports you are disabling.

Restrictions

None.

Examples

To disable port et.1.3 on the X-Pedition:

```
xp(config)# port disable et.1.3
```

To disable ports 1 through 5 on the Ethernet line card in slot 3 of the X-Pedition chassis:

```
xp(config)# port disable et.3.1-5
```

port enable 8021p

Purpose

Enables 802.1p encapsulation.

Format

port enable 8021p port <*port-list*>|**all-ports**

Mode

Configure

Description

The **port enable 8021p** command enables 802.1p encapsulation on the specified ports. The 802.1p standard provides the ability to classify traffic into eight priority categories or class of services. This classification scheme is based upon MAC frame information and is used for QoS (Quality of Service) for VLANs.

Parameters

port <*port-list*>|**all-ports**	Specifies the port(s) you are enabling. Specify **all-ports** to enable 802.1p encapsulation on all relevant ports

Restrictions

None.

Example

To enable 802.1p encapsulation on port et.1.3:

```
xp(config)# port enable 8021p port et.1.3
```

port flow-bridging

Purpose

Set ports to use flow-based bridging.

Format

port flow-bridging <*port-list*>|**all-ports**

Mode

Configure

Description

The **port flow-bridging** command changes the specified ports from using address-based bridging to using flow-based bridging. A port can use only one type of bridging at a time.

Each port has an L2 lookup table where MAC address or flows are stored.

- If the port is configured for address-based bridging (default), each L2 table entry consists of a MAC address and a VLAN ID.
- If the port is configured for flow-based bridging, each L2 table entry consists of a source MAC address, a destination MAC address, and a VLAN ID.

Suppose that a port on the X-Pedition is connected to a hub that is connected to three workstations, A, B, and C. If each workstation is talking to one another and sending broadcast traffic, the L2 table on the X-Pedition's port would contain the following entries for the workstations. Assume that the VLAN ID is "1" for all entries.

If the ports are configured for address-based bridging:

- MAC address A
- MAC address B
- MAC address C
- MAC broadcast address

If the ports are configured for flow-based bridging:

- MAC addresses A->B
- MAC addresses B->A
- MAC addresses B->C
- MAC addresses A->C

- MAC addresses C->A
- MAC addresses C->B
- MAC addresses A->broadcast
- MAC addresses B->broadcast
- MAC addresses C->broadcast

Parameters

<*port-list*> **| all-ports**	Specifies the ports you are changing to flow-based bridging. The keyword **all-ports** changes all the ports on the X-Pedition to flow-based bridging.

Restrictions

None.

Examples

To configure Ethernet port et.3.7 for flow-based bridging:

```
xp(config)# port flow-bridging et.3.7
```

port enabled forced-return-flows

Purpose

Enable the forced-return-flows to function on a port or list of ports.

Format

port enable forced-return-flows port <*port-list*>**|all-ports**

Mode

Configure

Description

The **port enable forced-return-flows** command allows you to perform routing without the use of a layer 3 protocol such as RIP or OSPF—these setups may involve VRRP and static gateways. In the case of VRRPs and static gateways, when a remote gateway goes down it does not notify routers more than one hop away. This can cause existing flows to restrict the flow of traffic. With forced-return-flows enabled, each new flow checks the exit port for a "reverse flow" (the source/destination IP addresses are the reverse of this new flow). If the "reverse flow" has an exit port that differs from the new flow's entry port, the exit port of the "reverse flow" changes to the new flow's entry port.

Parameters

<*port-list*> **| all-ports**	Enable forced return flows to a specific port on the port list. Use the keyword **all-ports** to specify all ports.

Restrictions

None.

Examples

```
xp(config)# port enable forced-return-flows port et.4.6
```

port set

Format

port set [<*port-list*>**|all-ports] [duplex full|half] [speed 10Mbps|100Mbps|**<*number*>**] [auto-negotiation on|off] [auto-negotiation-speed 10Mbps|100Mbps|10_100Mbps] [auto-negotiation-duplex half|full|both] [auto-negotiation-flowctl off| asymmetric| symmetric| both] [hash-mode m0|m1|m2|m3|m-auto] [rx-hashmode m0|m1|m2|m3|m4|m5|m6|m7|m8|m9] [wan-encapsulation frame-relay| ppp] [ifg** <*number*>**] [input-encapsulation forced-ethernet_ii] [link-timer** <*number*>**] [clock** <*clock-source*>**] [transmit-clock-source local|loop] [framing cbit-parity|m23|esf|g832|g751] [mtu** <*number*>**] [mc-vlan-encap** <*number*>**]**

Mode

Configure

Description

The **port set** command allows users to set the port operating mode and speed. Depending on the port's media type, users may set any of the following parameters:

Note: By default, all ports use autosensing to detect the operating mode and speed of the network segment to which they are connected. If you use this command to set a port parameter, the setting disables autosensing for that parameter on the port. For example, if you set the speed of a segment to 10-Mbps, that segment no longer uses autosensing for the port speed and will always attempt to operate at 10-Mbps.

- Ethernet ports (e.g., et.2.1)
  - Auto-negotiation (on or off, speed, and duplex)
  - Duplex (half or full)
  - Port speed
  - Hash-mode
  - Input-encapsulation
  - Inter-frame Gap (IFG)
  - Packet VLAN encapsulation
- Gigabit ports (e.g., gi.4.1)
  - Auto-negotiation (on or off and flow control)
  - Hash-mode
  - Link-timer
  - Inter-frame Gap (IFG)
  - Packet VLAN encapsulation
  - Maximum Transmit Unit (MTU)
- Ten Gig ports (e.g., xg.3.1)
  - Auto-negotiation flow control
  - hash-mode
  - Packet VLAN encapsulation

  - RX-hashmode
  - Maximum Transmit Unit (MTU)

- HSSI ports (e.g., hs.3.1)
  - Clock source
  - wan-encapsulation
  - shared-flags
  - speed
  - clock

- Serial ports (e.g., se.3.1)
  - wan-encapsulation
  - shared-flags
  - speed

- ATM ports (e.g., at.3.1)
  - Hash-mode
  - Transmit-clock-source
  - Framing (ATM ports only, no VP or VC)

- POS (Sonet) ports (e.g., so.3.1)
  - Hash-mode
  - Packet VLAN encapsulation

- FDDI ports (e.g., fi.3.1)
  - Input-encapsulation
  - Hash-mode
  - Packet VLAN encapsulation
  - Inter-frame Gap (IFG)

Parameters

<port-list>|**all-ports**
Specify the port to which you will apply the setting (supported ports include Ethernet, Gigabit, Ten Gig, ATM, POS, HSSI, Serial, Sonet, FDDI, FE, and WAN ports). The **all-ports** keyword applies the settings you select to all the X-Pedition ports.

duplex full|half
Sets the operating mode to half duplex or full duplex. This option is valid for 10/100 Mbps Ethernet only.

speed 10Mbps|100Mbps
On Fast Ethernet ports with auto-negotiation disabled, set the port speed to 10Mbps or 100Mbps using appropriate keywords, or set the speed to a number in bits/sec on WAN ports.

auto-negotiation on| off
Turns on or off auto-negotiation for Gigabit Ethernet ports. Auto-negotiation is a process whereby both ports on a connection resolve the best line speed, duplex mode and flow control scheme to communicate with each other.

auto-negotiation-speed 10Mbps| 100Mbps| 10_100Mbps
Sets the auto-negotiation speed on a Fast Ethernet port.

- **10Mbps** – Sets the auto-negotiation line speed capability advertised to 10Mbits/sec
- **100Mbps** – Sets the auto-negotiation line speed capability advertised to 100Mbits/sec
- **10_100Mbps** – Sets the auto-negotiation line speed capability advertised to 10Mbits/sec and 100Mbits/sec

auto-negotiation-duplex half|full|both
Sets the auto-negotiation duplex mode on a Fast Ethernet port.

- **half** – Sets the auto-negotiation duplex mode advertised to half-duplex.
- **full** – Sets the auto-negotiation duplex mode advertised to full-duplex.
- **both** – Sets the auto-negotiation duplex mode advertised to half-duplex and full-duplex.

auto-negotiation-flowctl off|asymmetric|symmetric|both
Sets the flow-control on a full-duplex Gigabit Ethernet port.

- **off** – Clears the flow-control capability advertised by the port.
- **asymmetric** – Sets the flow-control capability advertised to asymmetric pause.
- **symmetric** – Sets the flow-control capability advertised to symmetric pause.
- **both** – Sets the flow-control capability advertised to asymmetric pause and symmetric pause.

hash-mode m0|m1|m2|m3|m-auto
Sets the Layer-2 hash mode for this port. This hash mode defines the algorithm scheme that will be used to calculate the hash value used for the Layer-2 and Layer-3 lookup table.

The 48-bit MAC address is hashed into 8-bit groupings, represented by either B5, B4, B3, B2, B1, or B0. Assuming a MAC address of the value B5B4:B3B2:B1B0, the following describes the various hash mode and the resulting MAC address format:

- **m0** – B5B4:B3B2:B1B0
- **m1** – B5B4:B3B2:B0B1
- **m2** – B5B4:B2B3:B1B0 (default hash mode)
- **m3** – B4B5:B3B2:B1B0
- **m-auto** hashing
  Auto-hashing periodically queries the L2 or L3 tables for hash bucket overflow on a port. If the number of overflows exceed a certain threshold level, auto-hashing will automatically change the hash mode for that port. Eventually a ‘best’ hash mode for the particular traffic will be found, which will provide optimal distribution across the L2 or L3 lookup table.

The 16 bit hash index is calculated by the performing the following calculation:

(B5B4) XOR (B3B2) XOR (B1B0)

rx-hashmode m0| m1| m2| m3| m4| m5| m6| m7| m8| m9
The 10-Gbps port can distribute packets by Destination IP, Destination MAC, Source IP,

Source MAC, Destination/Source MAC, or Destination/Source IP. The distribution parameters are configurable by the end user in rx-hashmode. Users may select one of the following:

- **m0** Hash on Source IP/MAC, Destination IP/MAC, Ether type/L3 protocol, and vlan ID. This is the default hash mode.
- **m1** Hash on Source MAC, Destination MAC.
- **m2** Hash on Destination MAC.
- **m3** Hash on Source MAC.
- **m4** Hash on Source IP/MAC and Destination IP/MAC.
- **m5** Hash on Destination IP/MAC.
- **m6** Hash on Source IP/MAC.
- **m7** Hash on Source IP and Destination IP.
- **m8** Hash on Destination IP.
- **m9** Hash on Source IP.

wan-encapsulation frame-relay|ppp
Sets the encapsulation for the WAN port to either frame-relay or ppp.

ifg <*number*>
Changes the *Inter-frame Gap* (IFG) for the port by multiplying it by the <*number*> specified. The IFG values are 600-nanosecond units for 10mb connections, 40-nanosecond units for 100mb connections, and 16-nanosecond units for 1000mb connections. Possible values for this parameter are -3 through 24.

Note: When SmartTRUNKing to an Ethernet GIGASwitch product, set the IFG to 4 or greater to allow the GIGASwitch enough time to properly process incoming frames.

input-encapsulation forced-ethernet_ii
Changes the interpretation of the input MAC encapsulation to Ethernet II.

link-timer <*number*>
Sets the auto-negotiation link timer to the number of milliseconds specified by <*number*>. The <*number*> is a value between 0 and 20. This option is valid for Gigabit ports only.

clock <*clock-source*>
Sets the clock source. This parameter is applicable only when the **wan-encapsulation** parameter is specified for a HSSI port that will be connected back-to-back with a HSSI port on another router. The <*clock-source*> is one of the following values:

external-clock	External transmit clock (DCE provided)
internal-clock-51mh	Internal transmit clock at 51.84 Mhz
internal-clock-25mh	Internal transmit clock at 25.92 Mhz
external-rx-clock	External receive clock for transmit clocking

Note: For WAN cards without an internal clock, an external CSU/DSU is required.

transmit-clock-source local|loop

Sets the ATM port transmit clock source. The expected value is one of the following timing sources:

local	Selects the on board crystal oscillator as the clock source **local** is the default value
loop	Selects the receiver inputs as the clock source

Note: Do not set both ports in the same connection with a loop clock source. At least one port must be set to local clock source.

framing cbit-parity|m23|esf|g832|g751

Specifies the type of framing used by the ATM port. The expected value is one of the following framing types:

cbit-parity	Valid for T3 only
m23	Valid for T3 only
esf	Valid for T1 only
g832	Valid for E3 only
g751	Valid for E3 only

mtu <*number*>

Sets the Maximum Transmit Unit (MTU) for the port by the amount specified (64–65442). The default value depends on the port type. This parameter is not valid for ethernet ports.

mc-vlan-encap <*number*>

The X-Pedition can forward multicast packets to only one vlan on an 802.1Q trunk. To resolve this problem, all outgoing multicast traffic on a Q trunk port is redirected to **vlan** <*number*>. On the other end of the Q trunk link, a SmartSwitch 2000/6000 is used and configured with vlan classification. This allows multicast traffic to forward successfully.

Restrictions

For 10/100 Mbps Ethernet, you must set both the operating mode and the speed. You cannot set one without setting the other. For Gigabit Ethernet, you can only turn on or off auto-negotiation. You cannot set the speed or duplex for Gigabit modules.

Examples

To configure port et.1.5 to be 10 Mbps and half duplex:

```
xp(config)# port set et.1.5 speed 10mbps duplex half
```

To turn off auto-negotiation for the Gigabit port gi.4.2:

```
xp(config)# port set gi.4.2 auto-negotiation off
```

To set the Layer 2 hash mode for all ports to the m0 hash algorithm:

```
xp(config)# port set all-ports hash-mode m0
```

To set the speed for a HSSI ppp WAN port located on port 1 of slot 3:

```
xp(config)# port set hs.3.1 wan-encapsulation ppp speed 45000000
```

To set an internal clock source (25.92 Mhz) for a HSSI ppp WAN port located on port 1 of slot 3:

```
xp(config)# port set hs.3.1 wan-encapsulation ppp speed 45000000 clock internal-clock-25mh
```

To set the speed for a serial frame relay WAN port located at port 4 of slot 2, VC 100:

```
xp(config)# port set se.2.4.100 wan-encapsulation frame-relay speed 1500000
```

To increase the inter-frame gap for port et.1.1 by 400 nanoseconds (10 * 40ns):

```
xp(config)# port set ifg et.1.1 ifg 10
```

To view the RX hashmode options available for all ports, enter the following:

```
xp(config)# port set all-ports rx-hashmode ?
[rx-hashmode] requires a value of this type:
  [keyword]      - One of the following keywords:
    m0           - Hash on Source IP/MAC, Destination IP/MAC, Ether type/L3 protocol, and vlan ID.
                   This is the default hash mode.
    m1           - Hash on Source MAC, Destination MAC.
    m2           - Hash on Destination MAC.
    m3           - Hash on Source MAC.
    m4           - Hash on Source IP/MAC and Destination IP/MAC.
    m5           - Hash on Destination IP/MAC.
    m6           - Hash on Source IP/MAC.
    m7           - Hash on Source IP and Destination IP.
    m8           - Hash on Destination IP.
    m9           - Hash on Source IP.
```

To hash incoming 10-Gigabit traffic on port **xg.3.1** by Source MAC address for distribution over the module's backplane connections, enter the following:

```
xp(config)# port set xg.3.1 rx-hashmode m3
```

port show 8021p

Purpose

Displays 802.1p encapsulation status.

Format

port show 8021p <*port-list*>|**all-ports**

Mode

Enable

Description

The **port show 8021p** command displays whether 802.1p encapsulation is enabled or disabled on a port or list of ports. The 802.1p standard provides the ability to classify traffic into eight priority categories or class of services. This classification scheme is based upon MAC frame information and is used for QoS (Quality of Service) for VLANs.

Parameters

<*port-list*>|**all-ports**	Specifies the ports for which you want to display the description. The **all-ports** keyword displays the description for all the X-Pedition ports.

Restrictions

None.

Example

To display 802.1p encapsulation status for port et.2.1:

```
xp# port show 8021p et.2.1

Port        802.1p Status
 ----        -------------
 et.2.1        Disabled
```

port show autonegotiation

Purpose

Displays auto-negotiation information.

Format

port show autonegotiation <*port-list*>|**all-ports**

Mode

Enable

Description

The **port show autonegotiation** command displays auto-negotiation information. This command displays port number, administration status, current status, remote signaling, fault advertised, and fault received. Auto-negotiation is a process whereby both ports on a connection resolve the best line speed, duplex mode and flow control scheme to communicate with each other.

Parameters

<*port-list*>|**all-ports**	Specifies the ports for which you want to display the description. The **all-ports** keyword displays the description for all the X-Pedition ports.

Restrictions

None.

Example

To display auto-negotiation information for port et.2.1:

```
xp# port show autonegotiation et.2.1

              Admin    Current       Remote       Fault        Fault
Port   Status  Status       Signalling   Advertised   Received
------ -------- ------------- ------------ ------------- ----------
et.2.1  disabled  other         not detected  n/a           n/a
```

port show autonegotiation-capabilities

Purpose

Displays auto-negotiation capabilities.

Format

port show autonegotiation-capabilities <*port-list*>|**all-ports**

Mode

Enable

Description

The **port show autonegotiation-capabilities** command displays auto-negotiation capabilities. This command displays a list of port capabilities, advertised capabilities, and any received capabilities from another port. Auto-negotiation is a process whereby both ports on a connection resolve the best line speed, duplex mode and flow control scheme to communicate with each other.

Parameters

<*port-list*>|**all-ports**	Specifies the ports for which you want to display the description. The **all-ports** keyword displays the description for all the X-Pedition ports.

Restrictions

None.

Example

To display auto-negotiation capabilities for port et.2.1:

```
xp# port show autonegotiation-capabilities et.2.1

Port     Capability       Advertised       Received
------   --------------   --------------   --------------
et.2.1   other            other
         10 baseT         10 baseT
         10 baseT FD      10 baseT FD
         100 baseT4       100 baseT4
         100 baseTX       100 baseTX
         100 baseTX FD    100 baseTX FD
         100 baseT2       100 baseT2
         100 baseT2 FD    100 baseT2 FD
         Pause            Pause
         Asymetric Pause  Asymetric Pause
         Symmetric Pause  Symmetric Pause
         Asym-Sym Pause   Asym-Sym Pause
         1000 baseX       1000 baseX
         1000 baseX FD    1000 baseX FD
         1000 baseT       1000 baseT
         1000 baseT FD    1000 baseT FD
```

port show bmon

Purpose

Display broadcast monitoring information for X-Pedition ports.

Format

port show bmon [config][detail][port <*port list*>**][stats]**

Mode

Enable

Description

The **port show bmon** command lets you display broadcast monitoring information for X-Pedition ports.

Parameters

If no parameters are specified, the current states of all ports are displayed.

config	Displays configuration information for broadcast monitoring.
detail	Displays all information for broadcast monitoring.
port <*port-list*>	Specifies the ports for which you want to display information.
stats	Displays statistics information for broadcast monitoring.

Restrictions

None.

Example

To display the state of ports with broadcast monitoring:

```
xp# port show bmon
Port: et.1.1 State: On

Port: et.6.8 State: ShutDn Shutdown: 39 (sec)

Port: et.7.8 State: On
```

The above example shows three ports, with the port et.6.8 shut down for 39 seconds.

To display broadcast monitoring configuration values set for the ports:

```
xp# port show bmon config
Port: et.1.1 Rate (Kpps): 10 Burst (sec): 1 Shutdown (sec):300

Port: et.6.8 Rate (Kpps): 10 Burst (sec): 5 Shutdown (sec):60

Port: et.7.8 Rate (Kpps): 2 Burst (sec): 2 Shutdown (sec):60
```

In the above example, port et.1.1 has been configured with default values.

To display broadcast monitoring statistics for the ports:

```
xp# port show bmon stats
Port: et.1.1 Current Broadcast Rate (Kpps): 0.000

Port: et.6.8 Burst at port shutdown (Kpps): 10.032
ShutDn Count: 2

Port: et.7.8 Current Broadcast Rate (Kpps): 0.000
```

In the above example, the current broadcast traffic on et.1.1 and et.7.8 is zero. The port et.6.8 is currently shut down and it shows a burst of 10.032K packets per second at its shut down. This port has been shut down twice because of excessive broadcast traffic.

To show broadcast monitoring details for the ports:

```
xp# port show bmon detail
Port: et.1.1 Rate (Kpps): 10 Burst (sec): 1 Shutdown (sec):300
State: On
Current Broadcast Rate (Kpps): 0.000

Port: et.6.8 Rate (Kpps): 10 Burst (sec): 5 Shutdown (sec):60
State: ShutDn Expire: 39 (sec)
Burst at port shutdown (Kpps): 10.032
ShutDn Count: 2

Port: et.7.8 Rate (Kpps): 2 Burst (sec): 2 Shutdown (sec):60
State: On
Current Broadcast Rate (Kpps): 0.000
```

The above example shows configuration, state, and statistics information.

port show bridging-status

Purpose

Display the bridging status of X-Pedition ports.

Format

port show bridging-status <*port-list*>|**all-ports**

Mode

Enable

Description

The **port show bridging-status** command lets you display bridging-status information for X-Pedition ports.

Parameters

<*port-list*>|**all-ports**	Specifies the ports for which you want to display information. The **all-ports** keyword displays the selected information for all the X-Pedition ports.

Restrictions

None.

Example

To display the bridging status for all available ports:

```
xp# port show bridging-status all-ports
Port       Mgmt Status  phy-state     link-state Bridging Mode
----       -----------  ---------     ---------- -------------
et.4.1     No Action    Disabled      Link Down  Address
et.4.2     No Action    Disabled      Link Down  Address
et.4.3     No Action    Forwarding    Link Up    Address
et.4.4     No Action    Disabled      Link Down  Address
et.4.5     No Action    Disabled      Link Down  Address
et.4.6     No Action    Forwarding    Link Up    Address
et.4.7     No Action    Disabled      Link Down  Address
et.4.8     No Action    Disabled      Link Down  Address
```

port show description

Purpose

Display the user defined descriptions of X-Pedition ports.

Format

port show description <*port-list*>|**all-ports**

Mode

Enable

Description

The **port show description** command lets you display the user defined description for X-Pedition ports. The description is defined using the **port description** command.

Parameters

<*port-list*>|**all-ports**	Specifies the ports for which you want to display the description. The **all-ports** keyword displays the description for all the X-Pedition ports.

Restrictions

This command is valid for Ethernet and WAN only.

Example

To display the bridging status for all available ports:

```
xp# port show description et.2.1

Port Name    Description
---------    -----------
et.2.1       vlan1-2
```

port show MAU

Purpose

Displays Media Access Control information.

Format

port show MAU <*port-list*>|**all-ports**

Mode

Enable

Description

The **port show MAU** command displays Media Access Control (MAC) information. This command displays port number, media type, default media type, jack type, operational status, and support level.

Parameters

<*port-list*>|**all-ports**	Specifies the ports for which you want to display the description. The **all-ports** keyword displays the description for all the X-Pedition ports.

Restrictions

None.

Example

To display MAC information for port et.2.1:

```
xp# port show MAU et.2.1

Port     MUA Type         Default Type     Jack Type   Status Supported
------   --------------   --------------   ---------   -----------   ---
et.2.1   100 BaseFX HD    100 BaseFX HD    fiber SC    operational   no
```

port show MAU-statistics

Purpose

Displays Media Access Control statistics.

Format

port show MAU-statistics <*port-list*>|**all-ports**

Mode

Enable

Description

The **port show MAU-statistics** command displays Media Access Control (MAC) statistics. This command displays port number, media availability, media availability state exits totals, jabber (excessively long frames) state, jabbering state enters totals, and false carriers totals.

Parameters

<*port-list*>|**all-ports**	Specifies the ports for which you want to display the description. The **all-ports** keyword displays the description for all the X-Pedition ports.

Restrictions

None.

Example

To display MAC statistics for port et.2.1:

```
xp# port show MAU-statistics et.2.1

                                        Media Avail.  Jabber     Jabbering     False
Port    Media Avail.      State Exits   State      State Enters Carriers
------  ----------------  ------------  ---------  ------------ ----
et.2.1  not available     0             other      0            0
```

port show port-status

Purpose

Display various information about specified ports.

Format

port show port-status <*port-list*/SmartTRUNK-*list*>**|all-ports|all-smarttrunks**

Mode

Enable

Description

The **port show port-status command** lets you display port-status information for X-Pedition ports or SmartTRUNKs.

Parameters

<*port-list*/SmartTRUNK-*list*>**|all-ports|all-smarttrunks**

Specifies the LAN/WAN ports or SmartTRUNKs for which you want to display status information. The **all-ports** keyword displays information for all the X-Pedition ports. The **all-smarttrunks** keyword displays information for all SmartTRUNKs.

Restrictions

This command does not show Virtual Circuit (VC) information. To see the state of sub-interfaces, you need to use the appropriate facility command, such as the **frame-relay show stats** command.

Example

To display the port status for all ports on Ethernet module 1 (et.1):

```
xp# port show port-status et.1.*

Flags: M - Mirroring enabled  S - SmartTRUNK port

                                                  Link  Admin
Port     Port Type            Duplex Speed      Negotiation State State Flags
----     ---------            ------ -----      ----------- ----- ----- -----
et.1.1   10/100-Mbit Ethernet   Half   10 Mbits   Manual      Up    Up
et.1.2   10/100-Mbit Ethernet   Half   10 Mbits   Manual      Up    Up
et.1.3   10/100-Mbit Ethernet   Half   10 Mbits   Manual      Up    Up
et.1.4   10/100-Mbit Ethernet   Half   10 Mbits   Manual      Up    Up
et.1.5   10/100-Mbit Ethernet   Half   10 Mbits   Manual      Up    Up
et.1.6   10/100-Mbit Ethernet   Half   10 Mbits   Manual      Up    Up
et.1.7   10/100-Mbit Ethernet   Half   10 Mbits   Manual      Up    Up
et.1.8   10/100-Mbit Ethernet   Half   10 Mbits   Manual      Up    Up
```

port show stp-info

Purpose

Display Spanning Tree (STP) information for X-Pedition ports.

Format

port show stp-info <*port-list*>|**all-ports [rstp]**

Mode

Enable

Description

The **port show stp-info** command lets you display Spanning-Tree information for X-Pedition ports.

Parameters

<*port-list*>|**all-ports**	Specifies the ports for which you want to display information. The **all-ports** keyword displays the selected information for all the X-Pedition ports.
rstp	Display RSTP-related information.

Restrictions

None.

Example

To display the spanning tree information for all available ports:

```
xp# port show stp-info all-ports
                                                                    Designated
Port     Priority  Cost    STP     State          Designated-Bridge      Port
----     --------  ----    ---     -----          -----------------      ----------
et.2.1     008   00001   Enabled   Forwarding   0064:00e06336b60e      8 011
et.2.2     000   00000   Disabled  Disabled     0000:000000000000      0 000
et.2.3     000   00000   Disabled  Disabled     0000:000000000000      0 000
et.2.4     000   00000   Disabled  Disabled     0000:000000000000      0 000
et.2.5     000   00000   Disabled  Disabled     0000:000000000000      0 000
et.2.6     008   00010   Enabled   Forwarding   07d0:00e0633680ce      8 026
et.2.7     008   00010   Enabled   Forwarding   07d0:00e0633680ce      8 027
et.2.8     008   00010   Enabled   Forwarding   07d0:00e0633680ce      8 028
gi.3.1     008   00001   Enabled   Blocking     03e8:00e06334eb4e      8 035
gi.3.2     008   00001   Enabled   Blocking     03e8:00e06334eb4e      8 035
```

```
xp# port show stp-info all-ports rstp
                      Point To Point Status    Edge Port Status
Port     Port Role    (Admin/Oper)             (Admin/Oper)        Sends RSTP?
----     ---------    ---------------------    ----------------    -----------
et.2.1   Root          Auto/True               False/False          Yes
et.2.2   Disabled      Auto/False              False/False          Yes
et.2.3   Disabled      Auto/False              False/False          Yes
et.2.4   Disabled      Auto/False              False/False          Yes
et.2.5   Disabled      Auto/False              False/False          Yes
et.2.6   Designated    Auto/True               True/True            Yes
et.2.7   Designated    Auto/True               True/True            Yes
et.2.8   Designated    Auto/True               True/True            Yes
gi.3.1   Alternate     Auto/True               False/False          Yes
gi.3.2   Alternate     Auto/True               False/False          Yes
```

port show pvst-info

Purpose

Display Spanning Tree (STP) information for a particular spanning tree.

Format

port show pvst-info <*port-list*>|**all-ports spanning-tree** <*string*> **[rstp]**

Mode

Enable

Description

The **port show pvst-info** command lets you display Spanning-Tree information for a particular spanning tree.

Parameters

<*port-list*>|**all-ports**	Specifies the ports for which you want to display information. The **all-ports** keyword displays the selected information for all the X-Pedition ports.
<*string*>	Specifies the name of the spanning tree for which you want to display information.
rstp	Display RSTP-related information.

Restrictions

None.

Example

To display the spanning tree information for spanning tree on all ports:

```
xp# port show pvst-info all-ports spanning-tree red
                                                                    Designated
Port     Priority  Cost     STP      State          Designated-Bridge      Port
----     --------  ----     ---      -----          -----------------      ----------
et.7.1     008     00010    Enabled  Forwarding     8000:00001d17ed21      8 071
et.7.2     000     00000    Enabled  Disabled       0000:000000000000      0 000
et.7.3     008     00010    Enabled  Blocking       8000:00e0630457c0      8 005
et.7.4     008     00010    Enabled  Blocking       8000:00e0630457c0      8 006
```

```
xp# port show pvst-info all ports spanning-tree red rstp

                        Point To Point Status    Edge Port Status
Port      Port Role       (Admin/Oper)            (Admin/Oper)     Sends RSTP?
----      ---------     ---------------------     ---------------  -----------
et.7.1    Root          Auto/True                 False/False      Yes
et.7.2    Disabled      Auto/False                False/False      Yes
et.7.3    Alternate     Auto/True                 False/False      Yes
et.7.4    Alternate     Auto/True                 False/False      Yes
```

port show vlan-info

Purpose

Display VLAN information for X-Pedition ports.

Format

port show vlan-info <*port-list*>|**all-ports**

Mode

Enable

Description

The **port show vlan-info** command lets you display VLAN information about X-Pedition ports.

Parameters

<*port-list*>|**all-ports**	Specifies the ports for which you want to display information. The **all-ports** keyword displays the selected information for all the X-Pedition ports.

Restrictions

None

Example

To display the VLAN information for all available ports:

```
xp# port show vlan-info all-ports
Port       Access Type    IP VLANs    IPX VLANs    Bridging VLANs
------     ---------      ---------   --------     --------------
et.4.1     access         DEFAULT     DEFAULT      DEFAULT
et.4.2     access         DEFAULT     DEFAULT      DEFAULT
et.4.3     access         DEFAULT     DEFAULT      DEFAULT
et.4.4     access         DEFAULT     DEFAULT      DEFAULT
et.4.5     access         DEFAULT     DEFAULT      DEFAULT
et.4.6     access         DEFAULT     DEFAULT      DEFAULT
et.4.7     access         DEFAULT     DEFAULT      DEFAULT
et.4.8     access         DEFAULT     DEFAULT      DEFAULT
```

port show mirroring-status

Purpose

Show the port mirroring status for ports in the X-Pedition chassis.

Format

port show mirroring-status <*port list*>**| all-ports| all-acls**

Mode

Enable

Description

The **port show mirroring-status** command shows the following port mirroring status information for the specified chassis ports:

- Whether port mirroring is enabled
- The ports or acls that are being mirrored
- The mirroring mode (input port, output port, or both)

Parameters

<*port list*>	List of Ethernet ports or WAN modules. For example: et.1.3,hs.3,et.2.(1-3),se.4
all-ports	Display information for all ports.
all-acls	Display information for all flow mirroring rules.

Restrictions

None.

Example

To display the port mirroring status for port 5:

```
xp(config)# port show mirroring-status 5
```

port show hash-mode

Purpose

Displays the Layer 2 hash mode for a particular port(s).

Format

port show hash-mode <*port-list*>|**all-ports**

Mode

Enable

Description

The **port show hash-mode** command displays the Layer 2 hash mode used by a particular port(s). An example is displayed to show the resulting MAC address format by using this hash mode. See *port set* on page 757 for a description of all hash modes.

Parameters

<*port-list*>|**all-ports**	Specifies the ports for which you will display the description. Use the **all-ports** keyword to display all X-Pedition ports.

Restrictions

None.

Examples

To display the hash mode for port et.2.1:

```
xp# port show hash-mode et.2.1

L2 Port Hash Mode (assume a MAC address = 0011:2233:4455
--------------------------
Port et.2.1          Mode-2          0011_3322_4455
```

port show mc-vlan-encap

Purpose

Displays the vlan ID where an outbound multicast packet on an 802.1q trunk port will be redirected.

Format

port show mc-vlan-encap <*port-list*>|**all-ports**

Mode

Enable

Description

The **port show mc-vlan-encap** command displays which vlan(s) to redirect the outbound multicast traffic to on a port or a list of ports. The X-Pedition can forward multicast packets to only one vlan on an 802.1Q trunk. To resolve this problem, all outgoing multicast traffic on a Q trunk port is redirected to **vlan** <*number*>. On the other end of the Q trunk link, a SmartSwitch 2000/6000 is used and configured with vlan classification. This allows multicast traffic to forward successfully.

Parameters

<*port-list*>|**all-ports**	Specifies the ports for which you will display the description. Use the **all-ports** keyword to display all X-Pedition ports.

Restrictions

This command applies only to Q-trunk ports.

Examples

To display the hash mode for port et.2.1:

```
xp# port show mc-vlan-encap et.4.6
Port et.4.6          MC Encapsulation Vlan: 10
```

port show serial-link-info

Purpose

Displays the serial link information for X-Pedition ports.

Format

port show serial-link-info <*port-list*>|**all-ports**

Mode

Enable

Description

The **port show serial-link-info** command lets you display the status of the DTE-DCE control signals for a particular port(s).

Parameters

<*port-list*>|**all-ports** Specifies the port(s) for which you will display all serial-port information. Use the **all-ports** keyword to display all X-Pedition ports.

Restrictions

This command applies only to WAN ports.

Examples

```
xp# port show serial-link-info all-ports

Port      Port Type      CD    CTS     RTS/DTR      DSR    LL      LINK
----    -------------    --    ---    ----------    ----   ---     -----
se.5.1    V.35           on    on       on          on     off     on
se.5.2    V.35           on    on       on          on     off     on
se.5.3    WAN Serial     off   off      off         off    off     off
se.5.4    WAN Serial     off   off      off         off    off     off
```

CD Carrier Detected
CTS Clear To Send
RTS/DTR Request To Send/Data Terminal Ready
DSR Data Set Ready
LL Local Loopback

Chapter 48

port mirroring Command

port mirroring

Purpose

Apply port mirroring to a target port on an X-Pedition or to traffic specified by an ACL profile.

Format

port mirroring dst-ports <*port_list*> **[src-ports** <*port_list*>|**src-acl** <*acl name*>**]**

Mode

Configure

Description

The **port mirroring** command can be used to duplicate traffic from a single port to another single port, a single port to multiple ports, multiple ports to a single port, or multiple ports to multiple ports. You may also use the Port Mirroring facility in conjunction with an ACL. When you set up a mirror for traffic coming into the X-Pedition that matches a specific ACL, the X-Pedition mirrors the traffic out to one or more ports.

Parameters

dst-ports <*port_list*>
The port(s) you will use to monitor activity. This is the port or port list to which you will want to connect the traffic sniffer.

src-ports <*port_list*>
The port(s) for which you want to monitor activity. You can specify any ports. Traffic will be mirrored from these ports to the monitor port.

src-acl <*acl name*>
The name of the ACL that specifies the profile of the traffic that you want to monitor. The ACL must be a previously created IP ACL. The ACL may contain either **permit** or **deny** keywords. The **port mirroring** command only looks at the following ACL rule parameter values: protocol, source IP address, destination IP address, source port, destination port, and TOS.

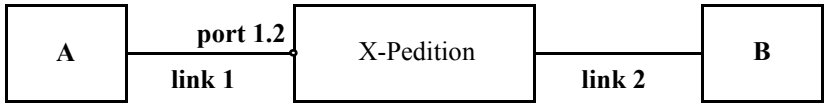
Restrictions

- Enterasys recommends that you monitor Gigabit ports through other Gigabit ports—you will almost certainly experience speed-inconsistency-related problems monitoring a Gigabit port through a 10Base-T or 100Base-TX port.
- When you enable L4-bridging on a mirrored ACL, the X-Pedition will mirror only established unicast flows and ACL denied or dropped flows that match the given ACL.
- The X-Pedition cannot mirror traffic from an ATM interface.
- ACLs and ports cannot be mirrored simultaneously.
- You may define up to 10 port mirrors via a maximum of 10 port mirroring commands and 128 ACL mirrors, but the actual limit will depend on the resources available.
- Packets that are lost due to CRC and BUFFER_OVERFLOW errors are not mirrored to the destination port. If ACL mirroring is configured, packets dropped due to unresolved ARPs, rejected routes, and ICMP packets generated by the X-Pedition in response to these will not be mirrored.
- Packets on mirrored Q-Trunk ports will not carry the IEEE 802.1Q tag header.
- Because the X-Pedition mirrors packets after they are routed, the mirrored outputs for routed packets will appear to have the same destination MAC addresses as the egress flows, even when mirroring the ingress ports. Additionally, due to hardware limitation, these mirrored outputs will list the X-Pedition's system MAC address as their source rather than the MAC address associated with the ingress or egress interface.

Note: This restriction does not apply to packets switched within the same VLAN.

For example, routed packets from source A to destination B on link 2 in the diagram below are seen leaving the X-Pedition even when port 1.2 is being monitored. The mirrored output will

list the X-Pedition's system MAC address as its source, even if you create the link 1 and link 2 interfaces with different MAC addresses.



Examples

To mirror traffic on ethernet ports et.2.2 to port et1.2:

```
xp(config)# port mirroring dst-ports et.1.2 src-ports et.2.2
```

After configuring et.1.2 as a monitor-port, et.1.2 is unusable for any other function in the system. This is indicated by a A LINK_DOWN message. However, et.1.2 is capable of transmitting packets and its LED will be lit while in operation.

To mirror traffic that is specified by the profile in the ACL “101” to port et1.2:

```
xp(config)# port mirroring dst-ports et.1.2 src-acl 101
```

Chapter 49

ppp Commands

The **ppp** commands allow you to define Point-to-Point Protocol (PPP) service profiles, and specify and monitor PPP High-Speed Serial Interface (HSSI) and standard serial ports.

Command Summary

Table 38 lists the **ppp** commands. The sections following the table describe the command syntax.

Table 38. ppp commands

ppp add-to-mlp <*mlp*> **ports** <*port list*>
ppp apply service <*service name*> **ports** <*port list*>
ppp clear stats-counter [frame-drop-qdepth-counter] [max-frame-enqued-counter] [frame-drop-red-counter] [rmon] ports <*port list*>
ppp create-mlp <*mlp list*> **slot** <*number*>
ppp define service <*service name*> **[bridging enable|disable] [high-priority-queue-depth** <*number*>**] [ip enable|disable] [ipx enable|disable] [lcp-echo on|off] [lcp-magic on|off] [low-priority-queue-depth** <*number*>**] [max-configure** <*number*>**] [max-failure** <*number*>**] [max-terminate** <*number*>**] [med-priority-queue-depth** <*number*>**] [red on|off] [red-maxTh-high-prio-traffic** <*number*>**] [red-maxTh-low-prio-traffic** <*number*>**] [red-maxTh-med-prio-traffic** <*number*>**] [red-minTh-high-prio-traffic** <*number*>**] [red-minTh-low-prio-traffic** <*number*>**] [red-minTh-med-prio-traffic** <*number*>**] [retry-interval** <*number*>**]**
ppp restart lcp-ncp ports <*port list*>
ppp set mlp-encaps-format ports <*port list*> **[format short-format]**
ppp set mlp-frag-size ports <*port list*> **[size** <*size*>**]**
ppp set mlp-fragq-depth ports <*port list* > **qdepth** <*number-of-packets*>

Table 38. ppp commands (Continued)

ppp set mlp-orderq-depth ports <*port list* > **qdepth** <*number-of-packets*>
ppp set payload-compress [max-histories 0|1] [type stac] ports <*port list*>
ppp set payload-encrypt [type des-bis] ports <*port list*>
ppp set peer-addr [ip-address <*IP address*>**] [ipx-address** <*IPX address*>**] [ports** <*port list*>**]**
ppp set ppp-encaps-bgd ports <*port list*>
ppp show mlp <*mlp list*>**|all-ports**
ppp show service <*service name*>**|all**
ppp show stats ports <*port*> **[bridge-ncp] [ip-ncp] [link-status] [summary]**

ppp add-to-mlp

Purpose

Add PPP ports to an MLP bundle.

Format

ppp add-to-mlp <*mlp*> **port** <*port list*>

Mode

Configure

Description

The **ppp add-to-mlp** command allows you to add one or more PPP ports to a previously defined MLP bundle.

Parameters

<*mlp*>	The name of the previously defined MLP bundle.
<*port list*>	The WAN port(s) you want to add to the MLP bundle.

Restrictions

Usage is restricted to PPP WAN ports only.

Example

To add the port “hs.3.1” to the MLP bundle “mp.1”:

```
xp(config)# ppp add-to-mlp mp.1 port hs.3.1
```

ppp apply service

Purpose

Apply a pre-defined service profile to an interface.

Format

ppp apply service <*service name*> **ports** <*port list*>

Mode

Configure

Description

Issuing the **ppp apply service ports** command allows you to apply a previously defined service profile to a given PPP WAN port.

Parameters

<*service name*>	The name of the previously defined service you wish to apply to the given port(s) or interfaces.
<*port list*>	The port(s) to which you wish to apply the pre-defined service profile. You can specify a single port or a comma-separated list of ports.

Restrictions

Usage is restricted to PPP WAN ports only.

Example

To apply the service “s1” to slot 2, serial ports 1 and 2:

```
xp(config)# ppp apply service s1 ports se.2.1,se.2.2
```

ppp clear stats-counter

Purpose

Clears the specified statistics counter.

Format

ppp clear stats-counter [frame-drop-qdepth-counter] [max-frame-enqued-counter] [frame-drop-red-counter] [rmon] ports <*port list*>

Mode

Enable

Description

The **ppp clear stats-counter** command allows you to specify a particular statistic counter and have the statistics reset to zero. There are statistic counters on each PPP WAN port, and you can use the **ppp clear stats-counter** to clear the counter for an individual WAN port or for a group of ports.

Parameters

frame-drop-qdepth-counter	Specify this optional parameter to reset the frame drop counter to zero.
max-frame-enqued-counter	Specify this optional parameter to reset the max enqued frames counter to zero.
frame-drop-red-counter	Specify this optional parameter to reset the packet drop counter to zero.
rmon	Specify this optional parameter to reset the rmon counter to zero.
<*port list*>	The WAN port(s) that you wish to clear the counter.

Restrictions

Usage is restricted to PPP WAN ports only.

Example

To clear the frame drop counter to zero on WAN port hs.3.1:

```
xp# ppp clear stats-counter frame-drop-qdepth-counter ports hs.3.1
```

ppp create-mlp

Purpose

Create MLP bundles.

Format

ppp create-mlp <*mlp list*> **slot** <*number*>

Mode

Configure

Description

The **ppp create-mlp** command allows you to create one or more MLP bundles.

Parameters

<*mlp list*>	The name(s) of the MLP bundles you want to create. You can specify a single bundle or a comma-separated list of MLP bundles.
<*slot*>	The slot number for the MLP bundle(s).

Restrictions

Usage is restricted to PPP WAN ports only.

Example

To create the MLP bundle “mp.1” for slot 1:

```
xp(config)# ppp create-mlp mp.1 slot 1
```

ppp define service

Purpose

Define a service profile for WAN ports.

Format

ppp define service <*service name*> **[bridging enable|disable] [high-priority-queue-depth** <*number*>**] [ip enable|disable] [ipx enable|disable] [lcp-echo on|off] [lcp-magic on|off] [low-priority-queue-depth** <*number*>**] [max-configure** <*number*>**] [max-failure** <*number*>**] [max-terminate** <*number*>**] [med-priority-queue-depth** <*number*>**] [red on|off] [red-maxTh-high-prio-traffic** <*number*>**] [red-maxTh-low-prio-traffic** <*number*>**] [red-maxTh-med-prio-traffic** <*number*>**] [red-minTh-high-prio-traffic** <*number*>**] [red-minTh-low-prio-traffic** <*number*>**] [red-minTh-med-prio-traffic** <*number*>**] [retry-interval** <*number*>**]**

Mode

Configure

Description

The **ppp define service** command allows you to specify the following attributes for a newly created service profile:

- Activate and deactivate bridging, IP, and/or IPX for PPP WAN ports. If you do not specify any bridging, IP, or IPX protocols for PPP WAN ports, they are all activated by default. If you specify a bridging, IP, or IPX protocol, you *must* also explicitly define the behavior of the other two (i.e., **enabled** or **disabled**).
- The allowable PPP queue depth for high-, low-, and medium-priority items.
- Enable and disable the sending of LCP Echo Request messages. LCP Echo Requests and their corresponding LCP Echo Responses determine if a link to a peer is down.
- Enable and disable the use of LCP magic numbers. Magic numbers are used to help detect loopback conditions.
- The maximum allowable number of unanswered/improperly answered configuration requests before determining that the connection to the peer is lost.
- The maximum allowable number of negative-acknowledgment responses for a given interface before declaring an inability to converge.
- The maximum allowable unacknowledged terminate requests before determining that the peer is unable to respond.
- Activate or deactivate Random Early Discard (RED) for PPP ports.

- The maximum and minimum threshold values for RED high-, low-, and medium-priority traffic.

  In general, Enterasys recommends that the maximum threshold values be less than or equal to the respective high-, low-, or medium-priority queue depth. The minimum threshold values should be one-third of the respective maximum threshold.

- The number of seconds that will pass before a subsequent "resending" of the configuration request will be transmitted.

Parameters

<service name>
The name you wish to assign to the newly created service profile.

bridging enable|disable
Specifying the **enable** keyword activates bridging for PPP WAN ports. Specifying the **disable** keyword deactivates bridging for PPP WAN ports.

high-priority-queue-depth *<number>*
The number of items allowed in the PPP queue. You can specify a number between 1 and 65,535. Enterasys recommends a value within the 5 - 100 item range. The default value is 20.

ip enable|disable
Specifying the **enable** keyword activates IP for PPP WAN ports. Specifying the **disable** keyword deactivates IP for PPP WAN ports.

ipx enable|disable
Specifying the **enable** keyword activates IPX for PPP WAN ports. Specifying the **disable** keyword deactivates IPX for PPP WAN ports.

lcp-echo on|off
Specifying the **on** keyword enables the sending of LCP Echo Request messages. Specifying the **off** keyword disables the sending of LCP Echo Request messages. The sending of LCP Echo Requests is enabled by default.

lcp-magic on|off
Specifying the **on** keyword enables the use of LCP magic numbers. Specifying the **off** keyword disables the use of LCP magic numbers. The use of LCP magic numbers is enabled by default.

low-priority-queue-depth *<number>*
The number of items allowed in the PPP queue. You can specify a number between 1 and 65,535. Enterasys recommends a value within the 5 - 100 item range. The default value is 20.

max-configure *<number>*
The maximum allowable number of unanswered requests. You can specify any number greater than or equal to 1. The default value is 10.

max-failure *<number>*
The maximum allowable number of negative-acknowledgment transmissions. You can specify any number greater than or equal to 1. The default value is 5.

max-terminate <*number*>
The maximum allowable number of unanswered/improperly answered connection-termination requests before declaring the link to a peer lost. You can specify any number greater than or equal to 1. The default value is 2.

med-priority-queue-depth <*number*>
The number of items allowed in the PPP queue. You can specify a number between 1 and 65,535. Enterasys recommends a value within the 5 - 100 item range. The default value is 20.

red on|off
Specifying the **on** keyword enables RED for PPP WAN ports. Specifying the **off** keyword disables RED for PPP WAN ports.

red-maxTh-high-prio-traffic <*number*>
The maximum allowable threshold for high-priority RED traffic. You can specify a number between 1 and 65,535. The default value is 12.

red-maxTh-low-prio-traffic <*number*>
The maximum allowable threshold for low-priority RED traffic. You can specify a number between 1 and 65,535. The default value is 12.

red-maxTh-med-prio-traffic <*number*>
The maximum allowable threshold for medium-priority RED traffic. You can specify a number between 1 and 65,535. The default value is 12.

red-minTh-high-prio-traffic <*number*>
The minimum allowable threshold for high-priority RED traffic. You can specify a number between 1 and 65,535. The default value is 4.

red-minTh-low-prio-traffic <*number*>
The minimum allowable threshold for low-priority RED traffic. You can specify a number between 1 and 65,535. The default value is 4.

red-minTh-med-prio-traffic <*number*>
The minimum allowable threshold for medium-priority RED traffic. You can specify a number between 1 and 65,535. The default value is 4.

retry-interval <*number*>
The number of seconds between subsequent configuration request transmissions (the interval). You can specify any number greater than or equal to 1. The default value is 30.

Restrictions

Usage is restricted to PPP WAN ports only.

Example

To create a service profile named “pppserv4” with the following attributes:

- Bridging enabled
- IP and IPX enabled

- LCP Echo Requests disabled
- LCP magic numbers disabled
- RED disabled
- A retry interval of 20 seconds

Enter the following command line in Configure mode:

```
xp(config)# ppp define service pppserv4 bridging enable ip enable ipx enable lcp-echo off lcp-magic
      off red off retry-interval 20
```

ppp restart lcp-ncp

Purpose

Restart PPP LCP/NCP negotiation.

Format

ppp restart lcp-ncp ports <*port list*>

Mode

Enable

Description

The **ppp restart lcp-ncp** command allows you to reset and restart the LCP/NCP negotiation process for PPP WAN ports.

Parameters

<*port list*> The ports for which you would like to re-establish LCP/NCP negotiation.

Restrictions

This command line is available only for PPP WAN ports.

Example

To restart LCP/NCP negotiation on serial ports 1 and 2 of slot 4:

```
xp# ppp restart lcp-ncp ports se.4.1,se.4.2
```

ppp set mlp-encaps-format

Purpose

Set MLP encapsulation format.

Format

ppp set mlp-encaps-format ports <*port list*> **[format short-format]**

Mode

Configure

Description

The **ppp set mlp-encaps-format** command allows you to specify the encapsulation format for MLP bundles. If this command is not configured, long format encapsulation is used for MLP bundles.

Parameters

<*port list*>
The MLP port(s) to which you want to apply the encapsulation format

format short-format
Specifies the use of short format for MLP encapsulation.

Restrictions

Usage is restricted to PPP WAN ports only.

Example

To specify short format encapsulation for the MLP bundles “mp.1” and “mp.4-7”:

```
xp(config)# ppp set mlp-encaps-format ports mp.1,mp.4-7 format short-format
```

ppp set mlp-frag-size

Purpose

Set the frame size under which no MLP fragmentation is needed.

Format

ppp set mlp-frag-size ports <*port list* > **[size** <*size*>**]**

Mode

Configure

Description

The **ppp set mlp-frag-size** command allows you to set the frame size under which no fragmentation is needed for transmission on the MLP bundle. The default size is 1500 bytes. Any frames that are less than the value set by the **ppp set mlp-frag-size** command are not fragmented. Any frames that are over the value are fragmented for transmission on the MLP bundle.

Parameters

<*port list*>	The MLP port(s) to which the frame size applies.
<*size*>	The size of the frame, in bytes, that are fragmented by MLP. The value can be between 64 and 1500, inclusive. The default value is 1500.

Restrictions

Usage is restricted to PPP WAN ports only.

Example

To specify that frames of 200 bytes or more are fragmented on the MLP bundles “mp.1” and “mp.4-7”:

```
xp(config)# ppp set mlp-frag-size ports mp.1,mp.4-7 size 200
```

ppp set mlp-fragq-depth

Purpose

Set the depth of the MLP fragment queue.

Format

ppp set mlp-fragq-depth ports <*port list* > **qdepth** <*number-of-packets*>

Mode

Configure

Description

The **ppp set mlp-fragq-depth** command allows you to set the depth of the queue used by MLP to hold packet fragments for reassembly.

Parameters

<*port list*> The MLP port(s) to which the queue depth applies.

<*number-of-packets*>
The depth of the queue, in packets, to hold unassembled packet fragments. The value can be between 100 and 4000, inclusive. The default value is 1000.

Restrictions

Usage is restricted to MLP WAN ports only.

Example

To specify a queue depth of 2500 packets to hold fragments for reassembly on the MLP bundles "mp.1":

```
xp(config)# ppp set mlp-fragq-depth ports mp.1 size 2500
```

ppp set mlp-orderq-depth

Purpose

Set the depth of the MLP packet order queue.

Format

ppp set mlp-orderq-depth ports <*port list* > **qdepth** <*number-of-packets*>

Mode

Configure

Description

The **ppp set mlp-orderq-depth** command allows you to set the depth of the queue used by MLP to hold MLP packets for preserving the packet order.

Parameters

<*port list*> The MLP port(s) to which the queue depth applies.

<*number-of-packets*>
The depth of the queue, in packets, to hold MLP packets. The value can be between 100 and 4000, inclusive. The default value is 1000.

Restrictions

Usage is restricted to MLP WAN ports only.

Example

To specify a queue depth of 2500 packets to hold packets for reordering on the MLP bundles "mp.1":

```
xp(config)# ppp set mlp-orderq-depth ports mp.1 size 2500
```

ppp set payload-compress

Purpose

Enables packet compression for PPP ports.

Format

ppp set payload-compress [max-histories <*number*>**] [type stac] ports** <*port list*>

Mode

Configure

Description

The **ppp set payload-compress** command allows you to enable the Stacker payload compression. You can enable compression on a single port, an entire multilink PPP (MLP) bundle, or on individual ports that are members of a multilink PPP bundle. If this command is not configured, payload compression is not enabled.

Parameters

<*number*>
Specifies the maximum number of compression history buffers to be kept. You can specify either 0 or 1. Specifying 0 disables the keeping of any histories and each packet is individually compressed. Specifying 1 allows a history buffer to be kept, which may result in better compression. The default value is 1.

type stac
Specifies the Stacker (STAC LZS) compression algorithm. This is the default.

<*port list*>
The port(s) on which you want to enable payload compression. You can specify a single port or a comma-separated list of ports.

Restrictions

Usage is restricted to PPP WAN ports only.

Example

To enable LZS Stac payload compression on slot 4, on serial port 2:

```
xp(config)# ppp set payload-compress ports se.4.2
```

ppp set payload-encrypt

Purpose

Enables packet encryption for PPP ports.

Format

ppp set payload-encrypt [type des-bis] ports *<port list>*

Mode

Configure

Description

The **ppp set payload-encrypt** command allows you to enable the encryption of packets using the DES-bis algorithm. You can enable encryption on a single port, an entire multilink PPP (MLP) bundle, or on individual ports that are members of an MLP bundle. If this command is not configured, payload encryption is not enabled.

Parameters

type des-bis
Specifies the DES-bis encryption algorithm. This is the default.

<port list>
The port(s) on which you want to enable payload encryption. You can specify a single port or a comma-separated list of ports.

Restrictions

Usage is restricted to PPP WAN ports only.

Example

To enable DES-bis payload encryption on slot 4, on serial port 2:

```
xp(config)# ppp set payload-encrypt ports se.4.2
```

Note: After the router executes this command, the CLI will prompt the user for transmit-key and receive-key information.

ppp set peer-addr

Purpose

Set the peer address in case that IPCP/IPXCP can't resolve the address.

Format

ppp set peer-addr [ip-address <*IP address*>**] [ipx-address** <*IPX address*>**] [ports** <*port list*>**]**

Mode

Configure

Description

Issuing the **ppp set peer-addr** command allows you to set the peer address if it can't be resolved by IPCP or IPXCP.

Parameters

<*address*>	The IP or IPX address you wish to use.
<*port*>	The port to which you wish to assign the address.

Restrictions

Usage is restricted to PPP port only.

Example

To assign an ip address 10.1.1.1/16 to slot 2, serial port 1:

```
xp(config)# ppp set peer-addr ip-address 10.1.1.1/16 ports se.2.1
```

ppp set ppp-encaps-bgd

Purpose

Force the ingress packets to be encapsulated in bridged format.

Format

ppp set ppp-encaps-bgd ports <*port list*>

Mode

Configure

Description

Issuing the **ppp set ppp-encaps-bgd** command allows you to use bridged format encapsulation on a given ppp port.

Parameters

<*port list*> The port(s) to which you wish to use bridged encapsulation. You can specify a single port or a comma-separated list of ports.

Restrictions

Usage is restricted to ppp port only.

Example

To force the bridged encapsulation to slot 2, serial ports 1 and 2:

```
xp(config)# ppp set ppp-encaps-bgd ports se.2.1,se.2.2
```

ppp show mlp

Purpose

Displays the PPP ports that have been added into an MLP bundle.

Format

ppp show mlp <*mlp list*>|**all-ports**

Mode

Enable

Description

The **ppp show mlp** command allows you to display information about one or more MLP bundles.

Parameters

<*mlp list*>	The name(s) of the MLP bundles on which you want information. You can specify a single bundle or a comma-separated list of MLP bundles.
all-ports	Displays information on all MLP ports.

Restrictions

None.

Example

To display the PPP ports for mp.1:

```
xp# ppp show mlp mp.1
mp.1:
  Slot: 4
  PPP ports: se.4.1,se.4.3
```

ppp show service

Purpose

Displays PPP service profiles.

Format

ppp show service <*service name*>**|all**

Mode

Enable

Description

The **ppp show service** command allows you to display one or all of the available PPP service profiles.

Parameters

<*service name*>	The service profile you wish to display.
all	Displays all of the available PPP service profiles.

Restrictions

None.

Example

To display the available PPP service profiles named profile_4:

```
xp# ppp show service profile_4
```

ppp show stats

Purpose

Displays bridge NCP, IP NCP, and link-status parameters.

Format

ppp show stats ports <*port*> **[bridge-ncp] [ip-ncp] [link-status] [summary]**

Mode

Enable

Description

The **ppp show stats** command allows you to display parameters for bridge NCP, IP NCP, and link-status on PPP WAN ports. You can specify one, two, or three of the available parameter types.

Parameters

<*port*>	The PPP WAN port for which you wish to view bridge NCP, IP NCP, and/or link-status parameters.
bridge-ncp	Specifies that you wish to view bridging NCP parameters for the given port.
ip-ncp	Specifies that you wish to view IP NCP parameters for the given port.
link-status	Specifies that you wish to view link-status parameters for the given port.
summary	Specifies that you wish to view summarized display.

Restrictions

None.

Example

To display the available link-status and IP NCP parameters for the PPP WAN interface located at slot 4, port 1:

```
xp# ppp show stats ports se.4.1 ip-ncp link-status
```

Chapter 50

pvst Commands

The **pvst** commands let you display and change settings for a VLAN spanning tree.

Command Summary

Table 39 lists the **pvst** commands. The sections following the table describe the command syntax.

Table 39. pvst commands

pvst create spanningtree vlan-name <*string*>
pvst enable port <*port-list*> **spanning-tree** <*string*>
pvst set bridging [**forward-delay** <*num*>**] [hello-time** <*num*>**] [max-age** <*num*>**]** **[priority** <*num*>**] spanning-tree** <*string*>
pvst set port <*port-list*> **priority** <*num*> **port-cost** <*num*> **spanning-tree** <*string*> **point-to-point [ForceTrue|ForceFalse|Auto] edge-port [True|False]**
pvst show bridging-info spanning-tree <*string*>
pvst reset-rstp port <*port-list*> **spanning-tree** <*string*>
pvst set protocol-version rstp spanning-tree <*string*>
pvst set no-special-encap

Note: The X-Pedition does *not* support PVST over POS. However, the router *will* support STP over POS.

pvst create spanningtree

Purpose

Create an instance of spanning tree for a particular VLAN.

Format

pvst create spanningtree vlan-name <*string*>

Mode

Configure

Description

The **pvst create spanningtree** command creates a spanning tree instance for a particular VLAN.

Parameters

vlan-name <*string*>
The name of the VLAN for which a new instance of spanning tree is to be created.

Note: The X-Pedition will display VLAN names up to 32 characters in length.

Restrictions

None.

pvst enable port spanning-tree

Purpose

Enable PVST on one or more ports on a particular spanning tree.

Format

pvst enable port <*port-list*> **spanning-tree** <*string*>

Mode

Configure

Description

The **pvst enable port** command enables STP on the specified port for the specified spanning tree.

Parameters

<*port-list*> The ports on which you are enabling STP. You can specify a single port or a comma-separated list of ports. Example: et.1.3,et.(1-3).(4,6-8).

<*string*> The name of the spanning-tree instance. This name is the same as the VLAN name.

Note: For default VLAN, use **stp** commands. The X-Pedition will display VLAN names up to 32 characters in length.

Restrictions

For PVST, the spanning tree instance must have previously been created.

pvst set bridging spanning-tree

Purpose

Set STP bridging parameters for a particular VLAN.

Format

pvst set bridging [**forward-delay** <*num*>**] [hello-time** <*num*>**] [max-age** <*num*>**]**
[priority <*num*>**] spanning-tree** <*string*>

Mode

Configure

Description

The **pvst set bridging spanning-tree** command lets you configure the following STP parameters for a particular VLAN:

- Bridging priority
- Hello time
- Maximum age
- Forward delay

Parameters

forward-delay <*num*>
Sets the STP forward delay for the X-Pedition. The forward delay is measured in seconds. Specify a number from 4– 30. The default is 15.

hello-time <*num*>
Sets the STP hello time for the X-Pedition. The hello time is measured in seconds. Specify a number from 1– 10. The default is 2.

max-age <*num*>
Sets the STP maximum age for the X-Pedition. Specify a number from 6–40. The default is 20.

priority <*num*>
Sets the STP bridging priority for the X-Pedition. Specify a number from 0 – 65535. The default is 32768

spanning-tree <*string*>
The name of the spanning-tree instance. This name is the same as the VLAN name.

Note: For default VLAN, use **stp** commands.The X-Pedition will display VLAN names up to 32 characters in length.

Restrictions

For PVST, the spanning tree instance must have previously been created.

Examples

To set the bridging priority of Spanning Tree for VLAN ip1 to 1:

```
xp(config)# pvst set bridging priority 1 spanning-tree ip1
```

pvst set port

Purpose

Set PVST port priority and port cost for ports for a particular VLAN.

Format

pvst set port <*port-list*> **priority** <*num*> **port-cost** <*num*> **spanning-tree** <*string*> **point-to-point [forcetrue|forcefalse|auto] edge-port [true|false]**

Mode

Configure

Description

The **pvst set port** command sets the STP priority and port cost for individual ports for a particular VLAN.

Parameters

port <*port-list*>
The port(s) for which you are setting STP parameters. You can specify a single port or a comma-separated list of ports. Example: et.1.3,et.(1-3).(4,6-8).

priority <*num*>
The priority you are assigning to the port(s). Specify a number from 0– 16 inclusive. The default is 8.

port-cost <*num*>
The STP cost you are assigning to the port(s). Specify a number from 1– 65535. The default depends on the port speed: 1 for Gigabit (100-Mbps) ports, 10 for 100-Mbps ports, and 100 for 10-Mbps ports.

spanning-tree <*string*>
The name of the spanning-tree instance. This name is the same as the VLAN name.

Note: The X-Pedition will display VLAN names up to 32 characters in length.

point-to-point [ForceTrue|ForceFalse|Auto]
Specify a point-to-point or a non-point-to-point link administratively. The default is ‘Auto.’

edge-port [True|False]
Specify whether the port(s) should be initialized as an edge port or a non-edge port. The default is ‘False.’

Note: For default VLAN, use **stp** commands.

Restrictions

- For PVST, the spanning tree instance must have previously been created.
- With the introduction of the ER16, an X-Pedition router can support up to 480 ports—this exceeds the 256-port limit allowed by the 8-bit port number field specified in the IEEE 802.1D-1998 standard. To accommodate the increase in the number of supported ports, Enterasys extended the port field to a 12-bit value and decreased the port priority field to a 4-bit value. As a result, the X-Pedition allows STP or PVST port configurations with a priority of 0 to 15 only. In spite of these changes, the X-Pedition remains compatible with other switches.

pvst show bridging-info spanning-tree

Purpose

Display STP bridging information for a particular VLAN.

Format

pvst show bridging-info spanning-tree <*string*>

Mode

Enable

Description

The **pvst show bridging-info** command displays STP bridging information for a particular VLAN.

Parameters

spanning-tree <*string*>
The name of the spanning-tree instance. This name is the same as the VLAN name.

Note: For default VLAN, use **stp** commands. The X-Pedition will display VLAN names up to 32 characters in length.

Restrictions

For PVST, the spanning tree instance must have previously been created.

pvst reset-rstp spanning-tree

Purpose

Reset RSTP.

Format

pvst reset-rstp port <*port list*> **spanning-tree** <*string*>

Mode

Enable

Description

The **pvst reset-rstp spanning-tree** command resets the point-to-point and edge port parameters to user-specified values and forces the specified ports to send RSTP BPDUs until a version 0 STP BPDU is received.

Parameters

port <*port-list*>
The port(s) for which you are setting the STP parameters. You can specify a single port or a comma-separated list of ports. For example: et.1.3, et.(1-3).(4,6-8).

spanning-tree <*string*>
The name of the spanning-tree instance. This name is the same as the VLAN name.

Note: As a default, use the STP commands. The X-Pedition will display VLAN names up to 32 characters in length.

Restrictions

For PVST, the spanning tree instance must have previously been created.

pvst set protocol-version rstp spanning-tree

Purpose

Set PVST version to 2 (IEEE 802.1w).

Format

stp set protocol-version rstp spanning-tree <*string*>

Mode

Configure

Description

The **stp set protocol-version** command changes the STP version from *STP compatible* (version 0) to *Rapid Spanning Tree Protocol* (version2).

Parameters

spanning-tree <*string*>
The name of the spanning-tree instance. This name is the same as the VLAN name.

Note: The X-Pedition will display VLAN names up to 32 characters in length.

Restrictions

For PVST, the spanning tree instance must have previously been created.

pvst set no-special-encap

Purpose

To disable PVST encapsulation on all port-based VLANs.

Format

pvst set no-special-encap

Mode

Configure

Description

The stp **set no-special-encap** command forces the X-Pedition's port-based VLAN to send and receive IEEE standard BPDU's. Issue this command if PVST needs to be compatible with STP.

Parameters

None

Restrictions

None.

Chapter 51

qos Commands

The **qos** commands define and display Quality of Service (QoS) parameters. Use the command to classify Layer 2, Layer 3, and Layer 4 traffic into the following priorities:

- control
- high
- medium
- low

By assigning priorities to network traffic, you can ensure that critical traffic will reach its destination even if the exit ports for the traffic are experiencing greater than maximum utilization. Use the **qos set l2**, **qos set ip**, and **qos set ipx** commands to assign priorities for Layer-2, IP, and IPX traffic respectively.

Flows

For Layer 3 (IP and IPX) traffic, you can define “flows”, blueprints or templates of IP and IPX packet headers.

- The IP fields are source IP address, destination IP address, UDP/TCP source port, UDP/TCP destination port, TOS (Type of Service), transport protocol (TCP or UDP) and a list of incoming interfaces.
- The IPX fields are source network, source node, destination network, destination node, source port, destination port, and a list of incoming interfaces.

The flows specify the contents of these fields. If you do not enter a value for a field, a wildcard value (all values acceptable) is assumed for the field.

Precedence

A precedence from 1 – 7 is associated with each field in a flow. The X-Pedition uses the precedence value associated with the fields to break ties if packets match more than one flow. The highest precedence is 1 and the lowest is 7. Here are the default precedences of the fields:

- **IP** – destination port (1), destination address (2), source port (3), source IP address (4), TOS (5), interface (6), protocol (7).
- **IPX** – destination network (1), source network (2), destination node (3), source node (4), destination port (5), source port (6), interface (7).

Use the **qos precedence ip** and **qos precedence ipx** commands to change the default precedences.

Queuing Policies

You can use one of two queuing policies on the X-Pedition:

- **strict priority** – assures the higher priorities of throughput but at the expense of lower priorities. For example, during heavy loads, low-priority traffic can be dropped to preserve throughout of control-priority traffic, and so on.
- **weighted fair queuing** – distributes priority throughput among the four priorities (control, high, medium, and low) based on percentages.

The X-Pedition can use only one queuing policy at a time. The policy is used on the entire X-Pedition. The default queuing policy is strict priority.

Command Summary

Table 40 lists the **qos** commands. The sections following the table describe the command syntax.

Table 40. qos commands

qos apply priority-map <*string*> **ports** <*port list*>
qos create priority-map <*string*> <*num*> **control| low| med| high**
qos precedence ip [sip <*num*>**] [dip** <*num*>**] [srcport** <*num*>**] [destport** <*num*>**] [tos** <*num*>**] [protocol** <*num*>**] [intf** <*num*>**]**
qos precedence ipx [srcnet <*num*>**] [srcnode** <*num*>**] [srcport** <*num*>**] [dstnet** <*num*>**] [dstnode** <*num*>**] [dstport** <*num*>**] [intf** <*num*>**]**
qos priority-map off
qos wred input [port <*port list*>**|all-ports] [queue control|high|medium|low] [exponential-weighting-constant** <*num*>**] [min-queue-threshold** <*num*>**] [max-queue-threshold** <*num*>**] [mark-prob-denominator** <*num*>**]**

Table 40. qos commands (Continued)

qos set ip <*name*> <*priority*>**|low |medium |high |control |num** <*srcaddr/mask*>**|any** <*dstaddr/mask*>**|any** <*srcport*>**|any** <*dstport*>**|any** <*tos*>**|**<*interface-list*>**|any** <*protocol*>
qos set ipx <*name*> <*priority*>**|low |medium |high |control |num** <*srcnet*>**|any** <*srcmask*>**|any** <*srcport*>**|any** <*dstnet*>**|any** <*dstmask*>**|any** <*dstport*>**|**<*interface-list*>**|any**
qos set l2 name <*name*> **source-mac** <*MACaddr*> **source-mac-mask dest-mac** <*MACaddr*> **dest-mac-mask [vlan** <*vlanID*>**|any] in-port-list** <*port-list*> **priority control|high|medium|low|**<*num*>
qos set queuing-policy weighted-fair port <*port list*>**|all-ports**
qos set weighted-fair control <*percentage*> **high** <*percentage*> **medium** <*percentage*> **low** <*percentage*> **port** <*port list*>**|all-ports**
qos show ip
qos show ipx
qos show l2 all-destination all-flow ports <*port-list*> **vlan** <*vlanID*> **source-mac** <*MACaddr*> **dest-mac** <*MACaddr*>
qos show precedence ip| ipx
qos show priority-map <*string*>**| all**
qos show wred [input port <*port list*> **|all-ports] [port** <*port list*> **|all-ports]**
qos show wfq port <*port list*> **|all-ports**

qos apply priority-map

Purpose

Applies a pre-defined priority map to a port(s).

Format

qos apply priority-map <*string*> **ports** <*port list*>

Mode

Configure

Description

The **qos apply priority-map** command allows you apply a previously defined priority map to a port or multiple ports. A priority map associates certain 802.1p tag values inside the frame to a certain internal priority queue. Use the **qos create priority-map** command to first create a priority map.

By default, the X-Pedition maps the number to the four internal priorities as follows: 0 or 1 = low; 2 or 3 = medium; 4 or 5 = high; 6 or 7 = control.

Parameters

priority-map <*string*>
Specifies the name of the map. Specify a string less than 25 characters.

port <*port list*>
Specifies the port(s) on which you want to apply the priority map.

Restrictions

None.

Example

The following command applies the priority map ‘map1’ to port so.2.1:

```
xp(config)# qos apply priority-map ‘map1’ port so.2.1
```

qos create priority-map

Purpose

Creates a priority map to an 802.1p tag.

Format

qos create priority-map <*string*> <*num*> **control| low| med| high**

Mode

Configure

Description

The **qos create priority-map** command lets you map 802.1p tags from a frame to one of the four internal priority queue classes: **control**, **low**, **medium**, and **high**. Internal priority queue classes are used in to prioritize flows during traffic congestion situations. The flows with the higher priority is given precedence over lower priorities. The internal priority class **control** receives the highest precedence, while **low** receives the lowest precedence.

The 802.1p standard provides a way of tagging frames to a certain internal priority. With this command, you can set a particular 802.1p priority tag to map to a specific internal priority queue.

By default, the X-Pedition maps the number to the four internal priorities as follows: 0 or 1 = low; 2 or 3 = medium; 4 or 5 = high; 6 or 7 = control.

Parameters

priority-map <*string*>
Specifies the name of the map. Specify a string less than 25 characters in length.

<*num*>
Specifies the 802.1p priority tag that you want to map. Specify a number between 0 and 7.

queue control|high|medium|low
Specifies the internal priority queue. Specify either the **control, high, medium, or low** queue.

Restrictions

None.

Example

The following command creates a priority map ‘map1’ that maps the 802.1p tags 0 and 1 to low, 2 and 3 to medium, 4 and 5 to high, and 6 and 7 to control queue:

```
xp(config)# qos create priority-map ‘map1’ 0 low 1 low 2 medium 3 medium 4 high 5 high 6 control 7
      control
```

qos precedence ip

Purpose

Set the precedence of the IP flow fields.

Format

qos precedence ip [sip <*num*>**] [dip** <*num*>**] [srcport** <*num*>**] [destport** <*num*>**]**
[tos <*num*>**] [protocol** <*num*>**] [intf** <*num*>**]**

Mode

Configure

Description

The **qos precedence ip** command lets you set the QoS precedence for various flow fields in IP traffic. You can set a precedence from 1 – 7 for the following IP fields:

- IP source address
- IP destination address
- Source TCP or UDP port
- Destination TCP or UDP port
- Type of Service (TOS) for the packet
- Protocol (TCP or UDP)
- Incoming interface

The precedence 1 is the highest priority. IP interfaces or flow fields within IP packets that have a precedence of 1 are given first priority. The default priorities are as follows:

- destination port (1)
- destination address (2)
- source port (3)
- source IP address (4)
- TOS (5)
- interface (6)
- protocol (7).

Parameters

sip <*num*>
Specifies the precedence of the source address field in IP flows. Specify a precedence from 1 – 7.

dip <*num*>
Specifies the precedence of the destination address field in IP flows. Specify a precedence from 1 – 7.

srcport <*num*>
Specifies the precedence of the source port field in IP flows. Specify a precedence from 1 – 7.

dstport <*num*>
Specifies the precedence of the destination port field in IP flows. Specify a precedence from 1 – 7.

tos <*num*>
Specifies the precedence of the TOS field in IP flows. Specify a precedence from 1 – 7.

protocol <*num*>
Specifies the precedence of the transport layer protocol name field in IP flows. Specify a precedence from 1 – 7.

intf <*num*>
Specifies the precedence of the IP interface based on the interface's name. Specify a precedence from 1 – 7.

Restrictions

None.

Examples

To change the precedence for fields within IP flows from the default precedences listed above:

```
xp(config)# qos precedence ip sip 3 dip 1 srcport 2 destport 4 tos 5
      protocol 6 intf 7
```

qos precedence ipx

Purpose

Set the precedence of the IPX flow fields.

Format

qos precedence ipx [srcnet <*num*>**] [srcnode** <*num*>**] [srcport** <*num*>**] [dstnet** <*num*>**] [dstnode** <*num*>**] [dstport** <*num*>**] [intf** <*num*>**]**

Mode

Configure

Description

The **qos precedence ipx** command lets you set the precedence of the following fields in IPX flows.

- Source network
- Source port
- Source node
- Destination network
- Destination node
- Destination port
- Incoming interface

You can set the precedence of the following fields from 1 – 7. The precedence 1 has the highest priority and 7 has the lowest. The default priorities are as follows:

- destination network (1)
- source network (2)
- destination node (3)
- source node (4)
- destination port (5)
- source port (6)
- interface (7).

Parameters

srcnet <*num*>
Specifies the precedence of the source network field in IPX flows. Specify a precedence from 1 – 7.

srcport <*num*>
Specifies the precedence of the source port field in IPX flows. Specify a precedence from 1 – 7.

srcnode <*num*>
Specifies the precedence of the source node field in IPX flows. Specify a precedence from 1 – 7.

dstnet <*num*>
Specifies the precedence of the destination network field in IPX flows. Specify a precedence from 1 – 7.

dstnode <*num*>
Specifies the precedence of the destination node field in IPX flows. Specify a precedence from 1 – 7.

dstport <*num*>
Specifies the precedence of the destination port field in IPX flows. Specify a precedence from 1 – 7.

intf <*num*>
Specifies the precedence of the IPX interface based on the interface's name. Specify a precedence from 1 – 7.

Restrictions

None.

Examples

To change the precedence for fields within IPX flows from the default precedences listed above:

```
xp(config)# qos precedence ipx srcnet 1 srcnode 2 srcport
      dstnet 3 srcport 4 dstnode 5 dstport 6 intf 7
```

qos priority-map off

Purpose

Turns off priority mapping on a port(s).

Format

qos priority-map off

Mode

Configure

Description

The **qos priority-map off** command allows you disable any priority maps applied on a port using the **qos apply priority-map** command and reverts back to the default priority mapping.

By default, the X-Pedition maps the number to the four internal priorities as follows: 0 or 1 = low; 2 or 3 = medium; 4 or 5 = high; 6 or 7 = control.

Parameters

None**.**

Restrictions

None.

Example

The following command disables all priority mapping:

```
xp(config)# qos priority-map off
```

qos wred input

Purpose

Enable WRED on input queues of specific ports.

Format

qos wred input [port <*port list*>**|all-ports] [queue control|high|medium|low] [exponential-weighting-constant** <*num*>**] [min-queue-threshold** <*num*>**] [max-queue-threshold** <*num*>**] [mark-prob-denominator** <*num*>**]**

Mode

Configure

Description

The **qos** w**red input** command lets you set the parameters for Weighted Random Early Detection algorithm and allow you to apply them to input queues of specific ports.

Weighted Random Early Detection alleviates traffic congestion issues by selectively dropping packets before the queue becomes completely flooded. WRED parameters allow you to set conditions and limits for dropping packets in the queue.

Parameters

port <*port list*>|**all-ports**
Specifies the port on which the WRED algorithm will be applied. Specify **all-ports** to apply WRED algorithm to all ports.

queue control|high|medium|low
Allows you to specify which queue to apply the WRED algorithm. Specify either the **control, high, medium, or low** queue.

exponential-weighting-constant <*num*>
Sets the queue weight. Specify a number from 7-10. The default is 8.

min-queue-threshold <*num*>
Sets the minimum queue length. When the queue length rises above this threshold, packets begin to drop. Specify any number between 5 and 652.

max-queue-threshold <*num*>
Sets the maximum queue length. When the queue length reaches this threshold, packets are dropped according to the mark probability denominator. Specify any number between 5 and 652.

mark-prob-denominator <*num*>
Specifies the fraction of the packets to be dropped when the queue length reaches the maximum threshold. Specify a number from 10-100. The default is 50.

Restrictions

WRED should only be applied for TCP/IP traffic.

Examples

The following command sets WRED on port et.2.1 for the input high queue, sets the queue weight at 8, minimum queue length at 10, maximum queue length at 100, and the fraction of packets dropped to be 50:

```
xp(config)# qos wred input port et.2.1 queue high exponential-weighting-constant 8 min-queue-
      threshold 10 max-queue-threshold 100 mark-prob-denominator 50
```

qos set ip

Purpose

Set a priority for an IP flow.

Format

qos set ip <*name*> <*priority*> **[**<*srcaddr/mask*>**|any] [**<*dstaddr/mask*>**|any] [**<*srcport*>**|any] [**<*dstport*>**|any] [**<*tos*>**|any] [**<*port list*>|<*interface-list*>**|any] [**<*protocol*>**|any] [**<*tos-mask*>**|any] [**<*tos-precedence-rewrite*>**|any] [**<*tos-rewrite*>**|any]**

Mode

Configure

Description

The **qos set ip** command sets the priority for an IP flow based on the following fields in the flow:

- Flow name
- Source IP address and network mask
- Destination IP address and network mask
- Source port
- Destination port
- TOS
- Layer 4 bridging port list or interface list
- Transport layer protocol (TCP or UDP)

You can set the priority of each field to control, low, medium, or high. The default is low.

Parameters

<*name*>
Specifies the IP flow name.

<*priority*>
Specifies the priority you are assigning to the flow parameters you specified from the list above. You can specify one of the following priorities:

control Assigns control priority to the IP flow parameters you have specified. This is the highest priority.

high Assigns high priority to the IP flow parameters you have specified.

medium Assigns medium priority to the IP flow parameters you have specified.

low Assigns low priority to the IP flow parameters you have specified. This is the default.

<*srcaddr/mask*>|**any**

Specifies the source IP address and network mask for which you are assigning a priority. You can specify the mask using the traditional IP address format ("255.255.0.0") or the CIDR format ("/16").

If you specify **any** instead of a network mask, the X-Pedition assumes a wildcard "don't care" condition. If you do not specify a mask, then the X-Pedition assumes a mask of 255.255.255.255. You cannot substitute the mask with the **any** keyword. The keyword **any** is for the entire <*srcaddr/mask*> pair.

<*dstaddr/mask*>|**any**

Specifies the destination IP address and network mask for which you are assigning a priority. The same requirements and restrictions for <*srcaddr/mask*> **apply to** <*dstaddr/mask*>.

If you specify **any** instead of a network mask, the X-Pedition assumes a wildcard "don't care" condition. If you do not specify a mask, then the X-Pedition assumes a mask of 255.255.255.255. You cannot substitute the mask with the **any** keyword. The keyword **any** is for the entire <*dstaddr/mask*> pair.

<*srcport*>|**any**

Specifies the source TCP or UDP port for which you are assigning a priority. Specify a port number from 1 – 65535 or **any** to allow any value. You may also specify a range of port numbers using operator symbols; for example, 10-20 (between 10 and 20 inclusive), >1024 (greater than 1024), <1024 (less than 1024), !=1024 (not equal to 1024).

<*dstport*>|**any**

Specifies the destination TCP or UDP port for which you are assigning a priority. Specify a port number from 1 – 65535 or **any** to allow any value. You may also specify a range of port numbers using operator symbols; for example, 10-20 (between 10 and 20 inclusive), >1024 (greater than 1024), <1024 (less than 1024), !=1024 (not equal to 1024).

<*tos*>|**any**

Specifies the TOS for which you are assigning a priority. Specify a number from 0– 255 or **any** to allow any value.

<*port list*>|<*interface-list*>|**any**

Specifies one or more Layer 4 bridging ports or one or more IP interface names for which you are assigning priority. If you specify a list, delimit the interface names with commas. Specify **any** to allow any IP interface name.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

<*protocol*>|**any**

Specifies the transport layer protocol for which you are assigning priority. You can specify one of the following values:

tcp Assigns the priority parameters to the TCP protocol.

udp Assigns the priority parameters to the UDP protocol.

any Assigns the priority parameters to both the TCP and UDP protocols.

<*tos-mask*>
Specifies the mask that is used for the TOS byte. Specify a number from 1-255 or **any** to specify any TOS value. The default is 30.

<*tos-precedence-rewrite*>
Rewrites the precedence portion of the TOS field with a new value. Specify a number from 0-7 or **any** to specify any TOS value.

<*tos-rewrite*>
Rewrites the entire TOS field with a new value. Specify a number from 0-31 or **any** to specify any TOS value.

Note: If you set **any** for the TOS precedence rewrite and specify a value for <*tos-rewrite*>, then the precedence portion of the TOS field remains the same as in the packet, but the rest of the TOS field is rewritten. If you specify values for both <*tos-precedence-rewrite*> and <*tos-rewrite*>, then the precedence portion of the TOS field is rewritten to the new <*tos-precedence-rewrite*> number and the rest of the TOS field is rewritten to the new <*tos-rewrite*> number.

Restrictions

None.

Examples

The following command creates a flow called “flow1”. This flow provides a template for an IP packet with the IP address 1.1.1.1, network mask 255.255.0.0, destination address 2.2.2.2 (and implied destination mask 255.255.255.255). The flow includes source TCP/UDP port 3010, destination port 3000, a TOS of 15, the interfaces mls1 and mls2, and the TCP protocol as transport layer. This very explicit flow has the highest priority—control.

```
xp(config)# qos set ip flow1 control 1.1.1.1/255.255.0.0 2.2.2.2 3010 3000 15 mls1,mls2 tcp
```

qos set ipx

Purpose

Set a priority for an IPX flow.

Format

qos set ipx <*name*> <*priority*> [<*srcnet*>|**any**] [<*srcmask*>|**any**] [<*srcport*>|**any**] [<*dstnet*>|**any**] [<*dstmask*>|**any**] [<*dstport*>|**any**] [<*port list*>|<*interface-list*>|**any**]

Mode

Configure

Description

The **qos set ipx** command lets you set the priority for an IPX flow based on the following fields in the flow:

- Flow name
- Source network
- Source network mask
- Source port
- Destination network
- Destination network mask
- Destination port
- Layer 4 bridging port list or interface list

You can set the priority of each field to control, low, medium, or high. The default is low.

Parameters

<*name*>
Specifies the IPX flow name.

<*priority*>
Specifies the priority you are assigning to the flow parameters you specified from the list above. You can specify one of the following priorities:

control Assigns control priority to the IPX flow parameters you have specified. This is the highest priority.

high Assigns high priority to the IPX flow parameters you have specified.

medium Assigns medium priority to the IPX flow parameters you have specified.

low Assigns low priority to the IPX flow parameters you have specified. This is the default.

<*srcnet*>|**any**
Specifies the IPX source network and node address. Specify them in the following format: <*netaddr*>**.**<*macaddr*>; for example: a1b2c3d4.aa:bb:cc:dd:ee:ff.

If you specify **any** instead of a **.**<*macaddr*>, the X-Pedition assumes a wildcard value. All MAC addresses are then valid.

<*srcmask*>|**any**
Specifies the IPX source network mask. Specify the mask in hexadecimal digits. If you do not specify a mask value and instead use the value **any**, the X-Pedition internally sets the mask to FFFFFFFF.

<*srcport*>|**any**
Specifies a port number from 1 – 65535 or any to allow any value. You may also specify a range of port numbers using operator symbols; for example, 10-20 (between 10 and 20 inclusive), >1024 (greater than 1024), <1024 (less than 1024), !=1024 (not equal to 1024).

<*dstnet*>|**any**
Specifies the IPX destination network and node address. The same requirements and restrictions for <*dstaddr*> *apply to* <*srcaddr*>.

<*dstmask*>|**any**
Specifies the IPX destination network mask. Specify the mask in hexadecimal digits or **any** to allow any value.

<*dstport*>|**any**
Specifies a port number from 1 – 65535 or **any** to allow any value. You may also specify a range of port numbers using operator symbols; for example, 10-20 (between 10 and 20 inclusive), >1024 (greater than 1024), <1024 (less than 1024), !=1024 (not equal to 1024).

<*port list*>|<*interface-list*>|**any**
Specifies one or more Layer 4 bridging ports or one or more IPX interface names for which you are assigning priority. If you specify a list, delimit the interface names with commas. Specify **any** to allow any IPX interface name.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

None.

Examples

The following command creates an IPX flow called “abc”. This flow gives a high priority to IPX traffic on interface mls1 from network 12345678.00:01:00:00:00:00, mask 0000ff00, port 55 to network 22222222.02:00:00:00:00:00, mask 0000ff00, port 65.

```
xp(config)# qos set ipx abc high 12345678.00:01:00:00:00:00 0000ff00 55 22222222.02:00:00:00:00:00
      0000ff00 65 mls1
```

qos set l2

Purpose

Configure priority for a Layer 2 flow.

Format

qos set l2 name <*name*> **source-mac** <*MACaddr*> **dest-mac** <*MACaddr*> **[vlan** <*vlanID*> **any] in-port-list** <*port-list*> **priority control | high | medium | low |** <*trunk-priority*>

Mode

Configure

Description

The **qos set l2** command lets you set a QoS priority for a Layer 2 flow. You can set a priority for a flow based on the following fields in the flow:

- L2 flow name
- Source MAC address
- Destination MAC address
- VLAN ID
- Incoming port(s)

You can set the priority in one of the following ways:

- The flow is assigned a priority within the X-Pedition. In this case you specify a priority of control, low, medium, or high. The default is low.
- The flow is assigned a priority within the X-Pedition, but in addition, if the exit ports are VLAN trunk ports, the flow is assigned an 802.1Q priority. In this case you specify a number from 0 – 7. The X-Pedition maps the number to the four internal priorities as follows: 0 or 1 = low; 2 or 3 = medium; 4 or 5 = high; 6 or 7 = control.

Note: A packet entering a Q-trunk has an 802.1Q header containing a priority field. Typically, users can change the 802.1Q priority using the **qos set l2** commands. However, current hardware restrictions ignore any request to overwrite the packet's priority—the packet simply replicates at the exit port and continues with its original priority.

Parameters

name <*name*>
Specifies the L2 flow name.

source-mac <*MACaddr*>
Specifies the L2 source MAC address. *Specify the MAC address in either of the following formats:*

xx:xx:xx:xx:xx:xx
xxxxxx:xxxxxx

dest-mac <*MACaddr*>
Specifies the L2 destination MAC address.

vlan <*vlanID*>
Specifies the name of a VLAN.

Note: The X-Pedition will display VLAN names up to 32 characters in length.

in-port-list <*port-list*>
Specifies the X-Pedition ports for which you are setting priority for this flow. The priority applies when the L2 packet enters the X-Pedition on one of the specified ports. The priority does not apply to exit ports.

priority control| high| medium| low| <*trunk-priority*>
Specifies the priority you are assigning to the flow parameters you specified from the list above. You can specify one of the following priorities:

control	Assigns control priority to the IP flow parameters you have specified. This is the highest priority.
high	Assigns high priority to the IP flow parameters you have specified.
medium	Assigns medium priority to the IP flow parameters you have specified.
low	Assigns low priority to the IP flow parameters you have specified. This is the default.
<*trunk-priority*>	Assigns n 802.1Q VLAN trunk priority when the exit port is a VLAN trunk port. The X-Pedition maps the number to the four internal priorities as follows: 0 = low; 1, 2, or 3 = medium; 4, 5,or 6 = high; 7 = control.

Restrictions

None.

qos set queuing-policy

Purpose

Change the queueing policy from strict priority to weighted fair.

Format

qos set queuing-policy weighted-fair port <*port list*>|**all-ports**

Mode

Configure

Description

The **qos set queuing-policy** command lets you override the default queuing policy (strict priority) in favor of weighted fair queuing on specific ports or on all ports. Only one type of queuing policy can be active at a time.

To set the queuing policy back to strict priority, enter the following command:

```
xp(config)# no qos set queuing-policy weighted-fair port <port list>
```

Parameters

weighted-fair
Sets the queuing policy to weighted fair.

port <*port list*>|**all-ports**
Specifies the Ethernet ports or WAN modules and ports on which weighted fair queuing apply. Specify **all-ports** to apply weighted fair queuing to all ports.

Restrictions

None.

qos set weighted-fair

Purpose

Set percentages for weighted-fair queuing.

Format

qos set weighted-fair control <*percentage*> **high** <*percentage*> **medium** <*percentage*> **low** <*percentage*> **port** <*port list*>|**all-ports**

Mode

Configure

Description

The **qos set weighted-fair** command lets you set the percentage of X-Pedition bandwidth allocated to the control, high, medium, and low priorities. The percentages apply to specific ports or to all ports. Make sure the total percentages for all four priorities equals 100.

Parameters

control <*percentage*>
Specifies the percentage of X-Pedition bandwidth allocated to the control priority. Specify a number from 1 – 100. The default is 25.

high <*percentage*>
Specifies the percentage of X-Pedition bandwidth allocated to the high priority. Specify a number from 1 – 100. The default is 25.

medium <*percentage*>
Specifies the percentage of X-Pedition bandwidth allocated to the medium priority. Specify a number from 1 – 100. The default is 25.

low <*percentage*>
Specifies the percentage of X-Pedition bandwidth allocated to the low priority. Specify a number from 1 – 100. The default is 25.

port <*port list*>|**all-ports**
Specifies the Ethernet ports or WAN modules and ports on which the defined percentages apply. Specify **all-ports** to apply the percentages to all ports.

Restrictions

The total percentages for all four QoS levels must equal 100%.

qos show ip

Purpose

Show QoS information for IP flows.

Format

qos show ip

Mode

Enable

Description

The **qos show ip** command lets you display QoS information for IP flows.

Parameters

None.

Restrictions

None.

qos show ipx

Purpose

Show QoS information for IPX flows.

Format

qos show ipx

Mode

Enable

Description

The **qos show ipx** command lets you display QoS information for IPX flows.

Parameters

None.

Restrictions

None.

qos show l2

Purpose

Show QoS information for L2 flows.

Format

qos show l2 all-destination all-flow ports <*port-list*> **vlan** <*vlanID*> **source-mac** <*MACaddr*> **dest-mac** <*MACaddr*>

Mode

Enable

Description

The **qos show l2** command lets you display QoS information for L2 flows. You can filter the display according to the following:

- Destinations
- Flows
- Ports
- VLANs
- Source MAC addresses
- Destination MAC addresses
- Priority

Parameters

all-destination
Filters the display to show all the L2 destination priorities.

all-flow
Filters the display to show all the L2 flow priorities.

ports <*port-list*>
Filters the display to show L2 priority information for specific ports.

vlan <*vlanID*>
Filters the display to show L2 priority information for specific VLANs.

source-mac <*MACaddr*>
Filters the display to show L2 priority information for specific source MAC addresses.

dest-mac <*MACaddr*>
Filters the display to show L2 priority information for specific destination MAC addresses.

Restrictions

None.

qos show precedence

Purpose

Shows IP or IPX precedence values.

Format

qos show precedence ip| ipx

Mode

Enable

Description

The **qos show precedence** command lets you display the precedence values for all fields in a flow.

IP flows consist of the following fields: destination port, destination address, source port, source IP address, TOS, interface, protocol.

IPX flows consist of the following fields: destination network, source network, destination node, source node, destination port, source port, interface.

Parameters

ip
Displays the precedence values for IP flows.

ipx
Displays the precedence values for IPX flows.

Restrictions

None.

qos show priority-map

Purpose

Shows the priority mapping and the ports that it is applied.

Format

qos show priority-map <*string*>**| all**

Mode

Enable

Description

The **qos show priority-map** command lets you display the priority mapping that is configured on a port. The command shows how each 802.1p tag values are mapped to a specific internal priority queue.

Parameters

<*string*>
Specifies the name of the priority map.

all
Displays all priority maps.

Restrictions

None.

qos show wred

Purpose

Shows WRED parameters for each port.

Format

qos show wred [input port <*port list*> **|all-ports] [port** <*port list*> **|all-ports]**

Mode

Enable

Description

The **qos show wred** command lets you display WRED information for a certain port or all ports. You can display WRED parameter information according to the following:

- Input ports
- All Ports

Parameters

input port <*port list*> **|all-ports**
Displays input port WRED parameters. Specify **all-ports** to display parameters for all ports.

port <*port list*> **|all-ports**
Displays WRED parameters for each port. Specify **all-ports** to display parameters for all ports.

Restrictions

None.

qos show wfq

Purpose

Shows bandwidth allocated for each port.

Format

qos show wfq [port <*port list*> **|all-ports] [input** <*slot num*> **|all-modules]**

Mode

Enable

Description

The **qos show wfq** command lets you display the bandwidth for each port allocated with weighted-fair queuing.

Parameters

port <*port list*> **|all-ports**
Displays bandwidth allocated for each port. Specify a list of ethernet or wan ports. Specify **all-ports** to display bandwidth for all ports.

input <*slot num*> **|all-modules**
Displays bandwidth allocated for each slot. Specify a list of occupied slots. Specify **all-modules** to display bandwidth for all modules.

Restrictions

None.

Chapter 52

radius Commands

The **radius** commands let you secure access to the X-Pedition using the Remote Authentication Dial-In User Service (RADIUS) protocol. When users log in to the X-Pedition or try to access Enable mode, they are prompted for a password. If RADIUS authentication is enabled on the X-Pedition, it will contact a RADIUS server to verify the user. If the user is verified, he or she is granted access to the X-Pedition.

Command Summary

Table 41 lists the **radius** commands. The sections following the table describe the command syntax.

Table 41. radius commands

radius accounting command level <*level*>
radius accounting shell start |stop |all
radius accounting snmp active |startup
radius accounting system fatal |error |warning |info
radius authentication login |enable
radius enable
radius set server <*IPaddr*> **[acct-port** <*number*>**] [auth-port** <*number*>**] [timeout** <*number*>**] [retries** <*number*>**] [deadtime** <*number*>**] [key** <*string*>**] [source** <*IFname_IPaddr*>**]** **radius set [timeout** <*number*>**] [retries** <*number*>**] [deadtime** <*number*>**] [key** <*string*>**] [source** <*IFname_IPaddr*>**] last-resort password |succeed |deny**
radius show stats|all

radius accounting command level

Purpose

Causes the specified types of commands to be logged to the RADIUS server.

Format

radius accounting command level <*level*>

Mode

Configure

Description

The **radius accounting command level** command allows you specify the types of commands that are logged to the RADIUS server. The user ID and timestamp are also logged.

Parameters

<level> Specifies the type(s) of commands that are logged to the RADIUS server. Enter one of the following values:

- 5 Log Configure commands.
- 10 Log all Configure and Enable commands.
- 15 Log all Configure, Enable, and User commands.

Restrictions

None.

Example

To cause Configure, Enable, and User mode commands to be logged on the RADIUS server:

```
xp(config)# radius accounting command level 15
```

radius accounting shell

Purpose

Causes an entry to be logged on the RADIUS server when a shell is stopped or started on the X-Pedition.

Format

radius accounting shell start|stop|all

Mode

Configure

Description

The **radius accounting shell** command allows you to track shell usage on the X-Pedition. It causes an entry to be logged on the RADIUS server when a shell is started or stopped. You can specify that an entry be logged when a shell is started, when a shell is stopped, or when a shell is either started or stopped.

Parameters

start Logs an entry when a shell is started.

stop Logs an entry when a shell is stopped

all Logs an entry when a shell is either started or stopped

Restrictions

None.

Example

To cause an entry to be logged on the RADIUS server when a shell is either started or stopped on the X-Pedition:

```
radius accounting shell all
```

radius accounting snmp

Purpose

Logs to the RADIUS server any changes made to the startup or active configuration via SNMP.

Format

radius accounting snmp active|startup

Mode

Configure

Description

The **radius accounting snmp** command allows you to track changes made to the active or startup configuration through SNMP. It causes an entry to be logged on the RADIUS server whenever a change is made to the ACL configuration. You can specify that an entry be logged to the active or startup configuration.

Parameters

active Logs an entry when a change is made to the active configuration.

startup Logs an entry when a change is made to the startup configuration.

Restrictions

None.

Example

To cause an entry to be logged on the RADIUS server whenever an ACL configuration change is made via SNMP to the active configuration:

```
xp(config)# radius accounting snmp active
```

radius accounting system

Purpose

Specifies the type(s) of messages to be logged on the RADIUS server.

Format

radius accounting system fatal|error|warning|info

Mode

Configure

Description

The **radius accounting system** command allows you to specify the types of messages that are logged on the RADIUS server.

Parameters

fatal
Logs only fatal messages.

error
Logs fatal messages and error messages.

warning
Logs fatal messages, error messages, and warning messages.

info
Logs all messages, including informational messages.

Restrictions

None.

Example

To log only fatal and error messages on the RADIUS server:

```
xp(config)# radius accounting system error
```

radius authentication

Purpose

Causes RADIUS authentication to be performed at either the X-Pedition login prompt or when the user tries to access Enable mode.

Format

radius authentication login|enable

Mode

Configure

Description

The **radius authentication** command allows you to specify when RADIUS authentication is performed: either when a user logs in to the X-Pedition, or tries to access Enable mode.

Parameters

login Authenticates users at the X-Pedition login prompt.

enable Authenticates users when they try to access Enable mode.

Restrictions

None.

Example

To perform RADIUS authentication at the X-Pedition login prompt:

```
radius authentication login
```

radius enable

Purpose

Enables RADIUS authentication on the X-Pedition. RADIUS authentication is disabled by default on the X-Pedition.

Format

radius enable

Mode

Configure

Description

The **radius enable** command causes RADIUS authentication to be activated on the X-Pedition. You set RADIUS-related parameters with the **radius set**, **radius accounting shell**, and **radius authorization** commands, then use the **radius enable** command to activate RADIUS authentication.

Parameters

None.

Restrictions

None.

Example

The following commands set RADIUS-related parameters on the X-Pedition. The commands are then activated with the **radius enable** command:

```
radius set server 207.135.89.15
radius set timeout 30
radius authentication login
radius accounting shell all
radius enable
```

radius set

Purpose

Sets parameters for authenticating the X-Pedition through a RADIUS server.

Format

radius set [timeout <*number*>**] [retries** <*number*>**] [deadtime** <*number*>**] [key** <*string*>**] [source** <*IFname_IPaddr*>**] last-resort password |succeed |deny**

Mode

Configure

Description

The **radius set** command allows you to set default RADIUS-related parameters on the X-Pedition, how long to wait for the RADIUS server to authenticate the user, an encryption key, and what to do if the RADIUS server does not reply by a given time.

Parameters

timeout <*number*> Is the maximum time (in seconds) to wait for a RADIUS server to reply. The default is 3 seconds.

retries <*number*> The number of times (1-10) to try contacting this RADIUS server.

deadtime <*number*> The length of time for transaction requests to skip over a RADIUS server—up to a maximum of 1440 minutes (24 hours). This command causes the X-Pedition to mark as “dead” any RADIUS server that fails to respond to authentication requests, thus avoiding the wait for the request to timeout before trying the next configured server. Additional requests for a RADIUS server marked as “dead” will skip the server for the duration of minutes specified (unless all servers are marked “dead”).

key <*string*> Is an encryption key to be shared with the RADIUS server.

source <*IFname_IPaddr*> Sets the source interface name or IP address for RADIUS messages.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

last-resort Is the action to take if a RADIUS server does not reply within the time specified by the **timeout** parameter. If this parameter is *not* specified, user authentication will always fail if the RADIUS server does not reply within the specified timeout period.

Specify one of the following keywords:

password The user is prompted for the password set with **system set password** command. This keyword is *recommended* for optimal security, however, note that you must set a password with the **system set password** command.

succeed Access to the X-Pedition is granted.

deny Unable to connect to RADIUS server, access to the X-Pedition is denied.

Restrictions

None.

Example

The following commands specify that hosts 137.72.5.9 and 137.72.5.41 are RADIUS servers, and the X-Pedition should wait no more than 30 seconds for a response from one of these servers. If a response from a RADIUS server doesn't arrive in 30 seconds, the user is prompted for the password that was set with the X-Pedition **system set password** command.

```
radius set server 137.72.5.9
radius set server 137.72.5.41
radius set timeout 30
radius set last-resort password
```

radius set server

Purpose

Sets parameters for authenticating the X-Pedition through a specific RADIUS server.

Format

radius set server <*IPaddr*> **[acct-port** <*number*>**] [auth-port** <*number*>**] [timeout** <*number*>**] [retries** <*number*>**] [deadtime** <*number*>**] [key** <*string*>**] [source** <*IFname_IPaddr*>**]**

Mode

Configure

Description

The **radius set server** command allows you to set RADIUS-related parameters on the X-Pedition, including the IP address of a specific RADIUS server, how long to wait for the RADIUS server to authenticate the user, an encryption key, and what to do if the RADIUS server does not reply by a given time.

Parameters

server <*IPaddr*>	Is the IP address of a specific RADIUS server. You can enter up to five RADIUS servers. Enter one server per **radius set server** command.
acct-port <*number*>	Enter the accounting port number. The default **Acct-port** number is 1813.
auth-port <*number*>	Enter the authentication port number. The default **Auth-port** number is 1812.
timeout <*number*>	Is the maximum time (in seconds) to wait for a RADIUS server to reply. The default is 3 seconds.
retries <*number*>	The number of times (1-10) to try contacting this RADIUS server.
deadtime <*number*>	The length of time for transaction requests to skip over a RADIUS server—up to a maximum of 1440 minutes (24 hours). This command causes the X-Pedition to mark as “dead” any RADIUS server that fails to respond to authentication requests, thus avoiding the wait for the request to timeout before trying the next configured server. Additional requests for a RADIUS server marked as “dead” will skip the server for the duration of minutes specified (unless all servers are marked “dead”).
key <*string*>	Is an encryption key to be shared with the RADIUS server.

source <*IFname_IPaddr*> Sets the source interface name or IP address for RADIUS messages.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

None.

Example

The following commands specify that hosts 137.72.5.9 and 137.72.5.41 are RADIUS servers, and the X-Pedition should wait no more than 30 seconds for a response from one of these servers. If a response from a RADIUS server doesn't arrive in 30 seconds, the user is prompted for the password that was set with the X-Pedition **system set password** command.

```
radius set server 137.72.5.9
radius set server 137.72.5.41
radius set timeout 30
radius set last-resort password
```

radius show

Purpose

Displays information about RADIUS configuration on the X-Pedition.

Format

radius show stats|all

Mode

Enable

Description

The **radius show** command displays statistics and configuration parameters related to RADIUS configuration on the X-Pedition. The statistics displayed include:

accepts	Number of times each server responded and validated the user successfully.
rejects	Number of times each server responded and denied the user access, either because the user wasn't known, or the wrong password was supplied.
timeouts	Number of times each server did not respond.

Parameters

stats	Displays the accepts, rejects, and timeouts for each RADIUS server.
all	Displays the configuration parameters set with the **radius set** command, in addition to the accepts, rejects, and timeouts for each RADIUS server.

Restrictions

None.

Example

To display configuration parameters and RADIUS server statistics:

```
radius show all
RADIUS status:                      ACTIVE
RADIUS last resort:                 Succeed when server fails
Command Level Logging:               15 - Log Configure, Enable and User Commands
Default RADIUS timeout (seconds):   3
Default RADIUS retries:             3
Default RADIUS deadtime (minutes):  0
Default RADIUS key:                 net
Default RADIUS source IP address:   Let system decide

RADIUS servers listed in order of priority:

Server:                  10.136.16.102
Port:                    49
Timeout (seconds):       <Default>
Retries:                 <Default>
Deadtime (minutes):      3
Key:                     net
Source IP:               <Default>
Server is dead. Will be made tested again in 2 minutes

Server:                  10.136.15.100
Port:                    49
Timeout (seconds):       <Default>
Retries:                 <Default>
Deadtime (minutes):      <Default>
Key:
Source IP:               <Default>

Server:                  10.136.15.101
Port:                    49
Timeout (seconds):       <Default>
Retries:                 <Default>
Deadtime (minutes):      <Default>
Key:                     net
Source IP:               <Default>

RADIUS server host statistics:

Host            Accepts   Rejects   Timeouts
10.136.16.102   0         0         3
10.136.15.100   1         0         0         * Sever being used
10.136.15.101   0         0         0
```

Chapter 53

rarpd Commands

The **rarpd** commands let you configure and display information about Reverse Address Resolution Protocol (RARP) on the X-Pedition.

Command Summary

Table 42 lists the **rarpd** commands. The sections following the table describe the command syntax.

Table 42. rarpd commands

rarpd add hardware-address <*mac-address*> **ip-address** <*IPaddr*>
rarpd set interface <*name*>**|all server-ip <***IPaddr***>**
rarpd show interface|mappings

rarpd add

Purpose

Maps a MAC address to an IP address.

Format

rarp add hardware-address <*mac-address*> **ip-address** <*IPaddr*>

Mode

Configure

Description

The **rarpd add** command allows you to map a MAC address to an IP address for use with RARP. When a host makes a RARP request on the X-Pedition, and its MAC address has been mapped to an IP address with the **rarp add** command, the RARP server on the X-Pedition responds with the IP address that corresponds to the host's MAC address.

Parameters

hardware-address <*mac-address*>
Is a MAC address in the form *xx*:*xx*:*xx*:*xx*:*xx*:*xx* or *xxxxxx*:*xxxxxx*.

ip-address <*IPaddr*>
Is the IP address to be mapped to the MAC address.

Restrictions

None

Example

To map MAC address 00:C0:4F:65:18:E0 to IP address 10.10.10.10:

```
xp(config)# rarpd add hardware-address 00:C0:4F:65:18:E0 ip-address 10.10.10.10
```

rarpd set interface

Purpose

Specifies the interface(s) to which the X-Pedition's RARP server responds.

Format

rarpd set interface <*name*>|**all**

Mode

Configure

Description

The **rarpd set interface** command allows you to specify which interfaces the X-Pedition's RARP server responds to when sent RARP requests. You can specify individual interfaces or all interfaces.

Parameters

<*name*> Is the name of an interface.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

all Causes the RARP server to respond to RARP requests from all interfaces.

server-ip The server IP address to use in replies.

Restrictions

None.

Example

To cause the X-Pedition's RARP server to respond to RARP requests from interface int1:

```
xp(config)# rarpd set interface int1
```

rarpd show

Purpose

Displays information about the X-Pedition's RARP configuration.

Format

rarpd show interface|mappings

Mode

Enable

Description

The **rarpd show** command displays information about the configuration of the X-Pedition's RARP server. You can list the MAC-to-IP address mappings or the interfaces to which the X-Pedition responds to RARP requests.

Parameters

interface Lists the interfaces to which the X-Pedition responds to RARP requests.

mappings Displays the list of MAC-to-IP address mappings that was set with the **rarp add command**.

Restrictions

None.

Example

To display the RARP server's list of MAC-to-IP address mappings:

```
xp# rarpd show mappings
```

Chapter 54

rate-limit Command

The **rate-limit** commands allow you to define rate limits and apply them to IP interfaces or ports. There are several types of rate limiting supported:

- Per-flow rate limiting—limits individual flows to a specified rate.
- Flow-aggregate rate limiting—software-based rate limiting that limits an aggregation of flows (i.e., all flows that match an ACL) to a specific rate.
- VLAN rate limiting—rate limiting for traffic that enters or leaves a particular VLAN.
- Aggregate rate limiting—rate limiting for an aggregation of flows enabled on a per-line card basis.

Note: Since aggregate rate limiting is performed completely in hardware and must be enabled on a per-line card basis—if you enable aggregate rate limiting on a line card, you cannot use per-flow or flow-aggregate rate limiting with that card. Aggregate and flow-aggregate rate limiting are not supported on 802.1q trunk ports.

- Port-level rate limiting—rate limiting for individual ports.

Note: For a complete list of hardware and the features they support, consult the Release Notes on the Enterasys Networks web site: **www.enterasys.com**

Command Summary

Table 43 lists the **rate-limit** commands. The sections following the table describe the command syntax.

Table 43. rate-limit commands

rate-limit <*name*> **aggregate acl** <*acl list*> **rate** <*num*> **[drop-packets| no-action| lower-priority| lower-priority-except-control| tos-precedence-rewrite** <*num*>**| tos-precedence-rewrite-lower-priority** <*num*>**] [allocate-resources-during-apply| allocate-resources-during-traffic]| [burst-compensating]**
rate-limit <*name*> **apply interface** <*interface*>**| all**
rate-limit <*name*> **flow-aggregate acl** <*acl list*> **rate** <*rate*> **exceed-action** <*action*> **[sequence** <*number*>**]| [burst-compensating]| min-bandwidth** <*min-bw*>**| distribute-among** <*number-of-flows*>**]**
rate-limit <*name*> **input acl** <*acl list*> **rate** <*number*> **exceed-action drop-packets| set-priority-low| set-priority-medium| set-priority-high [sequence** <*number*>**]| [burst-compensating]**
rate-limit <*name*> **port-level input port** <*port list*> **rate** <*num*> **[drop-packets| no-action| lower-priority| lower-priority-except-control| tos-precedence-rewrite** <*num*>**| tos-precedence-rewrite-lower-priority** <*num*>**] [burst-compensating]**
rate-limit <*name*> **port-level slot** <*num*> **ignore-control-priority**
rate-limit <*name*> **port-level output port** <*port list*> **rate** <*num*> **drop-packets**
rate-limit show [all] |[policy-type flow-policies| flow-aggregate-policies| aggregate-policies| portlevel-policies|all]|[policy-name <*name*>**]|[interface** <*interface*>**]|[port-level port** <*port list*> **|all-port]| [port-level policy-name** <*name*>**]|[rate-limiting-mode]**
rate-limit <*name*> **vlan** <*name*> **port** <*port list*> | **all-ports destport** <*port list*> **|all-ports rate** <*num*> **exceed-action drop-packets| set-priority-low| set-priority-medium| set-priority-high [burst-compensating] [aggregate]**

rate-limit aggregate acl

Purpose

Defines an aggregate rate-limiting policy.

Format

rate-limit <*name*> **aggregate acl** <*acl list*> **rate** <*num*> **[drop-packets| no-action| lower-priority| lower-priority-except-control| tos-precedence-rewrite** <*num*>**| tos-precedence-rewrite-lower-priority** <*num*>**] [allocate-resources-during-apply| allocate-resources-during-traffic]| [burst-compensating** <*num*>**]**

Mode

Configure

Description

The **rate-limit aggregate acl** command allows you to specify the rate limiting policy for an aggregation of flows. An aggregation of flows is all the flows with the same ACLs. The rate limiting policy affects the whole aggregation and not an individual flow. Example of this type of policy is rate limiting traffic from one subnet to another. The line card to which you apply this command must be in Aggregate rate limiting mode. See *system enable aggregate-rate-limiting* on page 1215.

Parameters

<name>
The name of the rate limit.

acl <*acl list*>**|all-ports**
Specifies the ACL which will identify the flows to aggregate and rate limit. The keyword **all** specifies all ACLs.

rate <*num*>
Specifies the rate limit, in bps, for the flow. This value can be between 1000 and 1000000000.

drop-packets
This optional parameter specifies that if the rate-limit is exceeded, then packets will be dropped.

no-action
This optional parameter specifies that if the rate-limit is exceeded, then no action will be taken.

lower-priority
This optional parameter specifies that if the rate-limit is exceeded, then the packets priority is lowered.

lower-priority-except-control
This optional parameter specifies that if the rate-limit is exceeded, then the packets priority is lowered, except control packets.

tos-precedence-rewrite <*num*>
This optional parameter specifies that if the rate-limit is exceeded, then the tos precedence in the IP packet header will be rewritten to a specified value. This value can be between 0 and 7.

tos-precedence-rewrite-lower-priority <*num*>
This optional parameter specifies that if the rate-limit is exceeded, then the tos precedence in the IP packet header will be rewritten to a specified value and the packet priority will be lowered. This value can be between 0 and 7.

allocate-resources during-apply
This optional parameter allocates resources to the policy when its applied to an interface.

allocate-resources during-traffic
This optional parameter allocates resources to the policy when actual traffic flow is present.

burst-compensating <*num*>
When you choose the burst-compensating option, the X-Pedition invokes a different algorithm for calculating the rate limit values used by the hardware. This algorithm is better at compensating for *burst capacity*—the ability to maintain an average close to the specified rate—even with large bursts of traffic. The burst-compensating option is available on all rate-limiting policies except port-level output and requires that you enter a burst-compensator value of 1-100 to represent how much burst capacity (in Mbps) to build into the rate limit. For example, setting a low burst-compensator value on a rate limit policy to restrict the flow of an FTP server and client that are capable of very high transfer rates will choke off the flow and produce realized rates that are smaller than the specified rate limit. Conversely, setting a higher burst-compensator value than the flow's unrestricted capabilities will result in realized rates that are higher than the specified rate limit.

Note: Due to hardware constraints, the realized rates for rate limits set above 20 Mbps will become increasingly less consistent and accurate.

The X-Pedition uses a *credit bucket* bandwidth policing scheme to perform rate limiting. This policy creates a *credit bucket* and *time slice* in the Input Packet Processor hardware and, for a given rate limit, calculates a credit bucket size to represent the amount of traffic that can pass through the processor within a specific time period (i.e., the time slice value). When you specify the burst-compensating option, the credit bucket and time slice values are calculated to take into account traffic spikes or *bursts* and to achieve an average rate that is as close as possible to the specified rate. When the bucket is *filled* within the specified time slice, the X-Pedition will drop packets or change the priority of the packets, depending on the exceed-action specified. Because the burst compensating option allows you to create larger credit buckets and smaller time slices, you can prevent constricting the flow of rate-limited, bursty traffic.

Note: If you do not specify this option, rate-limiting will provide accurate results (to within 10-15%) for "smooth" traffic only (e.g., traffic created by a traffic generator). For example, if you use a small credit bucket and a large time slice, a burst of traffic can fill the bucket and cause the X-Pedition to drop traffic (until the time slice expires and refreshes the credit bucket). This can choke off the traffic rate. For bursty traffic such as TCP and most traffic running on a live network, use the burst-compensating option.

Restrictions

Aggregate and flow-aggregate rate limiting are not supported on 802.1q trunk ports.

Example

To define an aggregate rate limiting policy based on the ACL 'engacl':

```
xp(config)# rate-limit eng aggregate acl engacl rate 10000000 drop-packets allocate-resources
      during-apply
```

rate-limit apply

Purpose

Applies a rate limiting policy to an interface.

Format

rate-limit <*name*> **apply interface** <*interface*>| **all**

Mode

Configure

Description

The **rate-limit apply** command allows you to apply a previously-defined rate limiting policy to an interface.

Parameters

<*name*>
The name of the rate limiting policy.

interface <*interface*>|**all**
The name of the IP interface. The keyword **all** applies the policy to all IP interfaces.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

Port-level and VLAN policies do not use this command in conjunction with their policy commands.

Example

To apply the rate limiting policy ‘engacl’ to the interface ‘ip16’:

```
xp(config)# rate-limit engacl apply interface ip16
```

rate-limit flow-aggregate

Purpose

Used to specify a profile for software-based aggregate rate limiting.

Note: You cannot use Flow-Aggregate Rate Limiting on line cards that support Aggregate Rate Limiting. Flow-Aggregate Rate Limiting allows you to rate limit the aggregation of flows where the hardware does not support aggregate rate limiting.

Format

rate-limit <*name*> **flow-aggregate acl** <*acl list*> **rate** <*rate*> **exceed-action** <*action*> **[sequence** <*number*>**]| [burst-compensating]| min-bandwidth** <*min-bw*> **| distribute-among** <*number-of-flows*>**]**

Mode

Configure.

Description

The **rate-limit flow-aggregate** command is used to specify a profile for software-based aggregate rate limiting. Use the rate-limit apply command to apply the profile to an IP interface. The line card to which you apply this command must be in Per-flow rate limiting mode. See *system enable aggregate-rate-limiting* on page 1215.

rate-limit <*name*>
The name of the rate limiting policy. The maximum length for this name is 30 bytes.

acl <*acl list*>
The ACL(s) that define a rate limiting policy. The rate-limit flow-aggregate command disregards the permit/deny keywords in the ACL rule definition. However, it does look at all parameters in the ACL rule.

rate <*rate*>
The rate limit, in bps, for the aggregation of flows. The range for <rate> is 3000 to 1000000000.

exceed-action <*action*>
The action taken if the rate limit is exceeded.

drop-packets	Drop the packets.
set-priority-low	Set packet priority to low.
set-priority-medium	Set packet priority to medium.
set-priority-high	Set packet priority to high.

sequence <*number*>
The sequence number for this policy. The range for <number> is 1 to 65535.

burst-compensating <*num*>

When you choose the burst-compensating option, the X-Pedition invokes a different algorithm for calculating the rate limit values used by the hardware. This algorithm is better at compensating for *burst capacity*—the ability to maintain an average close to the specified rate—even with large bursts of traffic. The burst-compensating option is available on all rate-limiting policies except port-level output and requires that you enter a burst-compensator value of 1-100 to represent how much burst capacity (in Mbps) to build into the rate limit. For example, setting a low burst-compensator value on a rate limit policy to restrict the flow of an FTP server and client that are capable of very high transfer rates will choke off the flow and produce realized rates that are smaller than the specified rate limit. Conversely, setting a higher burst-compensator value than the flow's unrestricted capabilities will result in realized rates that are higher than the specified rate limit.

Note: Due to hardware constraints, the realized rates for rate limits set above 20 Mbps will become increasingly less consistent and accurate.

The X-Pedition uses a *credit bucket* bandwidth policing scheme to perform rate limiting. This policy creates a *credit bucket* and *time slice* in the Input Packet Processor hardware and, for a given rate limit, calculates a credit bucket size to represent the amount of traffic that can pass through the processor within a specific time period (i.e., the time slice value). When you specify the burst-compensating option, the credit bucket and time slice values are calculated to take into account traffic spikes or *bursts* and to achieve an average rate that is as close as possible to the specified rate. When the bucket is *filled* within the specified time slice, the X-Pedition will drop packets or change the priority of the packets, depending on the exceed-action specified. Because the burst compensating option allows you to create larger credit buckets and smaller time slices, you can prevent constricting the flow of rate-limited, bursty traffic.

Note: If you do not specify this option, rate-limiting will provide accurate results (to within 10-15%) for "smooth" traffic only (e.g., traffic created by a traffic generator). For example, if you use a small credit bucket and a large time slice, a burst of traffic can fill the bucket and cause the X-Pedition to drop traffic (until the time slice expires and refreshes the credit bucket). This can choke off the traffic rate. For bursty traffic such as TCP and most traffic running on a live network, use the burst-compensating option.

min-bandwidth <*min-bw*>

The minimum bandwidth for each flow. The range for <min-bw> is 3000 to 1000000000.

distribute-among <*number-of-flows*>

The number of flows among which freed bandwidth is distributed. The range for <number-of-flows> is 1 to 10. The default is 1.

Restrictions

Aggregate and flow-aggregate rate limiting are not supported on 802.1q trunk ports.

Example

The following example defines a rate limit profile client1 for traffic from the 10.10.10.0 network. Packets will be dropped if the rate limit of 10 million bps is exceeded, and each flow will have a minimum bandwidth of 10,000 bps.

```
xp(config)# acl 100 permit ip 10.10.10.0/24 any
xp(config)# rate-limit client1 flow-aggregate acl 100 rate
10000000 exceed-action drop-packets min-bandwidth
xp(config)# rate-limit apply client1 interface in1
```

rate-limit input acl

Purpose

Defines a policy to enable per flow rate limiting.

Format

rate-limit <*name*> **input acl** <*acl list*> **rate** <*number*> **exceed-action drop-packets| set-priority-low| set-priority-medium| set-priority-high [sequence** <*number*>**]| [burst-compensating]**

Mode

Configure

Description

The **rate-limit input** command allows you to specify the profile for per flow rate limiting by specifying IP ACLs, the rate limit, and the action to be performed if the rate limit is reached. You then use the **rate-limit apply** command to apply the rate limit to an IP interface. The line card to which you apply this command must be in Per-flow rate limiting mode. See *system enable aggregate-rate-limiting* on page 1215

Parameters

<name>
The name of the rate limiting policy. The maximum length for this name is 30 bytes or less.

input acl <*acl list*>
The ACL(s) that define a per flow rate limiting policy. The **rate-limit input** command disregards the **permit**/**deny** keywords in the ACL rule definition, however, it does look at all parameters in the ACL rule.

rate <*number*>
The rate limit, in bps, for the flow. This value can be between 1000 and 1000000000.

exceed-action <*action*>
The action to be taken if the rate limit is exceeded. Specify one of the following keywords:

drop-packets	Drop the packets.
set-priority-low	Set the priority to low.
set-priority-medium	Set the priority to medium.
set-priority-high	Set the priority to high.

sequence <*number*>
The sequence number for this policy. This value can be between 1 and 65535.

burst-compensating <*num*>
When you choose the burst-compensating option, the X-Pedition invokes a different algorithm for calculating the rate limit values used by the hardware. This algorithm is better at compensating for *burst capacity*—the ability to maintain an average close to the specified rate—even with large bursts of traffic. The burst-compensating option is available on all rate-limiting policies except port-level output and requires that you enter a burst-compensator value of 1-100 to represent how much burst capacity (in Mbps) to build into the rate limit. For example, setting a low burst-compensator value on a rate limit policy to restrict the flow of an FTP server and client that are capable of very high transfer rates will choke off the flow and produce realized rates that are smaller than the specified rate limit. Conversely, setting a higher burst-compensator value than the flow's unrestricted capabilities will result in realized rates that are higher than the specified rate limit.

Note: Due to hardware constraints, the realized rates for rate limits set above 20 Mbps will become increasingly less consistent and accurate.

The X-Pedition uses a *credit bucket* bandwidth policing scheme to perform rate limiting. This policy creates a *credit bucket* and *time slice* in the Input Packet Processor hardware and, for a given rate limit, calculates a credit bucket size to represent the amount of traffic that can pass through the processor within a specific time period (i.e., the time slice value). When you specify the burst-compensating option, the credit bucket and time slice values are calculated to take into account traffic spikes or *bursts* and to achieve an average rate that is as close as possible to the specified rate. When the bucket is *filled* within the specified time slice, the X-Pedition will drop packets or change the priority of the packets, depending on the exceed-action specified. Because the burst compensating option allows you to create larger credit buckets and smaller time slices, you can prevent constricting the flow of rate-limited, bursty traffic.

Note: If you do not specify this option, rate-limiting will provide accurate results (to within 10-15%) for "smooth" traffic only (e.g., traffic created by a traffic generator). For example, if you use a small credit bucket and a large time slice, a burst of traffic can fill the bucket and cause the X-Pedition to drop traffic (until the time slice expires and refreshes the credit bucket). This can choke off the traffic rate. For bursty traffic such as TCP and most traffic running on a live network, use the burst-compensating option.

Restrictions

None.

Example

To define a rate limit profile 'client1' for the ACL '100' that causes packets to be dropped if the rate limit of 10 million bps is exceeded:

```
xp(config)# rate-limit client1 input acl 100 rate-limit 10000000 exceed-action drop-packets
```

rate-limit port-level input

Purpose

Defines a rate limiting policy on a per-port basis for incoming traffic.

Format

rate-limit <*name*> **port-level input port** <*port list*> **rate** <*num*> **[drop-packets| no-action| lower-priority| lower-priority-except-control| tos-precedence-rewrite** <*num*>**| tos-precedence-rewrite-lower-priority** <*num*>**]**

Mode

Configure

Description

The **rate-limit port-level input** command allows you to specify the profile for a rate limiting policy on a per-port basis. This policy only affects incoming traffic to the port. The defined policy will only apply to that specific port and not an aggregation of flows. The line card to which you apply this command must be in Aggregate rate limiting mode. See *system enable aggregate-rate-limiting* on page 1215.

Parameters

<name>
The name of the rate limit.

port <*port list*>**| all-ports**
Specifies which ports to apply the rate-limiting policy. Specify **all-ports** to enable rate-limiting on all the ports.

rate <*num*>
Specifies the rate limit, in bps, for the flow. This value can be between 1000 and 1000000000.

drop-packets
This optional parameter specifies that if the rate-limit is exceeded, then packets will be dropped.

no-action
This optional parameter specifies that if the rate-limit is exceeded, then no action will be taken.

lower-priority
This optional parameter specifies that if the rate-limit is exceeded, then the packets priority is lowered.

lower-priority-except-control

This optional parameter specifies that if the rate-limit is exceeded, then the packets priority is lowered, except for control packets.

tos-precedence-rewrite <*num*>

This optional parameter specifies that if the rate-limit is exceeded, then the tos precedence in the IP packet header will be rewritten to a specified value. This value can be between 0 and 7.

tos-precedence-rewrite-lower-priority <*num*>

This optional parameter specifies that if the rate-limit is exceeded, then the tos precedence in the IP packet header will be rewritten to a specified value and the packet priority will be lowered. This value can be between 0 and 7.

burst-compensating <*num*>

When you choose the burst-compensating option, the X-Pedition invokes a different algorithm for calculating the rate limit values used by the hardware. This algorithm is better at compensating for *burst capacity*—the ability to maintain an average close to the specified rate—even with large bursts of traffic. The burst-compensating option is available on all rate-limiting policies except port-level output and requires that you enter a burst-compensator value of 1-100 to represent how much burst capacity (in Mbps) to build into the rate limit. For example, setting a low burst-compensator value on a rate limit policy to restrict the flow of an FTP server and client that are capable of very high transfer rates will choke off the flow and produce realized rates that are smaller than the specified rate limit. Conversely, setting a higher burst-compensator value than the flow's unrestricted capabilities will result in realized rates that are higher than the specified rate limit.

Note: Due to hardware constraints, the realized rates for rate limits set above 20 Mbps will become increasingly less consistent and accurate.

The X-Pedition uses a *credit bucket* bandwidth policing scheme to perform rate limiting. This policy creates a *credit bucket* and *time slice* in the Input Packet Processor hardware and, for a given rate limit, calculates a credit bucket size to represent the amount of traffic that can pass through the processor within a specific time period (i.e., the time slice value). When you specify the burst-compensating option, the credit bucket and time slice values are calculated to take into account traffic spikes or *bursts* and to achieve an average rate that is as close as possible to the specified rate. When the bucket is *filled* within the specified time slice, the X-Pedition will drop packets or change the priority of the packets, depending on the exceed-action specified. Because the burst compensating option allows you to create larger credit buckets and smaller time slices, you can prevent constricting the flow of rate-limited, bursty traffic.

Note: If you do not specify this option, rate-limiting will provide accurate results (to within 10-15%) for "smooth" traffic only (e.g., traffic created by a traffic generator). For example, if you use a small credit bucket and a large time slice, a burst of traffic can fill the bucket and cause the X-Pedition to drop traffic (until the time slice expires and refreshes the credit bucket). This can choke off the traffic rate. For bursty traffic such as TCP and most traffic running on a live network, use the burst-compensating option.

Restrictions

None.

Example

To define the port level rate limiting policy ‘department’ for the input port et.2.1 that causes packets to be dropped if the rate limit of 10 million bps is exceeded:

```
xp(config)# rate-limit department port-level input port et.2.1 rate 10000000 drop-packets
```

rate-limit port-level slot

Purpose

Sets rate limiting options for a module.

Format

rate-limit <*name*> **port-level slot** <*num*> **ignore-control-priority**

Mode

Configure

Description

The **rate-limit port-level slot** command allows you to set the output port level rate limiting policy to ignore the control priority traffic. This means that there will be no rate limiting for control priority traffic. Note that this policy does not actually try and rate limit the traffic.

Parameters

<name>
The name of the rate limiting policy.

slot <*num*>|**all**
Specifies the module or slot. This value can be between 0 and 32. Specify **all** to enable rate-limiting on all modules or slots.

ignore-control-priority
This optional parameter specifies that if the rate-limit is exceeded, then the control priority packets will not be dropped.

Restrictions

None.

Example

To define a rate limiting policy ‘dontdrop’ for all module or slots to prevent control priority packets from being dropped if the rate limit of 10 million bps is exceeded:

```
xp(config)# rate-limit dontdrop port-level slot all ignore-control-priority
```

rate-limit port-level output

Purpose

Defines a rate limiting policy on a per-port basis for outgoing traffic.

Format

rate-limit <*name*> **port-level output port** <*port list*> **rate** <*num*> **drop-packets**

Mode

Configure

Description

The **rate-limit port-level output** command allows you to specify the profile for a rate limiting policy on a per-port basis. This policy only affects outgoing traffic to the port, and the only exceed action available is dropping packets. The defined policy will only apply to that specific port and not an aggregation of flows.

Parameters

<name>
The name of the rate limit.

port <*port list*>|**all-ports**
Specifies which ports to apply the rate-limiting policy. Specify **all-ports** to enable rate-limiting on all the ports.

rate <*number*>
The rate limit, in bps, for the flow. This value can be between 1000-10000000.

drop-packets
This optional parameter specifies that if the rate-limit is exceeded, then packets will be dropped.

Restrictions

None.

Example

To define a rate limit policy ‘department’ for the output port et.2.1 that causes packets to be dropped if the rate limit of 10 million bps is exceeded:

```
xp(config)# rate-limit department port-level output port et.2.1 rate 10000000 drop-packets
```

rate-limit show

Purpose

Displays rate limiting policies.

Format

rate-limit show [all] |[policy-type flow-policies| flow-aggregate-policies| aggregate-policies| portlevel-policies|all]|[policy-name <*name*>**]|[interface** <*interface*>**]|[port-level port** <*port list*> **|all-port] | [port-level policy-name** <*name*>**]|[rate-limiting-mode]**

Mode

Enable

Description

The **rate-limit show** command shows information about rate limiting policies.

Parameters

all
Displays information on all rate limit policies configured on the X-Pedition.

policy-type
The type of the rate limit policy. The keyword **all** shows all rate limit types. You can specify the following types of policies:

flow-policies	All flow policies
flow-aggregate-policies	All software-based flow-aggregate policies
aggregate-policies	All aggregate policies
portlevel-policies	All port level policies
all	All policies

policy-name <*name*> | **all**
The name of the rate limiting policy. The keyword **all** shows all rate limit policies.

interface <*interface*> | **all**
The name of the IP interface. The keyword **all** shows rate limiting policies for all IP interfaces.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

port-level port <*port list*> | **all-ports**
The name of the port. The keyword **all-ports** shows rate limiting policies for all ports.

port-level policy-name <*name*>
The name of the rate limiting policy name.

rate-limiting-mode
Shows the current rate limiting mode, whether per-flow rate limiting or aggregate rate limiting.

Restrictions

None.

Example

To show all configured rate limit policies:

```
xp# rate-limit show all

--------------------------------------------------------------------------------
Rate Limit Policy name : rlpol ❶
Applied Interfaces : if0 ❷

 ❸        ❹              ❺           ❻        ❼      ❽     ❾
ACL     Source IP/Mask    Dest. IP/Mask    SrcPort   DstPort   TOS  Prot
---     --------------    -------------    --------- --------- ---- ---
100     10.212.10.11/32   anywhere         any       any       any  IP
200     10.212.10.12/32   anywhere         any       any       any  IP
300     10.212.10.13/32   anywhere         any       any       any  IP
400     10.212.10.14/32   anywhere         any       any       any  IP
500     10.212.10.10/32   anywhere         any       any       any  IP

 ❿ ⓫         ⓬           ⓭
Seq ACL      Rate Limit Exceed Action
--- ----     ---------   -------------
10  100      26000       Low
10  200      26000       Low
10  300      26000       Low
10  400      26000       Low
10  500      26000       Low
```

Legend:

1. The name of the rate limit.
2. The IP interface to which the rate limit is applied.
3. The name of the ACL(s) that define the rate limit.
4. The source address and filtering mask specified by the ACL.

5. The destination address and filtering mask specified by the ACL.
6. The number of the TCP or UDP source port.
7. The number of the TCP or UDP destination port.
8. The Type of Service value.
9. The protocol for the ACL.
10. The sequence number for this policy.
11. The name of the ACL.
12. The rate limit for the flow.
13. The action to be taken if the rate limit is reached: packets can be dropped or the priority set to low, medium, or high.

rate-limit vlan port

Purpose

Defines a rate limit policy on a per-vlan basis for *incoming* or per-port *outgoing* traffic.

Format

rate-limit <*name*> **vlan** <*name*> **port** <*port list*> | **all-ports destport** <*port list*> |**all-ports rate** <*num*> **exceed-action drop-packets| set-priority-low| set-priority-medium| set-priority-high [burst-compensating] [aggregate]**

Mode

Configure

Description

The rate-limit vlan port command allows you to specify the rate limiting policy for incoming traffic on a particular vlan, or outgoing traffic on a port belonging to a specific vlan. Like Port rate limiting policies, you do not specify an ACL when defining this type of policy. Vlan rate limiting policies do not need to be applied to an interface and take affect when they are created. The line card to which you apply this command must be in Per-flow rate limiting mode. See *system enable aggregate-rate-limiting* on page 1215.

Note: When you use the vlan rate limiting policy to limit outgoing traffic on a port belonging to a specified vlan, multicast and broadcast traffic will not be rate limited.

Parameters

<*name*>
The name of the rate limit.

vlan <*name*>
The name of the vlan.

Note: The X-Pedition will display VLAN names up to 32 characters in length.

port <*port-list*>**|all-ports**
Specifies which ports to apply the rate-limiting policy. Specify all-ports to enable rate-limiting on all the ports belonging to the specified vlan.

destport <*port-list*>**|all-ports**
Specifies which exit ports to apply the rate-limiting policy. Specify all-ports to enable rate-limiting on all the exit ports belonging to the specified vlan.

rate <*num*>
Specifies the rate limit, in bps, for the vlan or vlan port. This value can be between 1000 and 1000000000.

exceed-action <*action*>
The action to be taken if the rate limit is exceeded. Specify one of the following keywords:

drop-packets	Drop the packets
set-priority-low	Set the priority to low
set-priority-medium	Set the priority to medium
set-priority-high	Set the priority to high

burst-compensating <*num*>
When you choose the burst-compensating option, the X-Pedition invokes a different algorithm for calculating the rate limit values used by the hardware. This algorithm is better at compensating for *burst capacity*—the ability to maintain an average close to the specified rate—even with large bursts of traffic. The burst-compensating option is available on all rate-limiting policies except port-level output and requires that you enter a burst-compensator value of 1-100 to represent how much burst capacity (in Mbps) to build into the rate limit. For example, setting a low burst-compensator value on a rate limit policy to restrict the flow of an FTP server and client that are capable of very high transfer rates will choke off the flow and produce realized rates that are smaller than the specified rate limit. Conversely, setting a higher burst-compensator value than the flow's unrestricted capabilities will result in realized rates that are higher than the specified rate limit.

Note: Due to hardware constraints, the realized rates for rate limits set above 20 Mbps will become increasingly less consistent and accurate.

The X-Pedition uses a *credit bucket* bandwidth policing scheme to perform rate limiting. This policy creates a *credit bucket* and *time slice* in the Input Packet Processor hardware and, for a given rate limit, calculates a credit bucket size to represent the amount of traffic that can pass through the processor within a specific time period (i.e., the time slice value). When you specify the burst-compensating option, the credit bucket and time slice values are calculated to take into account traffic spikes or *bursts* and to achieve an average rate that is as close as possible to the specified rate. When the bucket is *filled* within the specified time slice, the X-Pedition will drop packets or change the priority of the packets, depending on the exceed-action specified. Because the burst compensating option allows you to create larger credit buckets and smaller time slices, you can prevent constricting the flow of rate-limited, bursty traffic.

Note: If you do not specify this option, rate-limiting will provide accurate results (to within 10-15%) for "smooth" traffic only (e.g., traffic created by a traffic generator). For example, if you use a small credit bucket and a large time slice, a burst of traffic can fill the bucket and cause the X-Pedition to drop traffic (until the time slice expires and refreshes the credit bucket). This can choke off the traffic rate. For bursty traffic such as TCP and most traffic running on a live network, use the burst-compensating option.

aggregate
Specifying the aggregate option will aggregate all flows matching this policy and distribute the specified rate among these flows. If you do not specify this option, each matching flow will be limited to the full rate.

Restrictions

When using the vlan rate limiting policy to limit outgoing traffic on a port belonging to a specified vlan, multicast and broadcast traffic will not be rate limited.

Example

To define a rate limit policy “client1” for the vlan “red” that causes packets to be dropped if the rate limit of 10 million bps is exceeded.

```
X-Pedition(config)# rate-limit client1 vlan red port all-ports rate 10000000 exceed-action drop-packets
```

Chapter 55

rdisc Commands

The **rdisc** commands allow you to configure router advertisement on the X-Pedition.

Command Summary

Table 44 lists the **rdisc** commands. The sections following the table describe the command syntax.

Table 44. rdisc commands

rdisc add address <*hostname-or-ipaddr*>
rdisc add interface all
rdisc set address <*ipaddr*> **type multicast|broadcast advertise enable|disable preference** <*number*> **|ineligible**
rdisc set interface all min-adv-interval <*number*> **max-adv-interval** <*number*> **lifetime** <*number*>
rdisc show all
rdisc start
rdisc stop

rdisc add address

Purpose

Defines the IP address(es) that are to be included in router advertisements sent by the X-Pedition.

Format

rdisc add address <*hostname-or-ipaddr*>

Mode

Configure

Description

The **rdisc add address** command lets you define addresses to be included in router advertisements. If you configure this command, only the specified hostname(s) or IP address(es) are included in the router advertisements.

Parameters

<*hostname-or-ipaddr*>	Defines the hostname or IP address(es) to be included in the router advertisements.

Restrictions

None.

Example

To define an address to be included in router advertisements:

```
xp(config)# rdisc add address 10.10.5.254
```

rdisc add interface

Purpose

Enables router advertisement on all interfaces.

Format

rdisc add interface all

Mode

Configure

Description

The **rdisc add interface** command lets you enable router advertisement on all interfaces. By default, all addresses on the interface are included in router advertisements sent by the X-Pedition. If you want to have only specific addresses included in router advertisements, use the **rdisc add address** command to specify those addresses.

Parameters

all	Enables router advertisement on all interfaces. By default, router advertisement is disabled on all interfaces.

Restrictions

None.

Example

To enable router advertisement on all interfaces:

```
xp(config)# rdisc add interface all
```

rdisc set address

Purpose

Configures router advertisement parameters that apply to a specific address.

Format

rdisc set address <*ipaddr*> **type multicast|broadcast advertise enable|disable preference** <*number*> |**ineligible**

Mode

Configure

Description

The **rdisc set address** command lets you specify the type of router advertisement in which the address is included and the preference of the address for use as a default route.

Parameters

<*ipaddr*> Specifies the IP address.

type multicast|broadcast
Specifies the type of router advertisement in which the IP address is to be included:

- **multicast** Specifies that the IP address should only be included in a multicast router advertisement. This is the default.
- **broadcast** Specifies that the IP address should only be included in a broadcast router advertisement, even if IP multicast is available.

advertise enable|disable
Specifies whether the IP address is included in the router advertisements:

- **enable** Include the IP address in router advertisements. This is the default.
- **disable** Do not include the IP address in router advertisements.

preference <*number*>|**ineligible**
Specifies the degree of preference of the IP address as a default route. The higher the value, the more preference. If the IP address is ineligible to be a default route, specify **ineligible**. The default value is 0.

Restrictions

None

Examples

To specify that an address be included only in broadcast router advertisements and that the address is ineligible to be a default route:

```
xp(config)# rdisc set address 10.20.36.0 type broadcast preference ineligible
```

rdisc set interface

Purpose

Configures router advertisement parameters applying to all interfaces.

Format

rdisc set interface all min-adv-interval <*number*> **max-adv-interval** <*number*> **lifetime** <*number*>

Mode

Configure

Description

The **rdisc set interface** command lets you specify the intervals between the sending of router advertisements and the lifetime of addresses sent in a router advertisement.

Parameters

all Applies parameters to all interfaces.

min-adv-interval <*number*>
Specifies the minimum time, in seconds, allowed between the sending of unsolicited broadcast or multicast router advertisements. This value can be between 3-1800. The default is 0.75 times the **max-adv-interval** value.

max-adv-interval <*number*>
Specifies the maximum time, in seconds, allowed between the sending of unsolicited broadcast or multicast router advertisements. This value can be between 4-1800. The default value is 600 seconds.

lifetime <*number*>
Specifies the lifetime, in seconds, of addresses in a router advertisement. This value can be between 4-9000. The default is 3 times the **max-adv-interval** value.

Restrictions

None

Examples

To specify the maximum time between the sending of router advertisements on all interfaces:

```
xp(config)# rdisc set interface all max-adv-interval 1200
```

Note that since the **min-adv-interval** and **lifetime** parameters were not specified, the default values for those parameters become 900 seconds and 3600 seconds, respectively.

rdisc show

Purpose

Shows the state of router discovery on the X-Pedition.

Format

rdisc show all

Mode

Enable

Description

The **rdisc show** command shows the state of router discovery on the X-Pedition.

Parameters

all
Displays all router discovery information.

Restrictions

None.

Examples

To display router discovery information:

```
xp# rdisc show all


Task State: <Foreground NoResolv NoDetach> ❶

   Send buffer size 2048 at 812C68F8
   Recv buffer size 2048 at 812C60D0

Timers:

   RouterDiscoveryServer Priority 30

      RouterDiscoveryServer_xp2_xp3_IP <OneShot>
        last: 10:17:21 next: 10:25:05 ❷

Task RouterDiscoveryServer:
   Interfaces:
      Interface xp2_xp3_IP: ❸
         Group 224.0.0.1: ❹
            minadvint 7:30 maxadvint 10:00 lifetime 30:00 ❺

            Address 10.10.5.254: Preference: 0 ❻

   Interface policy:
      Interface xp2_xp3_IP* MaxAdvInt 10:00 ❼
```

Legend:

1. Information about the RDISC task.
2. Shows when the last router advertisement was sent and when the next advertisement will be sent.
3. The interface on which router advertisement is enabled.
4. Multicast address.
5. Current values for the intervals between the sending of router advertisements and the lifetime of addresses sent in a router advertisement.
6. IP address that is included in router advertisement. The preference of this address as a default route is 0, the default value.
7. Shows configured values for the specified interface.

rdisc start

Purpose

Starts router discovery on the X-Pedition.

Format

rdisc start

Mode

Configure

Description

The **rdisc start** command lets you start router discovery on the X-Pedition. When router discovery is started, the X-Pedition multicasts or broadcasts periodic router advertisements on each configured interface. The router advertisements contain a list of addresses on a given interface and the preference of each address for use as the default route on the interface. By default, router discovery is disabled.

Parameters

None.

Restrictions

None

rdisc stop

Purpose

Stops router discovery.

Format

rdisc stop

Mode

Configure

Description

The **rdisc stop** command stops router discovery on the X-Pedition, thereby stopping router advertisements from being sent out.

Parameters

None.

Restrictions

None

Chapter 56

reboot Command

The **reboot** command reboots the X-Pedition.

Format

reboot

Mode

Enable.

Parameters

None.

Restrictions

None.

Chapter 57

rip Commands

The Routing Information Protocol, Version 1 and Version 2 (RIPv1 and RIPv2), is the most commonly used interior gateway protocol. RIP selects the route with the lowest metric as the best route. The metric is a hop count representing the number of gateways through which data must pass in order to reach its destination. The longest path that RIP accepts is 15 hops. If the metric is greater than 15, a destination is considered unreachable and the X-Pedition discards the route. RIP assumes that the best route is the one that uses the fewest gateways, that is, the shortest path. RIPv1 is described in RFC 1058 and RIPv2 is described in RFC 1723.

Note: The X-Pedition supports a maximum of 120 RIP interfaces.

Command Summary

Table 45 lists the **rip** commands. The sections following the table describe the command syntax.

Table 45. rip commands

rip add interface <*interfacename-or-IPaddr*>| **source-gateways**| **trusted-gateways** <*hostname-or-IPaddr*>
rip set auto-summary disable | enable
rip set broadcast-state always | choose | never
rip set check-zero disable | enable
rip set check-zero-metric disable | enable
rip set default-metric <*num*>
rip set interface <*interfacename-or-IPaddr*> **| all [receive-rip enable | disable] [send-rip enable | disable] [metric-in** <*num*>**] [metric-out** <*num*>**] [version 1|version 2 [type broadcast|multicast]] authentication-method [none|(simple|md5 key-chain** <*num-or-string*>**)]**

Table 45. rip commands (Continued)

rip set max-routes <*num***>**
rip set multipath off
rip set poison-reverse disable | enable
rip set preference <*num*>
rip show <*option-list*>
rip start
rip stop
rip trace [packets| request| response| local-options] [detail] [send| receive]

rip add

Purpose

Adds RIP entities.

Note: By default, RIP is disabled on all X-Pedition interfaces. To enable RIP on an interface, you must use the **rip add interface** command.

Format

rip add interface <*interfacename-or-IPaddr*>| **source-gateways|**
trusted-gateways <*hostname-or-IPaddr*>

Mode

Configure

Description

The **rip add** command lets you add the following RIP entities:

- Interfaces that will run RIP
- Routers that send RIP updates directly, rather than through broadcast or multicast
- Trusted gateways, from which the X-Pedition will accept RIP updates. when you add trusted gateways, the X-Pedition does not accept RIP updates from sources other than those trusted gateways.

Parameters

interface <*interfacename-or-IPaddr*>
Informs the RIP process about the specified interfaces. You can specify a list of interface names or IP addresses or use the **all** keyword to specify all interfaces.

Note: The X-Pedition supports a maximum of 120 RIP interfaces.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

source-gateways
Adds a router that sends RIP updates directly, rather than using broadcasts or multicasts. You can specify a single interface name or IP address.

Note: Updates to source gateways are not affected by the RIP packet transmission state of the interface. Enterasys recommends that you use alphabetic characters when defining

interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

trusted-gateway <*hostname-or-IPaddr*>

The hostname or IP address of the source or trusted gateway. Adds a trusted source for RIP updates. When you add trusted gateways, the X-Pedition will not accept RIP updates from any sources except the trusted gateways. You can specify a single interface name or IP address.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

The X-Pedition supports a maximum of 120 RIP interfaces.

rip set auto-summary

Purpose

Enables automatic summarization and redistribution of RIP routes.

Format

rip set auto-summary disable| enable

Mode

Configure

Description

The **rip set auto-summary** command specifies that routes to subnets should be automatically summarized by the classful network boundary and redistributed into RIP.

Note: The **rip set auto-summary** command must be enabled if the router will act as a border gateway using RIP Version 1.

Parameters

disable | enable
Enables or disables automatic summarization and redistribution of RIP routes.

Restrictions

None.

rip set broadcast-state

Purpose

Determines if RIP packets will be broadcast regardless of the number of interfaces present. This is useful when propagating static routes or routes learned from another protocol into RIP. In some cases, the use of broadcast when only one network interface is present can cause data packets to traverse a single network twice.

Format

rip set broadcast-state always | choose | never

Mode

Configure

Description

The **rip set broadcast-state** command specifies whether the X-Pedition broadcasts RIP packets regardless of the number of interfaces present.

Note: The X-Pedition supports a maximum of 120 RIP interfaces.

Parameters

always | choose | never
Specifies whether the X-Pedition broadcasts RIP packets regardless of the number of interfaces present. Specify one of the following:

always Always sends RIP broadcasts regardless of the number of interfaces present.

choose Sends RIP broadcasts only if more than one interface is configured on the X-Pedition. This is the default state.

never Never sends RIP broadcasts on attached interfaces.

Restrictions

The X-Pedition supports a maximum of 120 RIP interfaces.

rip set check-zero

Purpose

Specifies whether RIP should make sure that reserved fields in incoming RIP V1 packets are zero. Normally, RIP will reject packets where the reserved fields are non-zero.

Format

rip set check-zero disable | enable

Mode

Configure

Description

The **rip set check-zero** command specifies whether RIP should make sure that reserved fields in incoming RIP V1 packets are zero. RIP will reject packets where the reserved fields are non-zero.

- If you use the **disable** keyword, RIP does not check the reserved field.
- If you use the **enable** keyword, RIP on the X-Pedition checks to ensure that the reserved fields in incoming RIP packets are zero. If the reserved field in a RIP packet is not zero, the X-Pedition discards the packet. This is the default state.

Parameters

disable | enable
Enables or disables checking of the reserved field.

Restrictions

None.

rip set check-zero-metric

Purpose

Specifies whether RIP should accept routes with a metric of zero. Normally, RIP will reject routes with a metric of zero.

Format

rip set check-zero-metric disable | enable

Mode

Configure

Description

The **rip set check-zero-metric** command specifies whether RIP should accept routes with a metric of zero. This may be necessary for interoperability with other RIP implementations that send routes with a metric of zero.

- If you use the **disable** keyword, RIP accepts routes that have a metric of zero and treats them as though they were received with a metric of 1.
- If you use the **enable** keyword, RIP rejects routes that have a metric of zero. This is the default state.

Parameters

disable | enable
Enables or disables acceptance of RIP routes that have a metric of zero.

Restrictions

None.

rip set default-metric

Purpose

Defines the metric used when advertising routes via RIP that were learned from other protocols. If not specified, the default value is 16 (unreachable). This choice of values requires you to explicitly specify a metric in order to export routes from other protocols into RIP. This metric may be overridden by a metric specified in the export command.

Note: The metric 16 is equivalent in RIP to “infinite” and makes a route unreachable. You must set the default metric to a value other than 16 in order to allow the X-Pedition to export routes from other protocols such as OSPF and BGP-4 into RIP.

Format

rip set default-metric <*num*>

Mode

Configure

Description

The **rip set default metric** command defines the metric used when advertising routes via RIP that were learned from other protocols.

Parameters

<*num*> Specifies the metric. Specify a number from 1 – 16. The default is 16.

Restrictions

None.

rip set interface

Purpose

Set the RIP state, version, type of update messages, metric and authentication scheme used for each interface running RIP.

Format

rip set interface <*interfacename-or-IPaddr*> **| all [advertise-classfull enable | disable] [receive-rip enable | disable] [send-rip enable | disable] [metric-in** <*num*>**] [metric-out** <*num*>**] [version 1|version 2 [type broadcast|multicast]] [authentication-method none|(simple|md5 key-chain** <*num-or-string*>**)]**

Mode

Configure

Description

The **rip set interface** command lets you set the following parameters for RIP interfaces:

- Whether the interface will accept RIP updates
- Whether the interface will send RIP updates
- The RIP version (RIP V1 or RIP V2)
- The packet type used for RIP V2 updates (broadcast or multicast)
- The metric added to incoming RIP updates
- The metric added to outgoing RIP updates
- The key-chain for RIP update authentication
- The authentication method used for RIP updates (none, simple, or MD5)

Parameters

<*interfacename-or-IPaddr*>**|all**

The interface names or IP addresses of the interfaces for which you are setting RIP parameters. Specify the **all** keyword if you want to set RIP parameters for all IP interfaces on the X-Pedition.

Note: The X-Pedition supports a maximum of 120 RIP interfaces.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

advertise-classfull enable | disable
This command is used to announce a classfull network onto a subnetted RIP Version 1 interface having the same classfull network.

receive-rip enable|disable
Specifies whether the interface(s) can receive RIP updates. Specify **enable** if you want to receive RIP updates on the interface. Otherwise, select **disable**.

The default is **enable**.

Note: This option affects RIP updates sent from trusted gateways. If you specify **disable**, the X-Pedition will not receive any RIP updates, including those sent from trusted gateways. If you specify **enable** *and* you have set up trusted gateways, the X-Pedition will accept updates only from those trusted gateways.

send-rip enable|disable
Specifies whether the interface(s) can send RIP updates. Specify **enable** if you want to send RIP updates from this interface. Otherwise, specify **disable**.

The default is **enable**.

Note: This option does not affect the sending of updates to source gateways.

metric-in <*num*>
Specifies a metric that the interface adds to incoming RIP routes before adding them to the interface table. Specify a metric from 1 – 16. Use this option to make the X-Pedition prefer RIP routes learned from the specified interfaces less than RIP routes from other interfaces. The default is 1.

metric-out <*num*>
Specifies a metric that the interface adds to outgoing RIP routes sent through the specified interfaces. The default is 0. Use this option to make other routers prefer other sources of RIP routes over this router.

version 1|version 2 [type broadcast|multicast]
Specifies the RIP version used on the interface(s).

broadcast
Causes RIP V2 packets that are RIP V1-compatible to be broadcast on this interface.

multicast
Causes RIP V2 packets to be multicasted on this interface; this is the default.

authentication-method none|(simple|md5 key-chain <*num-or-string*>**)**
The authentication method the interface uses to authenticate RIP updates. Specify one of the following:

none
The interface does not use any authentication.

simple
The interface uses a simple password in which an authentication key of up to 8 characters is included in the packet.

md5
The interface uses MD5 authentication. This method uses the MD5 algorithm to create a crypto-checksum of a RIP packet and an authentication key of up to 16 characters.

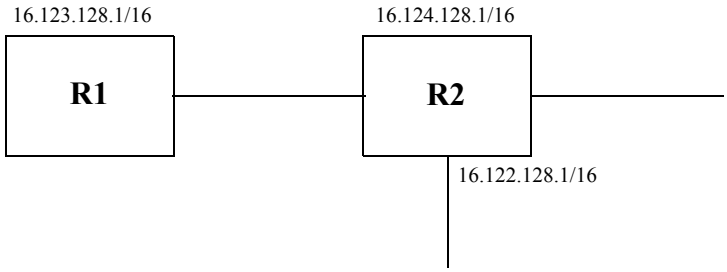
Note: If you choose the simple or md5 authentication method, you must also specify a key-chain identifier using the key-chain option.

key-chain <*num-or-string*>
The identifier of the key-chain containing the authentication keys. This parameter applies only if you specified simple or md5 for the authentication type.

Restrictions

The X-Pedition supports a maximum of 120 RIP interfaces.

Example



In this example, router R1 has the following three interfaces:

1. It is connected to router R2 over interface 16.123.128.1/16. It is running RIP version 1 on this interface.
2. It has two other interfaces with the following addresses (16.124.128.1/16, 16.122.128.1/16).
3. Router R1 the entire class A network (16.0.0.0/8) behind it.

By default, router R1 would not announce a classful network (16.0.0.0/8) over a subnet (16.123.128.1/16). If that is something which is desired, then the below given command should be entered.

```
rip set interface 16.123.128.1 advertise-classfull enable | disable
```

Typically, a user would enable automatic summarization for RIP. This would create an implicit aggregate 16.0.0.0/8. If it is desired, that this classfull network is announced over a subnetted RIP Version 1 interface, then the above command should be entered.

rip set max-routes

Purpose

Defines the maximum number of RIP routes.

Format

rip set default-metric <*num*>

Mode

Configure

Description

The **rip set max-routes** command defines the maximum number of RIP routes that can be maintained by the Routing Information Base (RIB).

Parameters

<*num*> Specifies the maximum number of routes. Specify a number from 1 – 4. The default is 4.

Restrictions

None.

rip set multipath

Purpose

Disables multipath route calculation for RIP routes.

Format

rip set multipath off

Mode

Configure

Description

The **rip set multipath** command disables multipath route calculation for RIP routes. No multipath forwarding occurs when this command is used.

Parameters

off Disables multipath route calculation.

Restrictions

If you negate this command from the configuration file, the X-Pedition will not automatically recreate multipath routes. To recreate multipath routes, stop and restart RIP.

rip set poison-reverse

Purpose

Enables poison reverse on all X-Pedition interfaces.

Format

rip set poison-reverse disable | enable

Mode

Configure

Description

The **rip set poison-reverse** command allows you to enable or disable poison reverse on all X-Pedition interfaces. The X-Pedition supports poison reverse as specified by RFC 1058.

Note: Turning on poison reverse will approximately double the amount of RIP updates.

Parameters

disable | enable
Enables or disables poison reverse on the X-Pedition.

Restrictions

None.

rip set preference

Purpose

Sets the preference of routes learned from RIP. The default preference is 100. This preference may be overridden by a preference specified in the import command.

Format

rip set preference <*num*>

Mode

Configure

Description

The **rip set preference** command sets the preference for destinations learned through RIP. The preference you specify applies to all IP interfaces for which RIP is enabled on the X-Pedition. The default preference is 100. You can override this preference by specifying a different preference in an import policy.

Parameters

<*num*> Specifies the preference. Specify a number from 0 – 255. The default is 100. Lower numbers have higher preference.

Restrictions

None.

rip show

Purpose

Display RIP information.

Format

rip show <*option-list*>

Mode

Enable

Description

The **rip show** command displays RIP information.

Parameters

<*option-list*>
Specifies the RIP dump information you want to display. Specify one or more of the following:

all
Displays all RIP tables.

globals
Displays RIP globals.

timers
Displays RIP timers.

RIP Responses The network sends and receives RIP responses each time it encounters a change to the network (e.g., a new interface) until it updates the routing table to reflect the change(s). If you have an unstable network, you may be inundated with RIP response packets that indicate a network change. In such a case, you will want to use a Flash Timer.

Flash Timer A Flash Timer consolidates the list of RIP response packets and sends all of them randomly every 1-5 seconds instead of as they appear. Once the routing table update is complete, the timer remains inactive until the next routing table change or until 30 seconds passes and the Update Timer solicits the network status.

Update Timer Every 30 seconds, regardless of whether or not your routing table changed, the update timer returns the network status.

Age Timer In many cases, a route or series of routes will expire after a specific time period. The Age Timer is an automated process used to delete any expired routes. The Age

Timer will fire every 3 minutes unless a route will expire within the 3-minute period—in this case, the timer will change to coincide with the route expiration.

Show Command The show command lists the status of each timer. The examples below depict the status changes over time:

```
xp(enable)# rip show timers

Timers:
-----------

Timer                    State     Last      Next      Intvl   Jitter   Flags
-------------------------------------------------------------------------------------------------
RIP.0.0.0.0+520_Flash    Inactive  -         -         -       -        Inactive
RIP.0.0.0.0+520_Update   Active    15:40:35  15:41:05  30      -
RIP.0.0.0.0+520_Age      Active    15:39:00  15:42:00  -       -        OneShot
```

```
xp(enable)# rip show timers

Timers:
-----------

Timer                    State     Last      Next      Intvl   Jitter   Flags
-------------------------------------------------------------------------------------------------
RIP.0.0.0.0+520_Flash    Active    15:41:05  15:41:09  4       -
RIP.0.0.0.0+520_Update   Active    15:41:05  15:41:35  30      -
RIP.0.0.0.0+520_Age      Active    15:39:00  15:42:00  -       -        OneShot
```

```
xp(enable)# rip show timers

Timers:
-----------

Timer                    State     Last      Next      Intvl   Jitter   Flags
-------------------------------------------------------------------------------------------------
RIP.0.0.0.0+520_Flash    Inactive  -         -         -       -        Inactive
RIP.0.0.0.0+520_Update   Active    15:41:05  15:41:35  30      -
RIP.0.0.0.0+520_Age      Active    15:39:00  15:42:00  -       -        OneShot
```

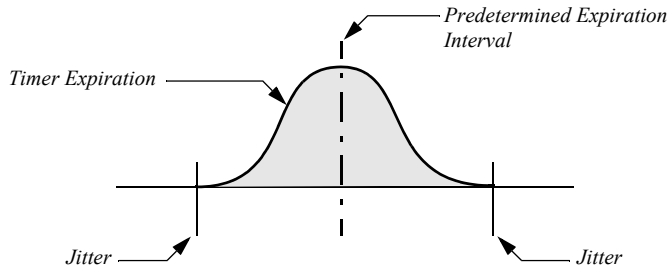
State (active or inactive) The state of the timer—scheduled timers are inactive.

Last and Next A timestamp of the last and next time the timer was and will be used.

Interval The time (in seconds) between timer activities that are periodic as opposed to OneShot. The Age Timer is considered a OneShot because the interval may not be consistent.

Jitter To prevent peak volumes of network activity associated with timers, the network uses a *Jitter*. Jitter is displayed when the next expiration will occur at the

predetermined interval plus or minus a few microseconds. This helps reduce network congestion. The example below depicts a jitter:



Flags Indicate whether a timer is inactive or a OneShot.

interface
Displays RIP interfaces.

active-gateways
Displays active gateways running RIP.

interface-policies
Displays RIP interface policies.

import-policies
Displays RIP import policies.

export-policies
Displays RIP export policies.

Restrictions

None.

rip start

Purpose

Start RIP on the X-Pedition.

Note: RIP is disabled by default.

Format

rip start

Mode

Configure

Description

The **rip start** command starts RIP on all IP interfaces on the X-Pedition for which RIP is enabled.

Parameters

None.

Restrictions

None.

rip stop

Purpose

Stop RIP on the X-Pedition.

Format

rip stop

Mode

Configure

Description

The **rip stop** command stops RIP on all IP interfaces on the X-Pedition for which RIP is enabled.

Parameters

None.

Restrictions

None.

rip trace

Purpose

Trace RIP packets.

Format

rip trace [packets| request| response| local-options] [detail | send| receive]

Mode

Configure

Description

The **rip trace** command traces the following sets of RIP packets:

- RIP request packets sent or received by the X-Pedition
- RIP response packets sent or received by the X-Pedition

Depending on the options you specify, you can trace all packets, request packets only, or receive packets only. In addition, you can select to trace the request packets, receive packets, or both that are sent by the X-Pedition, received by the X-Pedition, or all packets (both sent packets and received packets).

Parameters

packets Traces all RIP packets, both request packets and response packets. This is the default.

request Traces only request packets, such as REQUEST, POLL and POLLENTRY packets.

response Traces only response packets.

For the **packets**, **request**, and **response** parameters, you can optionally specify one of the following:

detail Shows detailed information about the traced packets.

receive Shows information about traced RIP packets received by the X-Pedition.

send Shows information about traced RIP packets sent by the X-Pedition.

Note: The default is to show both send and receive packets.

local-options Sets trace options for this protocol only. These trace options are inherited from those set by the **ip-router global set trace options** command, or you can override them here. Specify one or more of the following:

all	Turns on all tracing.
general	Turns on normal and route tracing.
state	Traces state machine transitions in the protocols.
normal	Traces normal protocol occurrences.

Note: Abnormal protocol occurrences are always traced.

policy	Traces application of protocol and user-specified policies to routes being imported and exported.
task	Traces system processing associated with this protocol or peer.
timer	Traces timer usage by this protocol or peer.
route	Traces routing table changes for routes installed by this protocol or peer.

Restrictions

None.

Chapter 58

rmon Commands

The **rmon** commands let you display and set parameters for RMON statistics on a per-port basis. RMON1 (RFC1757) accounts for bridged traffic in host and matrix statistics, but **not** routed traffic. RMON2 (RFC2021) accounts for routed traffic in host, matrix, and protocol distribution statistics, but **not** bridged traffic. This is a hardware limitation.

Note: **Do not** run NetFlow and RMON simultaneously.

Command Summary

Table 46 lists the **rmon** commands. The sections following the table describe the command syntax.

Table 46. rmon commands

rmon address-map index <*index-number*> {**port** <*port*> [**owner** <*string*>] [**status enable**|**disable**]} |**max-number** <*number*>
rmon al-matrix-top-n index <*index-number*> **matrix-index** <*number*> **ratebase terminal-packets**|**terminal-octets**|**all-packets**|**all-octets duration** <*number*> **size** <*number*> [**owner** <*string*>] [**status enable**|**disable**]
rmon alarm index <*index-number*> **variable** <*string*> [**interval** <*seconds*>] [**falling-event-index** <*num*>] [**falling-threshold** <*num*>] [**owner** <*string*>] [**rising-event-index** <*num*>] [**rising-threshold** <*num*>] [**startup rising**|**falling**|**both**] [**status enable**|**disable**] [**type absolute-value**|**delta-value**]
rmon apply cli-filters <*filter id*>
rmon capture index <*index-number*> **channel-index** <*number*> [**full-action lock**|**wrap**] [**slice-size** <*number*>] [**download-slice-size** <*number*>] [**download-offset** <*number*>] [**max-octets** <*number*>] [**owner** <*string*>] [**status enable**|**disable**]

Table 46. rmon commands

rmon channel index <*index-number*> **port** <*port*> [**accept-type matched**|**failed**] [**data-control on**|**off**] [**turn-on-event-index** <*number*>] [**turn-off-event-index** <*number*>] [**event-index** <*number*>] [**channel-status ready**|**always-ready**] [**description** <*string*>] [**owner** <*string*>] [**status enable**|**disable**]
rmon clear cli-filter
rmon enable
rmon etherstats index <*index-number*> **port** <*port*> [**owner** <*string*>] [**status enable**|**disable**]
rmon event index <*index-number*> **type none**|**log**|**trap**|**both** [**community** <*string*>] [**description** <*string*>] [**owner** <*string*>] [**status enable**|**disable**]
rmon filter index <*index-number*> **channel-index** <*number*> [**data-offset** <*number*>] [**data** <*string*>] [**data-mask** <*string*>] [**data-not-mask** <*string*>] [**pkt-status** <*number*>] [**status-mask** <*number*>] [**status-not-mask** <*number*>] [**owner** <*string*>] [**status enable**|**disable**]
rmon history index <*index-number*> **port** <*port*> [**interval** <*seconds*>] [**owner** <*string*>] [**samples** <*num*>] [**status enable**|**disable**]
rmon hl-host index <*index-number*> **port** <*port*> **nl-max-entries** <*number*> **al-max-entries** <*number*> [**owner** <*string*>] [**status enable**|**disable**]
rmon hl-matrix index <*index-number*> **port** <*port*> **nl-max-entries** <*number*> **al-max-entries** <*number*> [**owner** <*string*>] [**status enable**|**disable**]
rmon host index <*index-number*> **port** <*port*> [**owner** <*string*>] [**status enable**|**disable**]
rmon host-top-n index <*index-number*> **host-index** <*number*> [**base** <*statistics*>] [**duration** <*time*>] [**size** <*size*>] [**owner** <*string*>] [**status enable**|**disable**]
rmon matrix index <*index-number*> [**port** <*port*>] [**owner** <*string*>] [**status enable**|**disable**]
rmon nl-matrix-top-n index <*index-number*> **matrix-index** <*number*> **ratebase terminal-packets**| **terminal-octets**| **all-packets**| **all-octets duration** <*number*> **size** <*number*> [**owner** <*string*>] [**status enable**| **disable**]
rmon protocol-distribution index <*index-number*> **port** <*port*> [**owner** <*string*>] [**status enable**|**disable**]
rmon set lite| **standard**| **professional default-tables yes**| **no**
rmon set cli-filter <*filter-id*> <*parameter*>
rmon set memory <*number*>
rmon set ports <*port list*>| **all-ports**
rmon set protocol-directory <*protocol*>|**all-protocols [address-map on**| **off**| **na] [host on**| **off**| **na] [matrix on**| **off**| **na]**
rmon show address-map-control <*port-list* >| **all-ports**

Table 46. rmon commands

rmon show address-map-logs <*port-list* >**|all-ports**
rmon show al-host <*port-list*> **|all-ports [summary]**
rmon show al-matrix <*port-list*> **|all-ports [order-by srcdst |dstsrc] [summary]**
rmon show al-matrix-top-n
rmon show alarms
rmon show channels
rmon show cli-filters
rmon show etherstats <*port-list*> **|all-ports**
rmon show events
rmon show filters
rmon show history <*port-list*> **|all-ports**
rmon show host-top-n
rmon show hosts <*port-list*> **|all-ports [summary]**
rmon show matrix <*port-list*> **|all-ports [summary] [order-by srcdst| dstsrc]**
rmon show nl-host <*port-list*> **|all-ports [summary]**
rmon show nl-matrix <*port-list*> **|all-ports [order-by srcdst |dstsrc] [summary]**
rmon show nl-matrix-top-n rmon show
rmon show packet-capture control-table [captured-packets <*control-index*>**]**
rmon show probe-config [basic] [net-config] [trap-dest]
rmon show protocol-directory <*protocol*> **|all-protocols**
rmon show protocol-distribution <*port-list*> **|all-ports**
rmon show status
rmon <*string*> **show user-history [all-indexes]**
rmon user-history-apply <*groupname*> **to** <*user-history-index*> [**status enable |disable**]
rmon user-history-control index <*index-number*> **objects** <*number*> **samples** <*number*> **interval** <*number*> [**owner** <*string*>] [**status enable|disable**]
rmon user-history-objects <*groupname*> **variable** <*oid*> **type absolute-value |delta-value** [**status enable|disable**]

rmon address-map

Purpose

Configures the RMON 2 Address Map group.

Format

rmon address-map index <*index-number*> {**port** <*port*> [**owner** <*string*>] [**status enable|disable**]} |**max-number** <*number*>

Mode

Configure

Description

The Address Map group maps MAC addresses to network address bindings that are discovered by the X-Pedition on a per-port basis. The **rmon address-map** command sets various parameters of the RMON 2 Address Map table. If the default tables are turned on for the Professional group, an entry in the Address Map control table is created for each available port. Use the **rmon show address-map** command to display the address map.

Parameters

<*index-number*>
Is a number between 1 and 65535 that uniquely identifies a row in the Address Map table.

<*port*>
Specifies the port from which to collect data.

owner <*string*>
Specifies the owner of the event; for example, an IP address, machine name or person's name.

status enable|disable
Enables or disables this matrix. The default is enable.

max-number <*number*>
The maximum number of entries (1 to 2147483647) to allow in address-map tables. This is helpful for controlling memory used by the RMON task.

Restrictions

None.

Example

To create an entry in the Address Map table for port et.1.3:

```
xp(config)# rmon address-map index 20 port et.1.3
```

rmon al-matrix-top-n

Purpose

Gathers the top *n* Application Layer Matrix entries.

Format

rmon al-matrix-top-n index <*index-number*> **matrix-index** <*number*> **ratebase terminal-packets|terminal-octets|all-packets|all-octets duration** <*number*> **size** <*number*> [**owner** <*string*>] [**status enable|disable**]

Mode

Configure

Description

The **rmon al-matrix-top-n** command gathers the top *n* Application Layer Matrix entries sorted by a specified statistic. To do this, you must first configure the Application Layer/Network Layer Matrix table using the **rmon hl-matrix** command.

Use the **rmon show al-matrix-top-n** command to display the top *n* Application Layer Matrix entries.

Parameters

<*index-number*>
Is a number between 1 and 65535 that uniquely identifies a row in the application layer matrix table.

matrix-index <*number*>
Specifies the index into the hl-matrix table. The default is 0.

ratebase terminal-packets|terminal-octets|all-packets|all-octets
Specifies the sorting method:

terminal-packets	Sort by terminal packets.
terminal-octets	Sort by terminal octets.
all-packets	Sort by all packets.
all-octets	Sort by all octets.

duration <*number*>
Specifies the duration, in seconds, between reports. If the duration is 0 (the default), this implies that no reports have been requested for this entry. The default is 0.

size <*number*>
Specifies the maximum number of matrix entries to include in the report. The default is 150.

owner <*string*>
Specifies the owner of the event; for example, an IP address, machine name or person's name.

status enable|disable
Enables or disables this matrix. The default is enable.

Restrictions

None.

Example

To monitor the top *n* entries in the Application Layer Matrix, you should first configure the Application Layer/Network Layer Matrix table using the **rmon hl-matrix** command. Then, to gather the top 100 Application Layer Matrix entries sorted by all packets, use the following command:

```
xp(config)# rmon al-matrix-top-n index 25 matrix-index 50 ratebase all-packets duration 60 size 100
```

rmon alarm

Purpose

Configures the RMON 1 Alarm group.

Format

rmon alarm index <*index-number*> **variable** <*string*> [**interval** <*seconds*>] [**falling-event-index** <*num*>] [**falling-threshold** <*num*>] [**owner** <*string*>] [**rising-event-index** <*num*>] [**rising-threshold** <*num*>] [**startup rising|falling|both**] [**status enable|disable**] [**type absolute-value|delta-value**]

Mode

Configure

Description

The Alarm group takes periodic statistical samples and compares them with previously-configured thresholds. If a monitored variable crosses a threshold, an alarm is generated. The **rmon alarm** command sets various parameters of the RMON 1 Alarm control table.

Use the **rmon show alarm** command to display the alarm data.

Parameters

index <*index-number*>
Is a number that uniquely identifies an entry in the alarm table. The value must be between 1 and 65535, inclusive.

interval <*seconds*>
Specifies the sampling interval in seconds when statistical samples of variables are collected and compared to the rising and falling thresholds. The value must be between 1 and 2147483647, inclusive.

falling-event-index <*num*>
Is the action to be taken as defined by the row with this index in the event table when a falling threshold is crossed. The value must be between 1 and 65535, inclusive.

falling-threshold <*num*>
Specifies that the sample's value must be less than or equal to the threshold to trigger an alarm. When the sample's value is less than or equal to this threshold, and the value at the last sampling interval was greater than this threshold, a single event is generated. The value must be between 1 and 2147483647, inclusive.

owner <*string*>
Specifies the owner of the alarm resource; for example, an IP address, machine name or person's name.

rising-event-index <*num*>
Is the action to be taken as defined by the row with this index in the event table when a rising threshold is crossed. The value must be between 1 and 65535, inclusive.

rising-threshold <*num*>
Specifies that the sample's value must be greater than or equal to the threshold to trigger an alarm. When the sample's value is greater than or equal to this threshold, and the value at the last sampling interval was less than this threshold, a single event is generated. The value must be between 1 and 2147483647, inclusive.

startup <*keyword*>
Specifies the condition for which the alarm is to be generated. The condition can be one of the following:

rising	Causes an alarm to be generated if the sampled variable is greater than or equal to the rising threshold.
falling	Causes an alarm to be generated if the sampled variable is less than or equal to the falling threshold.
both	Causes an alarm to be generated if the sampled variable is greater than or equal to the rising threshold or less than or equal to the falling threshold.

status enable|disable
Enables or disables this alarm.

type <*keyword*>
Specifies the method of sampling the selected variable and calculating the value to be compared against the thresholds. The sampling method can be one of the following:

absolute-value	Monitor the absolute value over the sample interval of the variable against the threshold value.
delta-value	Monitor the change in value over the sample interval of the variable against the threshold value.

variable <*string*>
Specifies the object identifier of the variable to be sampled. Only variables that resolve to an ASN.1 primitive type of INTEGER may be sampled.

Restrictions

None.

Examples

To cause an alarm event if the variable defined in alarm 10 crosses the rising threshold:

```
xp(config)# rmon alarm index 10 startup rising interval 30 variable 1.3.6.1.2.1.5.14.0 rising-threshold
     40 rising-event-index 1
```

To monitor the absolute value of the variable against a threshold value:

```
xp(config)# rmon alarm index 10 type absolute-value startup rising interval 30 variable
     1.3.6.1.2.1.5.14.0 rising-threshold 40 rising-event-index 1
```

To specify Mike as the owner of alarm 10:

```
xp(config)# rmon alarm index 10 owner Mike type absolute-value startup rising interval 30 variable
     1.3.6.1.2.1.5.14.0 rising-threshold 40 rising-event-index 1
```

To specify a 5-second interval on alarm 10:

```
xp(config)# rmon alarm index 10 interval 5 type absolute-value startup rising interval 30 variable
     1.3.6.1.2.1.5.14.0 rising-threshold 40 rising-event-index 1
```

To specify the rising threshold at 10 on alarm 10:

```
xp(config)# rmon alarm index 10 rising-threshold 10 type delta-value startup rising interval 30
     variable 1.3.6.1.2.1.5.14.0 rising-event-index 1
```

rmon apply cli-filters

Purpose

Apply a specific CLI RMON filter.

Format

rmon apply cli-filters <*filter id*>

Mode

Enable

Description

The **rmon apply cli-filters** command applies a specific CLI RMON filter to the current Telnet or Console session. This enables different users to select the different CLI filters which you should define using the **rmon set cli-filter** command.

Use the **rmon show cli-filters** command to see the RMON CLI filters that have been defined on the X-Pedition. Use the **rmon clear cli-filter** command to clear the applied filter.

Parameter

<*filter id*>	Is a number between 1 and 65535 that identifies the filter ID to apply.

Restrictions

None.

Example

To apply filter ID 2:

```
xp> rmon apply cli-filters 2
```

To see a list of CLI RMON filters:

```
xp> rmon show cli-filters
RMON CLI Filters
Id   Filter
--   ------
 1   (inpkts >= 0)
 2   (inpkts >= 0 and outoctets >= 0)
 3   srcmac 222222222222 and (outoctets >= 0)
You have selected a filter: (inpkts >= 0)
```

rmon capture

Purpose

Configures the RMON 1 Packet Capture group.

Format

rmon capture index <*index-number*> **channel-index** <*number*> [**full-action lock|wrap**] [**slice-size** <*number*>] [**download-slice-size** <*number*>] [**download-offset** <*number*>] [**max-octets** <*number*>] [**owner** <*string*>] [**status enable|disable**]

Mode

Configure

Description

The Packet Capture group allows packets to be captured after they have flowed through a channel. The **rmon capture** command sets various parameters of the RMON 1 Packet Capture table.

Use the **rmon show packet-capture** command to display the Packet Capture table.

Note: **Rmon capture** cannot capture bad Jumbo Frames, Runt Packets, or CRC packets. This is due to a Hardware limitation.

Parameters

index <*index-number*>
Is a number between 1 and 65535 that uniquely identifies a row in the Packet Capture table.

channel-index <*number*>
Is a number between 1 and 65535 that identifies the channel that is the source of packets. The default is 0.

full-action lock|wrap
Specifies the action of the buffer when it reaches the full status:

lock Stop capturing packets when the buffer reaches the full status.

wrap Wrap around when the buffer reaches the full status.

slice-size <*number*>
Is a number between 0 and 2147483647 that is the maximum number of octets that will be saved in this capture buffer. The default is 100.

download-slice-size <*number*>
Is a number between 0 and 2147483647 that is the maximum number of octets that will be returned in an SNMP retrieval. The default is 100.

download-offset <*number*>
The offset of the first octet number (between 0 and 2147483647) of each packet that will be returned in an SNMP retrieval. The default is 0.

max-octets <*number*>
The maximum number of octets (between 0 and 2147483647) to use to hold captured packets and their control information. The default is 1.

Note: The **max-octets** default value allows packet capture to continue indefinitely until all available RMON memory resources are exhausted. If these resources exhaust, RMON features will be disabled.

owner <*string*>
Specifies the owner of the event; for example, an IP address, machine name or person's name.

status enable|disable
Enables or disables this channel. The default is enable.

Restrictions

Packet capture using RMON uses considerable CPU cycles. For best results, enable packet capture when the CPU is not busy running other important tasks and CPU utilization is low.

Example

To create an entry in the Packet Capture table:

```
xp(config)# rmon capture index 20 channel-index 1 full-action wrap
```

rmon channel

Purpose

Configures the RMON 1 Filter Channel group.

Format

rmon channel index <*index-number*> **port** <*port*> [**accept-type matched|failed**] [**data-control on|off**] [**turn-on-event-index** <*number*>] [**turn-off-event-index** <*number*>] [**event-index** <*number*>] [**channel-status ready|always-ready**] [**description** <*string*>] [**owner** <*string*>] [**status enable|disable**]

Mode

Configure

Description

The Filter Channel group must be configured in order to configure the Filter group. The **rmon channel** command sets various parameters of the RMON 1 Filter Channel table. After a channel row has been created, a filter must be defined with the **rmon filter** command.

Use the **rmon show channels** command to display all the channels configured on the X-Pedition.

Parameters

index <*index-number*>
Is a number between 1 and 65535 that uniquely identifies a row in the Filter Channel table.

port <*port*>
Identifies the port from which data is collected.

accept-type matched|failed
Specifies the action of the filters associated with this channel:

matched	Packets will be accepted if they are accepted by both the packet data and packet status matches of an associated filter.
failed	Packets will be accepted only if they fail either the packet data match or the packet status match of each of the associated filters.

data-control on|off
Specifies the flow control of the data:

on	Implies data, status, and events flow through this channel.
off	Implies data, status, and events will not flow through this channel.

turn-on-event-index <*number*>
Is a number between 0 and 65535 that identifies the event configured to turn the associated data control from off to on.

turn-off-event-index <*number*>
Is a number between 0 and 65535 that identifies the event configured to turn the associated data control from on to off.

event-index <*number*>
Is a number between 0 and 65535 that identifies the event configured to be generated when the associated data control is on and a packet is matched.

channel-status ready|always-ready
Specifies the status:

ready	A single event is generated.
always-ready	Allows events to be generated at will.

description <*string*>
Describes this channel in a maximum of 127 bytes.

owner <*string*>
Specifies the owner of packet capture; for example, an IP address, machine name or person's name.

status enable|disable
Enables or disables this channel. The default is enable.

Restrictions

None.

Example

To create an entry in the Filter Channel table:

```
xp(config)# rmon channel index 25 port et.1.3 accept-type matched data-control on turn-on-event-index 30 turn-off-event-index 55 event-index 60 channel-status ready
```

rmon clear cli-filter

Purpose

Clear the currently-selected CLI RMON filter.

Format

rmon clear cli-filter

Mode

Enable

Description

The **rmon clear cli-filter** command clears the CLI RMON filter that was applied with the **rmon apply cli-filters** command.

Parameters

None.

Restrictions

None.

rmon enable

Purpose

Enables RMON.

Format

rmon enable

Mode

Configure

Description

When the X-Pedition is booted, RMON is off by default. The **rmon enable** command turns RMON on. At least one of the Lite, Standard, or Professional RMON groups must be configured first before you can turn on RMON. Use the **rmon set** command to configure the Lite, Standard, or Professional RMON groups.

To disable RMON, the **rmon enable** command must be negated. This frees up all resources associated with RMON, including any memory allocated to RMON.

Parameters

None.

Restrictions

If the SNMP agent is disabled, RMON cannot be enabled. If RMON is enabled and the SNMP agent is disabled, then RMON will be turned off.

rmon etherstats

Purpose

Configures the RMON 1 Ethernet Statistics (Etherstats) group.

Format

rmon etherstats index <*index-number*> **port** <*port*> [**owner** <*string*>] [**status enable|disable**]

Mode

Configure

Description

The Etherstats group contains statistics for X-Pedition ports. The **rmon etherstats** command sets various parameters of the RMON 1 Etherstats control table. If default tables were turned on for the Lite group, a entry is created in the Etherstats control table for each available port.

Use the **rmon show etherstats** command to display the Etherstats data.

Parameters

index <*index-number*>
Is a number between 1 and 65535 that uniquely identifies a row in the Etherstats control table.

port <*port*>
Specifies the physical port from which to collect data.

owner <*string*>
Specifies the owner of the event; for example, an IP address, machine name or person's name.

status enable|disable
Enables or disables this Etherstat. The default is enable.

Restrictions

The RMON agent reports only traffic *received* on a port.

Example

To create an entry in the Etherstats control table:

```
xp(config)# rmon etherstats index 10 port et.1.3
```

rmon event

Purpose

Configures the RMON 1 Event group.

Format

rmon event index <*index-number*> **type none|log|trap|both** [**community** <*string*>] [**description** <*string*>] [**owner** <*string*>] [**status enable|disable**]

Mode

Configure

Description

The Event group controls the generation and notification of events. The **rmon event** command sets various parameters of the RMON 1 Event control table.

Use the **rmon show event** command to display the event data.

Parameters

index <*index-number*>
Is a number between 1 and 65535 that uniquely identifies an entry in the Event table.

community <*string*>
Specifies the SNMP community string to be sent with the trap. If an SNMP trap is to be sent, it will go to the SNMP community specified in this string.

description <*string*>
Specifies a comment describing this event.

owner <*string*>
Specifies the owner of the event; for example, an IP address, machine name or person's name.

status enable|disable
Enables or disables this event. The default is enable.

type none|log|trap|both
Specifies what action to be taken when the event occurs. The action can be one of the following:

none Causes no notification to be sent for the event.

log Causes an entry for the event to be made in the log table for each event.

trap Causes an SNMP trap to be sent to one or more management stations for the event.

both Causes both an entry to be made in the log table and an SNMP trap to be sent to one or more management stations.

Restrictions

None.

Examples

To set the event community string to public:

```
xp(config)# rmon event index 10 community public
```

To add the description “num-pkts” to event 10:

```
xp(config)# rmon event index 10 description num-pkts
```

To specify Ed as the owner of event 10:

```
xp(config)# rmon event index 10 owner Ed
```

To send an SNMP trap when event 10 is triggered:

```
xp(config)# rmon event index 10 type trap
```

rmon filter

Purpose

Configures the RMON 1 Filter group.

Format

rmon filter index <*index-number*> **channel-index** <*number*> [**data-offset** <*number*>] [**data** <*string*>] [**data-mask** <*string*>] [**data-not-mask** <*string*>] [**pkt-status** <*number*>] [**status-mask** <*number*>] [**status-not-mask** <*number*>] [**owner** <*string*>] [**status enable**|**disable**]

Mode

Configure

Description

The Filter group allows packets to be matched on certain criteria. The **rmon filter** command sets various parameters of the RMON 1 Filter table. To configure the Filter group, the Filter Channel group must first be configured with the **rmon channel** command.

Use the **rmon show filters** command to display the filters defined on the X-Pedition.

Parameters

index <*index-number*>
Is a number between 1 and 65535 that uniquely identifies a row in the Filter table.

channel-index <*number*>
Is a number between 1 and 65535 that identifies the channel of which this filter is a part.

data-offset <*number*>
Is a number between 0 and 2147483647 that is the offset from the beginning of each packet where a match of packet data will be attempted.

data <*string*>
Is a string of up to 512 characters that is the data that is to be matched with the input packet.

data-mask <*string*>
Is a string of up to 512 characters that is the mask that is applied to the match process.

data-not-mask <*string*>
Is a string of up to 512 characters that is the inversion mask that is applied to the match process.

pkt-status <*number*>
Is a number between 0 and 2147483647 that is the status that is to be matched with the input packet.

status-mask <*number*>
Is a number between 0 and 2147483647 that is the mask that is applied to the status match process.

status-not-mask <*number*>
Is a number between 0 and 2147483647 that is the inversion mask that is applied to the status match process.

owner <*string*>
Specifies the owner of the event; for example, an IP address, machine name or person's name.

status enable|disable
Enables or disables this channel. The default is enable.

Restrictions

None.

Example

To create an entry in the Filter table:

```
xp(config)# rmon filter index 25 channel-index 35 data kgreen
```

rmon history

Purpose

Configures the RMON 1 History group.

Format

rmon history index <*index-number*> **port** <*port*> [**interval** <*seconds*>] [**owner** <*string*>] [**samples** <*num*>] [**status enable|disable**]

Mode

Configure

Description

The RMON History group periodically records samples of variables and stores them for later retrieval. You use the r**mon history** command to specify the X-Pedition port to collect data from, the number of samples, the sampling interval, and the owner. If default tables were turned on for the Lite group, an entry is created in the History control table for each available port.

Use the **rmon show history** command to display the history data.

Parameters

index <*index-number*>
Is a number between 1 and 65535 that uniquely identifies an entry in the History table.

interval <*seconds*>
Specifies the sampling interval in seconds. This value must be between 1 and 3600, inclusive. The default value is 1800.

owner <*string*>
Specifies the owner of the history resource; for example, an IP address, machine name or person's name.

port <*port*>
Specifies the port from which to collect data.

samples <*num*>
Specifies the number of samples to be collected before wrapping counters. This value must be between 1 and 65535, inclusive. The default value is 50.

status enable|disable
Enables or disables this history control row.

Restrictions

None.

Example

To specify that port et.3.1 collect 60 samples at an interval of 30 seconds:

```
xp(config)# rmon history index 10 port et.3.1 samples 60 interval 30
```

rmon hl-host

Purpose

Configures the RMON 2 Application Layer and Network Layer Host groups.

Format

rmon hl-host index <*index-number*> **port** <*port*> **nl-max-entries** <*number*> **al-max-entries** <*number*> [**owner** <*string*>] [**status enable**|**disable**]

Mode

Configure

Description

The **rmon hl-host** command sets various parameters of the RMON 2 Application Layer and Network Layer Host groups. The Application Layer Host group monitors traffic from the network layer up to the application layer for any protocol communication defined in the protocol directory. The Network Layer Host group monitors traffic at the network layer for any protocol defined in the protocol directory.

Configuration of the Application Layer/Network Layer Host table involves configuring only one control row in the Application Layer Host control table. This table, when configured, captures both application layer and network layer host data. If the default tables were turned on for the Professional group, an entry is created in the Application Layer Host control table for each available port.

Use the **rmon show al-host** command to display the Application Layer Host table. Use the **rmon show nl-host** command to display the Network Layer Host table.

Parameters

index <*index-number*>
Is a number between 1 and 65535 that uniquely identifies a row in the application layer host control table.

<*port*>
Specifies the port from which to collect data.

nl-max-entries
Specifies the maximum number of network layer entries. The default is 1.

al-max-entries
Specifies the maximum number of application layer entries. The default is 1.

owner <*string*>
Specifies the owner of the event; for example, an IP address, machine name or person's name.

status enable|disable
Enables or disables this matrix. The default is enable.

Restrictions

None.

Example

To create an entry in the Application Layer Host control table:

```
xp(config)# rmon hl-host index 20 port et.1.3
```

rmon hl-matrix

Purpose

Configures the RMON 2 Application Layer Matrix and Network Layer Matrix groups.

Format

rmon hl-matrix index <*index-number*> **port** <*port*> **nl-max-entries** <*number*> **al-max-entries** <*number*> [**owner** <*string*>] [**status enable**|**disable**]

Mode

Configure

Description

The **rmon hl-matrix** command sets various parameters of the RMON 2 Application Layer Matrix and Network Layer Matrix groups. The Application Layer Matrix group monitors traffic from the network layer up to the application layer for any protocol communication defined in the protocol directory. The Network Layer Matrix group monitors traffic at the network layer for any protocol defined in the protocol directory.

Configuration of the Application Layer/Network Layer Matrix table involves configuring only one control row in the Application Layer Matrix control table. When configured, this table captures both application layer and network layer matrix data. If the default tables were turned on for the Professional group, an entry is created in the Application Layer Matrix control table for each available port.

Use the **rmon show al-matrix** command to display the Application Layer Matrix table. Use the **rmon show nl-matrix** command to display the Network Layer Matrix table.

Parameters

index <*index-number*>
Is a number between 1 and 65535 that uniquely identifies a row in the application layer matrix control table.

port <*port*>
Specifies the port from which to collect data.

nl-max-entries <*number*>
Specifies the maximum number of network layer entries. The default is 1.

al-max-entries <*number*>
Specifies the maximum number of application layer entries. The default is 1.

owner <*string*>
Specifies the owner of the event; for example, an IP address, machine name or person's name.

status enable|disable
Enables or disables this matrix. The default is enable.

Restrictions

None.

Example

To create an entry in the Application Layer Matrix control table:

```
xp(config)# rmon hl-matrix index 20 port et.1.3
```

rmon host

Purpose

Configures the RMON 1 Host group.

Format

rmon host index <*index-number*> **port** <*port*> [**owner** <*string*>] [**status enable|disable**]

Mode

Configure

Description

The RMON 1 Host group captures L2 information from hosts coming in on a particular port. The **rmon host** command sets various parameters of the Host group. If default tables were turned on for the standard group, an entry is created in the Host control table for each available port.

Use the **rmon show hosts** command to display the host data and logs.

Parameters

index <*index-number*>
Is a number between 1 and 65535 that uniquely identifies a row in the Host table.

port <*port*>
Specifies the physical port from which to collect data.

owner <*string*>
Specifies the owner of the event; for example, an IP address, machine name or person's name.

status enable|disable
Enables or disables this host. The default is enable.

Restrictions

None.

Example

To create an entry in the Host control table:

```
xp(config)# rmon hosts index 20 port et.1.3
```

rmon host-top-n

Purpose

Configures the RMON 1 HostTopN group.

Format

rmon host-top-n index <*index-number*> **host-index** <*number*> [**base** <*statistics*>] [**duration** <*time*>] [**size** <*size*>] [**owner** <*string*>] [**status enable**|**disable**]

Mode

Configure

Description

The HostTopN group displays the top n number of hosts, sorted by a specified statistic. The **rmon host-top-n** command sets various parameters of the RMON 1 HostTopN control table. The HostTopN group depends upon the Host group and the host-index specified in the HostTopN control table must correspond to a pre-defined host index in the Host control table.

Use the r**mon show host-top-n** command to display the control table row.

Note that Host Top N report runs once. To run the reports again via the CLI, the control row must be disabled and then enabled. If the report has already been run, the Time Remaining field is set to zero. Otherwise, the Time Remaining field will be decremented until the report is run.

Parameters

index <*index-number*>
Is a number between 1 and 65535 that uniquely identifies a row in the Host Top N table.

<*number*>
Is a number between 1 and 65535 that is the index into the host table identified by hostIndex.

<*statistics*>
Specifies the type of statistic from which to collect data. Specify one of the following keywords:

in-packets Gather top statistics according to In-Packets.

out-packets
Gather top statistics according to Out-Packets.

in-octets Gather top statistics according to In-Octets.

out-octets Gather top statistics according to Out-Octets.

out-errors Gather top statistics according to Out-Errors.

Note: This option is no longer valid—hardware restrictions prevent the host-top-n table from seeing out-errors.

out-broadcastPkts
Gather top statistics according to Out-BroadcastPkts.

out-multicastPkts
Gather top statistics according to Out-MulticastPkts.

<*time*>
Is a number between 1 and 2147483647 that is the duration, in seconds, between reports. The default is 0.

<size>
Is a number between 1 and 2147483647 that is the maximum number of hosts to include in the table. The default is 10.

owner <*string*>
Specifies the owner of the event; for example, an IP address, machine name or person's name.

status enable|disable
Enables or disables this hostTopN. The default is enable.

Restrictions

None.

Example

To create an entry in the HostTopN control table:

```
xp(config)# rmon host-top-n index 25 host-index 55 base in-packets duration 60 size 24
```

rmon matrix

Purpose

Configures the RMON 1 Matrix group.

Format

rmon matrix index <*index-number*> [**port** <*port*>] [**owner** <*string*>] [**status enable|disable**]

Mode

Configure

Description

The Matrix group captures L2 traffic on a particular port between two hosts (a source MAC and destination MAC address). The **rmon matrix** command sets various parameters of the RMON 1 Matrix control table. If default tables were turned on for the Standard group, an entry is created in the Matrix control table for each available port.

Note: By default, ports on the X-Pedition operate in address-bridging mode. The port must be enabled in *flow-bridging* mode in order for layer 2 matrix information to be captured.

Use the **rmon show matrix** command to display the matrix group and logs.

Parameters

index <*index-number*>
Is a number between 1 and 65535 that uniquely identifies a row in the Matrix table.

port <*port*>
Specifies the port from which to collect data.

owner <*string*>
Specifies the owner of the event; for example, an IP address, machine name or person's name.

status enable|disable
Enables or disables this matrix. The default is enable.

Restrictions

None.

Example

To create an entry in the Matrix control table:

```
xp(config)# rmon matrix index 25 port et.1.3
```

rmon nl-matrix-top-n

Purpose

Gathers the top n Network Layer Matrix entries.

Format

rmon nl-matrix-top-n index <*index-number*> **matrix-index** <*number*> **ratebase packets |octets duration** <*number*> **size** <*number*> [**owner** <*string*>] [**status enable|disable**]

Mode

Configure

Description

The **rmon nl-matrix-top-n** command gathers the top n Network Layer Matrix entries.Before you do this, you should first configure the Application Layer/Network Layer Matrix table using the **rmon hl-matrix** command.

Use the **rmon show nl-matrix-top-n** command to display the top n Network Layer Matrix entries.

Parameters

index <*index-number*>
Is a number between 1 and 65535 that uniquely identifies a row in the network layer matrix table.

matrix-index <*number*>
Specifies the index into the hl-matrix table. The default is 0.

ratebase packets|octets
Specifies the sorting method:

packets	Sort by packets.
octets	Sort by octets.

duration <*number*>
Specifies the duration, in seconds, between reports. The default is 0.

size <*number*>
Specifies the maximum number of matrix entries to include in the report. The default is 150.

owner <*string*>
Specifies the owner of the event; for example, an IP address, machine name or person’s name.

status enable|disable
Enables or disables this matrix. The default is enable.

Restrictions

None.

Example

To gather the top n Network Layer Matrix entries:

```
xp(config)# rmon nl-matrix-top-n index 2 matrix-index 25 ratebase all-packets duration 60 size 100
```

rmon protocol-distribution

Purpose

Configures the RMON 2 Protocol Distribution group.

Format

rmon protocol-distribution index <*index-number*> **port** <*port*> [**owner** <*string*>] [**status enable**|**disable**]

Mode

Configure

Description

The Protocol Distribution group displays the packets and octets on a protocol and port basis. The **rmon protocol-distribution** command sets various parameters of the RMON 2 Protocol Distribution control table. If default tables were turned on for the Professional group, an entry is created in the Protocol Distribution control table for each available port.

Use the **rmon show protocol-distribution** command to display the protocol distribution.

Parameters

index <*index-number*>
Is a number between 1 and 65535 that uniquely identifies a row in the Protocol Distribution table.

port <*port*>
Specifies the port from which to collect data.

owner <*string*>
Specifies the owner of the event; for example, an IP address, machine name or person's name.

status enable|disable
Enables or disables this matrix. The default is enable.

Restrictions

None.

Example

To create an entry in the Protocol Distribution control table:

```
xp(config)# rmon protocol-distribution index 25 port et.1.3
```

rmon set

Purpose

Configures the Lite, Standard, or Professional RMON groups.

Format

rmon set lite|standard|professional default-tables yes|no

Mode

Configure

Description

You can enable various levels of support (Lite, Standard, or Professional) for RMON groups on a specified set of ports.

Lite adds support for the following RMON 1 groups:

- Ethernet statistics (Etherstats)
- History
- Alarm
- Event

Standard adds support for the following RMON 1 groups:

- Host
- HostTopN
- Matrix
- Filter
- Packet Capture

Note: Packet capture using RMON uses considerable CPU cycles. For best results, enable packet capture when the CPU is not busy running other important tasks and CPU utilization is low.

Professional adds support for the following RMON 2 groups:

- Protocol Directory
- Protocol Distribution
- Address Map

- Network Layer Host
- Network Layer Matrix
- Application Layer Host
- Application Layer Matrix
- User History
- Probe Configuration

A group can consist of a control table and a data table. A control table specifies the statistics to be collected. Each row in the control table specifies the entities for which data is collected, for example, physical ports. The data tables contain the statistics that are collected based on the control table information.

Parameters

lite|standard|professional
Specifies the Lite, Standard, or Professional RMON groups.

default-tables yes
Creates control tables for the following Lite, Standard, or Professional RMON groups:

Lite groups:	Etherstats History
Standard groups:	Host Matrix
Professional groups:	Protocol Distribution Address Map Application Layer/Network Layer Host Application Layer/Network Layer Matrix

A row in each control table is created for each port on the X-Pedition, with the default owner "monitor."

default-tables no
Removes all control table rows with the owner "monitor". If you wish to save a particular control table row, you must change the owner to a value other than "monitor".

Restrictions

None.

Example

To configure the RMON Lite groups and create default control tables:

```
xp(config)# rmon set lite default-tables yes
```

rmon set cli-filter

Purpose

Defines filters that can be applied to certain RMON groups during a CLI session.

Format

rmon set cli-filter <*filter-id*> <*parameter*>

Mode

Configure

Description

You can define filters that CLI users can apply to certain RMON groups. The filters you define are visible to all users that have a Telnet or Console session on the X-Pedition. Each user has the choice of whether or not to apply a particular filter using the **rmon apply cli-filters command**.

RMON CLI filters only affect the output of the following RMON groups:

- Host
- Matrix
- Network Layer Host
- Application Layer Host
- Network Layer Matrix
- Application Layer Matrix
- Protocol Distribution

The **rmon show cli-filters** command displays the RMON CLI filters that have been defined on the X-Pedition.

Parameters

<*filter-id*>
Is a number between 1 and 65535 that uniquely identifies a CLI filter.

<*parameter*>
Specifies the parameter on which the filter is set:

src-mac Source MAC Address

dst-mac Destination MAC Address

inpkts In Packets

inoctets In Octets

outpkts out packets

outoctets out Octets

multicast Multicast packets

broadcast Broadcast packets

errors Errors

The following operands can also be used:

and AND

or Or

= Equal to

< Less than

<= Less than or equal to

> Greater than

>= Greater than or equal to

!= Not equal to

(Left bracket

) Right Bracket

src-mac and **dst-mac** can be specified once and the other parameters can be specified multiple times.

Restrictions

None.

Example

To configure an RMON CLI filter on a source MAC address of 123456:123456 and on input packets greater than 1000 and error packets greater than 10 or out packets less than 10000, use the following command:

```
xp(config)# rmon set cli-filter 3 src-mac 123456:123456 and ((inpkts > 1000 and errors > 10)
or (outpkts < 10000))
```

rmon set memory

Purpose

Increases the amount of memory allocated to RMON.

Format

rmon set memory <*number*>

Mode

Enable

Description

RMON allocates memory depending on the number of ports enabled for RMON, the groups that have been configured (Lite, Standard, or Professional) and whether or not default tables have been turned on or off. You can dynamically allocate additional memory to RMON, if needed.

Later, if this additional memory is no longer required, you can reduce the allocation; this change will not take effect until RMON is restarted. This is because memory cannot be freed while RMON is still using it. If the amount of memory specified is less than what RMON has currently allocated, a warning message is displayed and the action is ignored.

Use the **rmon show status** command to display the amount of memory currently allocated to RMON.

Parameters

<*number*>
Specifies the total amount of memory, in Mbytes, to be allocated to RMON. The value can be between 2 and 96.

Note: The number specified is the total number of Mbytes of memory to be allocated; it is not an increment of memory.

Restrictions

None.

Example

To show the amount of memory allocated to RMON:

```
xp# rmon show status
```

To increase the amount of memory allocated to RMON:

```
xp# rmon set memory 32
```

rmon set ports

Purpose

Enables RMON on one or more ports.

Format

rmon set ports <*port list*>|**allports**

Mode

Configure

Description

Since RMON uses many system resources, RMON can be enabled on a set of ports. Ports can be dynamically added and removed from the port list. For example, if default tables are turned on for the Lite group and port et.2.1 is then added to the port list, an entry for port et.2.1 is automatically created in the Etherstats and History control tables.

Parameters

<*port list*>
Specifies the port(s) on which RMON is enabled. Specify **allports** to enable RMON for all ports on the X-Pedition.

Restrictions

None.

Example

To enable RMON on all ports on the X-Pedition:

```
xp(config)# rmon set ports allports
```

rmon set protocol-directory

Purpose

Specifies the protocol encapsulations that are managed with the Protocol Directory group.

Format

rmon set protocol-directory <*protocol*>|**all-protocols [address-map on|off|na] [host on|off|na] [matrix on|off|na]**

Mode

Configure

Description

The **rmon set protocol-directory** command defines the protocols that are managed with RMON on the X-Pedition.

Parameters

<*protocol*>
Specifies the protocol encapsulations that are managed with the Protocol Directory group on the X-Pedition. (See Appendix A for a list of protocols supported on the X-Pedition.) Specify **all-protocols** to manage all protocols that are supported on the X-Pedition.

address-map on|off|na
Configures support for the Address Map group for the specified protocol(s).

host on|off|na
Configures support for the Host group for the specified protocol(s).

matrix on|off|na
Configures support for the Matrix group for the specified protocol(s).

Restrictions

The Protocol Directory group is part of the RMON Professional group. To use the **rmon set protocol-directory** command you must enable the RMON Professional group with the **rmon set professional** command.

Example

To configure a protocol encapsulation for the Protocol Directory group:

```
xp(config)# rmon set protocol-directory all-protocols address-map on host on matrix on
```

rmon show address-map-logs

Purpose

Displays MAC address to network address bindings for each protocol.

Format

rmon show address-map-logs <*port-list* >|**all-ports**

Mode

Enable

Description

The **rmon show address-map-logs** command displays entries in the RMON 2 Address Map log table. Entries in this table are created automatically when default tables are turned on for the Professional group. You can show address bindings for specific ports or for all ports.

Parameters

<*port-list* >|**all-ports**
The port(s) for which you want to display MAC-network address information. Use the keyword **all-ports** to show information for all ports.

Restrictions

This command is only available if you have configured the Professional group and Address Map control table entries exist for the specified port.

Example

To display the address map log table for all ports:

```
xp# rmon show address-map-logs all-ports
RMON II Address Map Control Table

❶       ❷          ❸           ❹
Port      macAdd       nlAdd          Protocol
----      ------       -----          --------
et.5.1    00001D:CBA3FD 192.100.81.1    ether2.ip-v4
et.5.1    00001D:CBA3FD 192.100.81.1    *ether2.ip-v4
et.5.1    00001D:CBA3FD 10.60.89.88     ether2.ip-v4
et.5.1    00001D:CBA3FD 10.60.89.88     *ether2.ip-v4
et.5.5    00001D:CBA3FD 192.100.81.3    ether2.ip-v4
et.5.5    00001D:CBA3FD 192.100.81.3    *ether2.ip-v4
et.5.5    080020:835CAA 10.60.89.88     ether2.ip-v4
et.5.5    080020:835CAA 10.60.89.88     *ether2.ip-v4
et.5.1    0080C8:C172A6 192.100.81.3    ether2.ip-v4
et.5.1    0080C8:C172A6 192.100.81.3    *ether2.ip-v4
```

Legend:

1. The port on which the MAC address-network address binding was discovered.
2. The MAC address for the binding.
3. The network layer address for the binding.
4. The protocol, as specified in the RMON Protocol Directory for the X-Pedition.

rmon show address-map-control

Purpose

Displays the address map control table.

Format

rmon show address-map-control

Mode

Enable

Description

The **rmon show address-map-control** command displays the collection of network layer addresses to physical addresses to interface mappings.

Note: This is not like the typical RMON controlTable and dataTable in which each entry creates its own datatable. Each entry in this table enables the discovery of addresses on a new interface and the placement of address mappings into the central addressMapTable.

Implementations are encouraged to add an entry per monitored interface upon initialization so that a default collection of address mappings is available.

Parameters

None.

Restrictions

None.

rmon show al-host

Purpose

Shows application layer traffic.

Format

rmon show al-host <*port-list*>|**all-ports** [**summary**]

Mode

Enable

Description

The **rmon show al-host** command shows entries in the RMON 2 Application Layer Host table for one or more ports. Entries in this table are created automatically when default tables are turned on for the Professional group.

If CLI filters have been applied, they will take effect when the Application Layer Host table is displayed. This command shows control rows and their corresponding logs only if there are logs. A control row with no data will not appear in the report.

The Application Layer host group is configured with the **rmon hl-host** command.

Parameters

<*port-list*>|**all-ports**
: The port(s) for which you want to display application layer traffic information. Use the keyword **all-ports** to show traffic information for all the ports.

[**summary**]
: Use the keyword **summary** to display control row summary information only.

Restrictions

This command is only available if you have configured the Professional group and control table entries exist for the specified port.

Example

To show Application Layer Host tables on all ports:

```
xp# rmon show al-host all-ports
RMON II Application Layer Host Table

Index: 500, Port: et.5.1, Inserts: 9, Deletes: 0, Owner: monitor ❶

❷            ❸    ❹       ❺     ❻       ❼
Address         InPkts  InOctets   OutPkts  OutOctets  Protocol
-------         ------  --------   -------  ---------  --------
10.60.89.88       1080    879418         2        164  *ether2.ip-v4
10.60.89.88       1080    879418         2        164  *ether2.ip-v4.tcp
10.60.89.88       1080    879418         2        164  *ether2.ip-v4.tcp.telnet
192.100.81.1         1       100         1        100  *ether2.ip-v4
192.100.81.1         1       100         1        100  *ether2.ip-v4.icmp
192.100.81.3         3       264      1081     879518  *ether2.ip-v4
192.100.81.3         1       100         1        100  *ether2.ip-v4.icmp
192.100.81.3         2       164      1080     879418  *ether2.ip-v4.tcp
192.100.81.3         2       164      1080     879418  *ether2.ip-v4.tcp.telnet

Index: 504, Port: et.5.5, Inserts: 6, Deletes: 0, Owner: monitor
Address         InPkts  InOctets   OutPkts  OutOctets  Protocol
-------         ------  --------   -------  ---------  --------
10.60.89.88          3       246      1141      92563  *ether2.ip-v4
10.60.89.88          3       246      1141      92563  *ether2.ip-v4.tcp
10.60.89.88          3       246      1141      92563  *ether2.ip-v4.tcp.telnet
192.100.81.3      1141     92563         3        246  *ether2.ip-v4
192.100.81.3      1141     92563         3        246  *ether2.ip-v4.tcp
192.100.81.3      1141     92563         3        246  *ether2.ip-v4.tcp.telnet
```

Legend:

1. The control table entry for this port:

   Index: uniquely identifies the entry in the control table.
   Port: port name.
   Inserts: number of Application Layer Host table entries for this port.
   Deletes: number of Application Layer Host table entries deleted for this port.
   Owner: default owner “monitor.”

2. Network address discovered on the port.

3. Number of packets transmitted without errors to the network address for the protocol.

4. Number of octets transmitted without errors to the network address for the protocol.

5. Number of packets transmitted without errors from the network address for the protocol.

6. Number of octets transmitted without errors from the network address for the protocol.

7. The protocol, as specified in the RMON Protocol Directory for the X-Pedition. Note that this shows the destination socket, as well as application/protocol information.

rmon show al-matrix

Purpose

Shows application layer traffic between source and destination addresses.

Format

rmon show al-matrix <*port-list*>|**all-ports [order-by srcdst|dstsrc] [summary]**

Mode

Enable

Description

The **rmon show al-matrix** command shows entries in the RMON 2 Application Layer Matrix table for one or more ports. Entries in this table are created automatically when default tables are turned on for the Professional group.

If CLI filters have been applied, they will take effect when this table is displayed. The control rows and their corresponding logs are displayed only if there are logs. A control row with no data will not appear in the report.

Parameters

<*port-list*>|**all-ports**
The port(s) for which you want to display application layer traffic information. Use the keyword **all-ports** to show traffic information for all the ports.

srcdst
Orders the logs by source address, then destination address (default).

dstsrc
Orders the logs by destination address, then source address.

summary
Displays control row summary information only.

Restrictions

This command is only available if you have configured the Professional group and control table entries exist for the specified port.

Example

To show the Application Layer Matrix table for all ports:.

```
xp# rmon show al-matrix all-ports
RMON II Application Layer Host Table

Index: 500, Port: et.5.1, Inserts: 10, Deletes: 0, Owner: monitor ❶

❷           ❸           ❹       ❺     ❻
SrcAddr       DstAddr          Packets    Octets  Protocol
-------       -------          -------    ------  --------
10.60.89.88    192.100.81.3          2       164  *ether2.ip-v4
10.60.89.88    192.100.81.3          2       164  *ether2.ip-v4.tcp
10.60.89.88    192.100.81.3          2       164  *ether2.ip-v4.tcp.telnet
192.100.81.1   192.100.81.3          1       100  *ether2.ip-v4
192.100.81.1   192.100.81.3          1       100  *ether2.ip-v4.icmp
192.100.81.3   10.60.89.88        1181    972211  *ether2.ip-v4
192.100.81.3   10.60.89.88        1181    972211  *ether2.ip-v4.tcp
192.100.81.3   10.60.89.88        1181    972211  *ether2.ip-v4.tcp.telnet
192.100.81.3   192.100.81.1          1       100  *ether2.ip-v4
192.100.81.3   192.100.81.1          1       100  *ether2.ip-v4.icmp

Index: 504, Port: et.5.5, Inserts: 6, Deletes: 0, Owner: monitor
SrcAddr       DstAddr          Packets    Octets  Protocol
-------       -------          -------    ------  --------
10.60.89.88    192.100.81.3       1242    100744  *ether2.ip-v4
10.60.89.88    192.100.81.3       1242    100744  *ether2.ip-v4.tcp
10.60.89.88    192.100.81.3       1242    100744  *ether2.ip-v4.tcp.telnet
192.100.81.3   10.60.89.88           3       246  *ether2.ip-v4
192.100.81.3   10.60.89.88           3       246  *ether2.ip-v4.tcp
192.100.81.3   10.60.89.88           3       246  *ether2.ip-v4.tcp.telnet
```

Legend:

1. The control table entry for this port:

   Index: uniquely identifies the entry in the control table.
   Port: port name.
   Inserts: number of application layer host table entries for this port.
   Deletes: number of application layer host table entries deleted for this port.
   Owner: default owner “monitor.”

2. Source address.

3. Destination address.

4. Number of link layer packets transmitted from the source to the destination without errors for the protocol.

5. Number of octets transmitted from the source to the destination without errors for the protocol.

6. The protocol, as specified in the RMON Protocol Directory for the X-Pedition.

rmon show al-matrix-top-n

Purpose

Reports the top *n* Application Layer Matrix entries, sorted by a specific metric.

Format

rmon show al-matrix-top-n

Mode

Enable

Description

The **rmon show al-matrix-top-n** command shows entries in the RMON 2 Application Layer Matrix Top N table.

Parameters

None.

Restrictions

This command is only available if you have enabled the Professional RMON group and entries exist in the Application Layer Matrix Top N table.

Example

Consider the following command to gather the top n Application Layer Matrix entries:

```
xp(config)# rmon al-matrix-top-n index 1 matrix-index 500 ratebase all-packets duration 20 size 5
```

To show the top n entries in the Application Layer Matrix table, as specified by the previous command:

```
xp# rmon show al-matrix-top-n
RMON II Al Matrix Table

❶ ❷ ❸ ❹ ❺ ❻ ❼ ❽ ❾
Index M-Index RateBase TimeRem Duration Size StartTime Reports Owner
    1     500 All-Packets      14      20    5 00D 00H 50M 25S       1 Usama

❿ ⓫ ⓬ ⓭ ⓮ ⓯ ⓰
SrcAddr DstAddr PktRate R-PktRate OctetRate R-OctetRate Protocol
------- ------- ------- --------- --------- ---------- ---------
192.100.81.3 10.60.89.88 21 0 19836 0 *ether2.ip-v4.tcp.telnet
192.100.81.3 10.60.89.88 21 0 19836 0 *ether2.ip-v4.tcp
192.100.81.3 10.60.89.88 21 0 19836 0 *ether2.ip-v4
192.100.81.1 192.100.81.3 0 0 0 0 *ether2.ip-v4
192.100.81.3 192.100.81.1 0 0 0 0 *ether2.ip-v4
```

Legend:

1. Index number that identifies this entry in the Application Layer Matrix Top N control table.
2. The Application Layer Matrix table for which the top N report is shown.
3. The parameter on which the entries are sorted.
4. Number of seconds left in the report currently being collected.
5. Number of seconds that this report has collected during the last sampling interval.
6. Maximum number of matrix entries in this report.
7. The time when this report was last started.
8. The number of reports generated by this entry.
9. The entity that configured this entry.
10. Network address of the source host.
11. Network address of the destination host.
12. Number of packets from the source to the destination during the sampling interval.
13. Number of packets from the destination to the source during the sampling interval.
14. Number of octets from the source to the destination during the sampling interval.
15. Number of octets from the destination to the source during the sampling interval.
16. The protocol, as defined in the RMON Protocol Directory group on the X-Pedition.

rmon show alarms

Purpose

Displays configured alarms.

Format

rmon show alarms

Mode

Enable

Description

The **rmon show alarms** command displays the RMON Alarm table.

Parameters

None.

Restrictions

This command is only available if you have configured the Lite group.

Example

To show configured RMON alarms:

```
xp# rmon show alarm
```

rmon show channels

Purpose

Shows the contents of the Filter Channel table.

Format

rmon show channels

Mode

Enable

Description

The **rmon show channels** command displays the contents of the Filter Channel table.

Parameters

None.

Restrictions

This command is only available if you have configured the Standard group.

Example

To show the contents of the Filter Channel table:

```
xp# rmon show channels
RMON 1 Channel Table
    No channels defined
```

rmon show cli-filters

Purpose

Displays previously-configured RMON CLI filters.

Format

rmon show cli-filters

Mode

User and Enable.

Description

The **rmon show cli-filters** command displays the RMON CLI filters that have been defined for use on the X-Pedition. Use the **rmon apply cli-filters** command to apply a filter to your current Telnet or Console session.

Parameters

None.

Restrictions

None.

Example

To show RMON CLI filters that are defined on the X-Pedition:

```
xp> rmon show cli-filters
RMON CLI Filters

❶  ❷
Id   Filter
--   ------
 1   (inpkts >= 0)
 2   (inpkts >= 0 and outoctets >= 0)
 3   srcmac 222222222222 and (outoctets >= 0)
You have selected a filter: (inpkts >= 0) ❸
```

Legend:

1. The filter ID. You use this value to apply a filter with the **rmon apply cli-filters** command.
2. The filter parameters that were specified with the **rmon set cli-filter** command.
3. This shows the parameters of the filter that is currently applied to your Telnet or Console session.

rmon show etherstats

Purpose

Displays Ethernet statistics for one or more ports.

Format

rmon show etherstats <*port-list*>|**all-ports**

Mode

Enable

Description

The **rmon show etherstats** command displays entries in the Ethernet table for one or more ports. Entries in this table are created automatically when default tables are turned on for the Lite group.

Parameters

<*port-list*>|**all-ports**
The port(s) for which you want Ethernet statistics displayed. Use the keyword **all-ports** to show Ethernet statistics on all ports.

Restrictions

The RMON agent reports only traffic *received* on a port. This command is available only if you have configured the Lite group.

Example

To display Ethernet statistics on a specified port:

```
xp# rmon show etherstats et.5.1
RMON I Ethernet Statistics Table
Index: 502, Port: et.5.1, Owner: monitor (1)
-----
RMON EtherStats              Total
---------------             -----
Octets                   83616016 (2)
Unicast Frames              86185 (3)
Broadcast Frames                0 (4)
Multicast Frames                0 (5)
Collisions                      0 (6)
64 Byte Frames                292 (7)
65-127 Byte Frames          16625
128-255 Byte Frames          6145
256-511 Byte Frames          4520
512-1023 Byte Frames         7992
1024-1518 Byte Frames        5011
```

Legend:

1. The EtherStats control table entry for this port:

   Index: uniquely identifies this entry.
   Port: port et.5.1.
   Owner: default owner “monitor.”

2. Number of octets of data received on the network.
3. Number of good frames received that were directed to a Unicast address.
4. Number of good frames received that were directed to a broadcast address.
5. Number of good frames received that were directed to a multicast address.
6. Number of collisions on this Ethernet segment.
7. Number of good and bad frames received, for various frame size ranges.

rmon show events

Purpose

Displays configured events and logs of triggered events.

Format

rmon show events

Mode

Enable

Description

The **rmon show events** command displays configured events and the logs, if any, of triggered events.

Parameters

None.

Restrictions

This command is only available if you have configured the Lite group.

Example

To show RMON events and logs:

```
xp# rmon show events
RMON I Event table

 ❶  ❷  ❸          ❹                ❺
Index Type Community      Description            Owner
   1 log  public         Log Only               Usama
No event logs found ❻
Index Type Community      Description            Owner
   2 both private        Log & Trap             Usama
No event logs found
```

Legend:

1. Index number that identifies this entry in the Event table.

2. Type of event: log, trap, or both log and trap.
3. Community string used for this event.
4. User-defined description of this event.
5. Owner of this event entry.

rmon show filters

Purpose

Shows the contents of the Filters table.

Format

rmon show filters

Mode

Enable

Description

The **rmon show filters** command displays the contents of the Filter table.

Parameters

None.

Restrictions

This command is only available if you have configured the Standard group.

Example

To show the contents of the Filter table:

```
xp# rmon show filters
RMON 1 Filter Table
    No filters defined
```

rmon show history

Purpose

Shows statistics over a period of time.

Format

rmon show history <*port-list*>|**all-ports**

Mode

Enable

Description

The **rmon show history** command displays statistical samples that are stored in the RMON History group. Entries in this table are created automatically when default tables are turned on for the Lite group.

Parameters

<*port-list*>|**all-ports**
The port(s) for which the history is to be displayed. Use the keyword **all-ports** to show history information on all the ports.

Restrictions

This command is only available if you have configured the Lite group.

Example

To display history information for a specific port:

```
xp# rmon show history et.5.1
RMON I History Table

❶ ❷       ❸          ❹    ❺
Index Port      Interval(secs) Buckets Owner
 502 et.5.1              300  50/50  monitor

 ❻ ❼              ❽    ❾      ❿    ⓫   ⓬   ⓭
 Index SysUpTime          Octets Packets  Bcst  Mcst  Colls  %Util  Other
   213 00D 17H 45M 47S    318114     336     0     0      0      0      0
   214 00D 17H 50M 47S    323928     341     0     0      0      0      0
   215 00D 17H 55M 48S    323586     335     0     0      0      0      0
   216 00D 18H 00M 49S    317186     320     0     0      0      0      0
   217 00D 18H 05M 49S    323470     333     0     0      0      0      0
                          .
                          .
                          .
   258 00D 21H 31M 03S    322264     312     0     0      0      0      0
   259 00D 21H 36M 03S    327944     315     0     0      0      0      0
   260 00D 21H 41M 04S    333138     309     0     0      0      0      0
   261 00D 21H 46M 06S    327782     312     0     0      0      0      0
   262 00D 21H 51M 07S    332268     294     0     0      0      0      0
```

Legend:

1. Index number that identifies the entry for this port in the History control table.
2. Port name.
3. Interval (in seconds) for data samples for each data bucket.
4. The actual number of buckets/the requested number of buckets.
5. Owner of this entry “monitor” (default).
6. Index number for this data bucket.
7. Time at which the sample was measured.
8. Total number of octets received on the network.
9. Number of packets received during the sampling period.
10. Number of good packets received during the sampling interval that were directed to a broadcast address.
11. Number of good packets received during the sampling interval that were directed to a multicast.
12. The number of collisions on this Ethernet segment during the sampling interval (best estimate).
13. The percentage of the network being utilized (best estimate).

rmon show host-top-n

Purpose

Displays the top *n* hosts.

Format

rmon show host-top-n

Mode

Enable

Description

The **rmon show host-top-n** command displays a report of the top hosts for a specified statistic. Note that the Host Top N report runs once. To run the reports again via the CLI, the control row must be disabled and then enabled. If the report has already been run, the Time Remaining field is set to zero. Otherwise, the Time Remaining field will be decremented until the report is run.

Restrictions

This command is only available if you have configured the Standard group and Host Top N control table entries exist.

Example

Consider the following command to gather the top n Host entries:

```
xp(config)# rmon host-top-n index 1 host-index 500 base out-octets duration 20 size 5
```

To display the Host Top N report, as specified by the previous command:

```
xp# rmon show host-top-n
RMON I HostTopN Table

❶ ❷ ❸ ❹ ❺ ❻ ❼ ❽
Index HostIndex RateBase  TimeRem Duration Buckets StartTime  Owner
1     500     Out-Octets 0    20   5/5   00D 00H 39M 29S  Usama

❾ ❿
Address          Rate
-------          ----
0080C8:C172A6    19911
00001D:CBA3FD    0
```

Legend:

1. Index number that identifies this entry in the Host Top N control table.
2. Index number that identifies the Host control table entry.
3. The parameter used to order the list of top “n” entries.
4. Number of seconds left in the report currently being collected.
5. Number of seconds that this report has collected during the last (or current) sampling interval.
6. Maximum number of hosts requested for the Top N table/maximum number of hosts in the Top N table.
7. The time of the sampling.
8. The owner of this entry.
9. The host address.
10. The value of the statistic for the host address.

rmon show hosts

Purpose

Shows statistics about the hosts discovered on the network.

Format

rmon show hosts <*port-list*>|**all-ports** [**summary**]

Mode

Enable

Description

The **rmon show hosts** command displays entries in the Hosts table for one or more ports. Entries in this table are created automatically when default tables are turned on for the Standard group.

If CLI filters have been applied, they will take effect when the Host table is displayed. This command will display control rows and their corresponding logs only if there are logs. A control row that has no data is not displayed.

Parameters

<*port-list*>|**all-ports**
The port(s) for which host information is to be shown. Use the keyword **all-ports** to show host information on all the ports.

summary
Use the keyword **summary** to show a summary of all control table rows with the number of logs in each row.

Restrictions

This command is only available if you have configured the Standard group and control table entries exist for the specified port.

Example

To show host information for a specific port:

```
xp# rmon show hosts et.5.1
RMON I Host Table
Index: 502, Port: et.5.1, Owner: monitor ❶

❷           ❸     ❹        ❺     ❻          ❼    ❽
Address        InPkts   InOctets    OutPkts  OutOctets     Bcst   Mcst
-------        ------  --------   -------  ---------      ----   ----
00001D:CBA3FD     88917   88436760     62132   5 095029     0        0
0080C8:C172A6     62132    5095029     88920   88437062     0        0
```

Legend:

1. Host control table information for this port:

   Index: number that identifies the entry for this port in the table.
   Port: port name.
   Owner: the default owner “monitor.”

2. MAC address of the discovered host.
3. Number of good packets transmitted to this address.
4. Number of good octets transmitted to this address.
5. Number of good packets transmitted from this address.
6. Number of good octets transmitted from this address.
7. Number of good packets transmitted by this address that were directed to a broadcast address.
8. Number of good packets transmitted by this address that were directed to a multicast address.

To show a summary of host information:

```
xp# rmon show all-ports summary
RMON I Host Table Summary

❶   ❷       ❸     ❹     ❺     ❻
Index Data Rows Port    Status Mode    Owner
----- --------- ----    ------ ----    -----
  500        1 et.5.1  Up     Address monitor
  501        1 et.5.2  Up     Address monitor
  502        0 et.5.3  Down   Flow    monitor
  503       17 et.5.4  Up     Flow    monitor
  504        0 et.5.5  Down   Flow    monitor
  505        0 et.5.6  Down   Flow    monitor
  506        0 et.5.7  Down   Flow    monitor
  507        0 et.5.8  Down   Flow    monitor
```

Legend:

1. Index number that identifies this entry in the Host control table.
2. Number of data rows associated with this index number.
3. Port.
4. Current state of the port.
5. Source of the data for this entry.
6. Owner of this entry.

rmon show matrix

Purpose

Shows statistics for source-destination address pairs.

Format

rmon show matrix <*port-list*>|**all-ports [summary] [order-by srcdst|dstsrc]**

Mode

Enable

Description

The **rmon show matrix** command displays entries in the Matrix table. Entries in this table are automatically created when default tables are turned on for the Standard group.

If CLI filters have been applied, they will take effect when the Matrix table is displayed. This command will display control rows and their corresponding logs only if there are logs. A control row that has no data is not displayed.

Parameters

<*port-list*>|**all-ports**
The port(s) for which you want to display information. Use the keyword **all-ports** to show matrix information on all the ports.

summary|order by
Use the keyword **summary** to display the control rows only. Use the keyword **order-by** to display entries by source/destination or by destination/source.

srcdst|dstsrc
Use the keyword **srcdst** to display the entries by source/destination. Use the keyword **dstsrc** to display entries by destination/source.

Restrictions

This command is only available if you have configured the Standard group.

Example

To show statistics for source-destination address pairs:

```
xp# rmon show matrix all-ports
RMON I Matrix Table

Port: et.5.1, Index: 500, Owner: monitor ❶

❷               ❸               ❹       ❺
SrcAddr         DstAddr         Packets   Octets
-------         -------         -------   ------
00001D:CBA3FD   0080C8:C172A6     3        264
0080C8:C172A6   00001D:CBA3FD     4        346

Port: et.5.5, Index: 504, Owner: monitor
SrcAddr         DstAddr         Packets   Octets
-------         -------         -------   ------
00001D:CBA3FD   080020:835CAA     3        246
080020:835CAA   00001D:CBA3FD     2        164
```

Legend:

1. The Matrix control table entry for this port:

   Port: the name of the port.
   Index: the index number for this port in the Matrix table.
   Owner: default “monitor.”

2. Source MAC address.

3. Destination MAC address.

4. Number of packets transmitted from the source to the destination address, including bad packets.

5. Number of octets transmitted from the source to the destination address.

To show control row summary statistics:

```
xp# rmon show matrix all-ports summary
RMON I Matrix Table Summary
 Index  Data Rows  Port   Status   Mode   Owner
-----   ---------  ----   ------   ----   ----
500       0      et.1.1   Up     Address monitor
501       0      et.1.2   Down   Address monitor
502       0      et.1.3   Down   Address monitor
503       0      et.1.4   Up     Address monitor
504       0      et.1.5   Down   Address monitor
505       0      et.1.6   Down   Address monitor
506       0      et.1.7   Down   Address monitor
507       0      et.1.8   Up     Address monitor
508       0      gi.4.1   Up     Address monitor
509       0      gi.4.2   Up     Address monitor
510       0      et.7.1   Up     Address monitor
511       0      et.7.2   Down   Address monitor
512       0      et.7.3   Down   Address monitor
513       0      et.7.4   Down   Address monitor
514       0      et.7.5   Down   Address monitor
515       0      et.7.6   Down   Address monitor
516       0      et.7.7   Down   Address monitor
517       0      et.7.8   Down   Address monitor
25        0     et.1.3   Down   Address
```

rmon show nl-host

Purpose

Shows the amount of traffic to and from each network address.

Format

rmon show nl-host <*port-list*>|**all-ports** [**summary**]

Mode

Enable

Description

The **rmon show nl-host** command shows entries in the RMON 2 Network Layer Host table for one or more ports. Entries in this table are created automatically when default tables are turned on for the Professional group.

If CLI filters have been applied, they will take effect when the Network Layer host table is displayed. This command shows control rows and their corresponding logs only if there are logs. A control row with no data will not appear in the report.

Parameters

<*port-list*>|**all-ports**
The port(s) for which you want to display traffic information. Use the keyword **all-ports** to show information on all the ports.

summary
Use the keyword **summary** to display control row summary information only.

Restrictions

This command is only available if you have configured the Professional RMON group and control table entries exist for the specified port.

Example

To display the network layer host table for all ports:

```
xp# rmon show nl-host all-ports

RMON II Network Layer Host Table

Index: 500, Port: et.5.1, Inserts: 3, Deletes: 0, Owner: monitor ❶

❷           ❸     ❹       ❺      ❻       ❼
Address         InPkts  InOctets   OutPkts  OutOctets  Protocol
-------         ------  --------   -------  ---------  --------
10.60.89.88       1159    952300         2        164  *ether2.ip-v4
192.100.81.1         1       100         1        100  *ether2.ip-v4
192.100.81.3         3       264      1160     952400  *ether2.ip-v4

Index: 504, Port: et.5.5, Inserts: 2, Deletes: 0, Owner: monitor
Address         InPkts  InOctets   OutPkts  OutOctets  Protocol
-------         ------  --------   -------  ---------  --------
10.60.89.88          3       246      1220      98962  *ether2.ip-v4
192.100.81.3      1220     98962         3        246  *ether2.ip-v4
```

Legend:

1. The control table entry for this port:

   Index: index number that identifies this entry in the hl host control table.
   Port: name of port.
   Inserts: number of inserts in the network layer host table for this entry.
   Deletes: number of deletions in the network layer host table for this entry.
   Owner: the entity that configured this entry.

2. The network address.

3. Number of packets received by this network address.

4. Number of octets received by this network address.

5. Number of packets sent by this network address.

6. Number of octets sent by this network address.

7. The protocol, as defined in the RMON Protocol Directory for the X-Pedition. Note that this shows the network layer protocol encapsulations only. If you want to see application/protocol information, such as the destination socket, use the **rmon show al-host** command.

rmon show nl-matrix

Purpose

Shows information about the traffic between network address pairs.

Format

rmon show nl-matrix <*port-list*>**|all-ports [order-by srcdst|dstsrc] [summary]**

Mode

Enable

Description

The **rmon show nl-matrix** command shows entries in the Network Layer Matrix table for one or more ports. Entries in this table are created automatically when default tables are turned on for the Professional group**.**

If CLI filters have been applied, they will take effect when this table is displayed. The control rows and their corresponding logs are displayed only if there are logs. A control row with no data will not appear in the report.

Parameters

<*port-list*>**|all-ports**
The port(s) for which you want to display network layer traffic information. Use the keyword **all-ports** to show information for all ports.

order-by srcdst
Orders the logs by source address, then destination address (default).

order-by dstsrc
Orders the logs by destination address, then source address.

summary
Use the keyword **summary** to display control row summary information only.

Restrictions

This command is only available if you have configured the Professional group and control table entries exist for the specified port.

Example

To show the Network Layer Matrix table for all ports:

```
xp# rmon show nl-matrix all-ports
RMON II Network Layer Matrix Table

Index: 500, Port: et.5.1, Inserts: 4, Deletes: 0, Owner: monitor ❶

❷           ❸             ❹        ❺      ❻
SrcAddr       DstAddr          Packets    Octets  Protocol
-------       -------          -------    ------  --------
10.60.89.88    192.100.81.3          2       164  *ether2.ip-v4
192.100.81.1   192.100.81.3          1       100  *ether2.ip-v4
192.100.81.3   10.60.89.88        1241   1025436  *ether2.ip-v4
192.100.81.3   192.100.81.1          1       100  *ether2.ip-v4

Index: 504, Port: et.5.5, Inserts: 2, Deletes: 0, Owner: monitor
SrcAddr       DstAddr          Packets    Octets  Protocol
-------       -------          -------    ------  --------
10.60.89.88    192.100.81.3       1302    105604  *ether2.ip-v4
192.100.81.3   10.60.89.88           3       246  *ether2.ip-v4
```

Legend:

1. The control table entry for this port:

   Index: index number that identifies this entry in the control table.
   Port: name of port.
   Inserts: number of inserts in the Network Layer Matrix table for this entry.
   Deletes: number of deletions in the Network Layer Matrix table for this entry.
   Owner: the entity that configured this entry.

2. Source network address.

3. Destination network address.

4. Number of packets transmitted without error from the source to the destination.

5. Number of octets transmitted without error from the source to the destination.

6. The protocol, as specified in the RMON Protocol Directory for the X-Pedition.

rmon show nl-matrix-top-n

Purpose

Reports the top *n* Network Layer Matrix entries, sorted by a specific metric.

Format

rmon show nl-matrix-top-n

Mode

Enable

Description

The rmon show nl-matrix-top-n command shows entries in the RMON 2 Network Layer Matrix Top N table.

Parameters

None.

Restrictions

This command is only available if you have configured the Professional group and entries exist in the Network Layer Matrix Top N table.

Example

Consider the following command to gather the top n Network Layer Matrix entries:

```
xp(config)# rmon nl-matrix-top-n index 1 matrix-index 500 ratebase all-octets duration 20 size 5
```

To show the top n entries in the Network Layer Matrix table, as specified by the previous command:

```
xp# rmon show nl-matrix-top-n
RMON II Nl Matrix Table

❶ ❷ ❸ ❹ ❺ ❻ ❼ ❽ ❾
Index M-Index RateBase TimeRem Duration Size StartTime Reports Owner
  1   500 Octets 20 20 5 00D 00H 51M 37S 1 Usama

❿ ⓫ ⓬ ⓭ ⓮ ⓯ ⓰
SrcAddr DstAddr PktRate R-PktRate OctetRate R-OctetRate Protocol
------- ------- ------- --------- --------- ----------- --------
192.100.81.3 10.60.89.88 23 0 19986 0 *ether2.ip-v4
192.100.81.1 192.100.81.3 0 0 0 0 *ether2.ip-v4
192.100.81.3 192.100.81.1 0 0 0 0 *ether2.ip-v4
10.60.89.88 192.100.81.3 0 23 0 19986 *ether2.ip-v4
```

Legend:

1. Index number that identifies this entry in the network layer Matrix Top N control table.
2. The Network Layer Matrix table for which the top N report is shown.
3. The parameter on which the entries are sorted.
4. Number of seconds left in the report currently being collected.
5. Number of seconds that this report has collected during the last sampling interval.
6. Maximum number of matrix entries in this report.
7. The time when this report was last started.
8. The number of reports generated by this entry.
9. The entity that configured this entry.
10. Network address of the source host.
11. Network address of the destination host.
12. Number of packets from the source to the destination during the sampling interval.
13. Number of packets from the destination to the source during the sampling interval.
14. Number of octets from the source to the destination during the sampling interval.
15. Number of octets from the destination to the source during the sampling interval.
16. The protocol, as defined in the RMON Protocol Directory for the X-Pedition.

rmon show packet-capture

Purpose

Shows packets captured after flowing through a channel.

Format

rmon show packet-capture control-table [captured-packets <*control-index*>**]**

Mode

Enable

Description

The **rmon show packet-capture** command shows the buffer table for captured packets. Before you use this command, first configure the Filter Channel group using the **rmon channel index** command. Then use the **rmon capture** command to configure the Packet Capture group which allows packets to be captured after they have flowed through a channel.

Parameters

control-table

Displays RMON packet capture filter information. Each packet captured belongs to one entry of the control-table.

captured-packets <*control-index*>

Displays all of the packets captured for RMON. If you supply the optional *control-index*, you will display only packets captured for the specified control index; otherwise, you will display all captured packets. To determine the *control-index*, use the command "**rmon show packet-capture control-table**."

Restrictions

- This command is available only if you have enabled the Standard RMON groups.
- Packet capture using RMON uses considerable CPU cycles. For best results, enable packet capture when the CPU is not busy running other important tasks and CPU utilization is low.

rmon show probe-config

Purpose

Shows the configuration of the X-Pedition for interaction with other RMON devices.

Format

rmon show probe-config [basic] [net-config] [trap-dest]

Mode

Enable

Description

The **rmon show probe-config** command shows entries in the RMON 2 Probe Configuration table.

Parameters

basic	Shows basic probe configuration information.
net-config	Shows network configuration table.
trap-dest	Shows trap destination table.

Restrictions

This command is only available if you have configured the Professional group.

rmon show protocol-directory

Purpose

Displays the protocols that the X-Pedition can monitor with RMON.

Format

rmon show protocol-directory <*protocol*>|**all-protocols**

Mode

Enable

Description

The **rmon show protocol-directory** command displays the protocol encapsulations that are defined in the RMON 2 Protocol Directory group for the X-Pedition.

Parameters

<*protocol*>|**all-protocols**
The specific protocol encapsulation that is managed with the RMON 2 Protocol Directory group. (See Appendix A for protocol encapsulations that are supported on the X-Pedition.) Use the keyword **all-protocols** to display all protocol encapsulations that are managed with the Protocol Directory group.

Restrictions

This command is only available if you have configured the Professional group.

Example

To show all protocol encapsulations that are managed with the Protocol Directory group:

:

```
xp# rmon show protocol-directory all-protocols
RMON II Protocol Directory Table

Last Change: 00D 00H 00M 00S
Index AddrMap Host Matrix Status Protocol
1     Off    Off  Off    Active ether2
2     NA     Off  Off    Active idp
3     NA     Off  Off    Active ip-v4
4     NA     Off  Off    Active chaosnet
5     NA     Off  Off    Active arp
6     NA     Off  Off    Active rarp
7     NA     Off  Off    Active vip
8     NA     Off  Off    Active vloop
9     NA     Off  Off    Active vloop2
10    NA     Off  Off    Active vecho
11    NA     Off  Off    Active vecho2
12    NA     Off  Off    Active ipx
13    NA     Off  Off    Active netbios-3com
14    NA     Off  Off    Active atalk
15    NA     Off  Off    Active aarp

...
```

NOTE: The example above shows a partial listing only.

rmon show protocol-distribution

Purpose

Shows the octets and packets detected for different protocols on a network segment.

Format

rmon show protocol-distribution <*port-list*>|**all-ports**

Mode

Enable

Description

The **rmon show protocol-distribution** command displays the RMON 2 Protocol Distribution table. This table contains a list of protocols, defined in the RMON 2 Protocol Directory, that are discovered by the X-Pedition. Entries in this table are created automatically when default tables are turned on for the Professional group. If you delete an entry in the Protocol Directory, then entries in this table associated with the deleted protocol are also deleted.

If CLI filters have been applied, they will take effect when the Protocol Distribution table is displayed.

Parameters

<*port-list*>|**all-ports**
The port(s) for which you want to show protocol distribution. Use the keyword **all-ports** to show protocol distribution information on all the ports.

Restrictions

This command is only available if you have configured the Professional group.

Example

To show the RMON 2 Protocol Distribution table:

:

```
xp(config)# rmon show protocol-distribution all-ports
RMON II Protocol Distribution Table

Index: 508, Port: gi.4.1, Owner: monitor
Pkts Octets Protocol
---- ------ --------
3312 304550 ether2
3312 304550 ip-v4
2459 234564 icmp
 853  69986 tcp
 853  69986 telnet
```

rmon show status

Purpose

Displays RMON status, groups, enabled ports, and memory utilization.

Format

rmon show status

Mode

Enable

Description

The **rmon show status** command shows whether RMON is enabled, the RMON groups that are configured, the ports on which RMON is enabled, and the memory allocated and used by RMON.

Parameters

None.

Example

To show RMON status:

```
xp# rmon show status
RMON Status
-----------
* RMON is ENABLED ❶
* RMON initialization successful.

+--------------------------+
| RMON Group Status        | ❷
+-------+--------+---------+
| Group | Status | Default |
+-------+--------+---------+
| Lite  |    On  |    Yes  |
+-------+--------+---------+
| Std   |    On  |    Yes  |
+-------+--------+---------+
| Pro   |    On  |    Yes  |
+-------+--------+---------+

RMON is enabled on: et.5.1, et.5.2, et.5.3, et.5.4, et.5.5, et.5.6, et.5.7, et.5.8 ❸

RMON Memory Utilization ❹
-----------------------
      Total Bytes Available:   48530436

Total Bytes Allocated to RMON:    4000000
         Total Bytes Used:    2637872
         Total Bytes Free:    1362128
```

Legend:

1. When the X-Pedition is booted, RMON is off by default. RMON is enabled with the **rmon enable** command.
2. Shows which RMON group (Lite, Standard, or Professional) is configured and whether default control tables are turned on.
3. Shows the ports on which RMON is enabled.
4. Shows RMON memory utilization. You can adjust the amount of memory allocated to RMON with the **rmon set memory** command.

rmon show user-history

Purpose

Shows user-defined collection of historical information from MIB objects on the X-Pedition.

Format

rmon <*string*> **show user-history [all-indexes]**

Mode

Enable

Description

The **rmon show user-history** command shows the User History table.

Parameters

<*string*> Specifies a particular rmon.

all-indexes This optional parameter displays all indexes.

Restrictions

This command is only available if you have configured the Professional group.

rmon user-history-apply

Purpose

Applies a specified group to the User History control table.

Format

rmon user-history-apply <*groupname*> **to** <*user-history-index*> [**status enable|disable**]

Mode

Configure

Description

The **rmon user-history-apply** command applies all objects in the group created with the **rmon user-history-objects** command to the row in the User History control table. If the number of objects specified in the control row is greater than those in the group, the remaining OIDs are set to 0.0. If the number of objects specified in the control row is less than those in the group, the remaining are discarded.

Parameters

<*groupname*>
Is the name of a group of objects that has been created with the **rmon-user-history-objects** command.

<*user-history-index*>
Specifies the row in the User History control table.

Restrictions

None.

rmon user-history-control

Purpose

Monitors a group of objects (OIDs) over a period of time.

Format

rmon user-history-control index <*index-number*> **objects** <*number*> **samples** <*number*> **interval** <*number*> [**owner** <*string*>] [**status enable**|**disable**]

Mode

Configure

Description

The **rmon user-history-control** command monitors the group of objects that are defined with the **rmon user-history-objects** command. This command creates an entry in the User History control table.

Use the **rmon show user-history** command to display the User History table.

Parameters

<*index-number*>
Is a number between 1 and 65535 that uniquely identifies a row in the user history control table.

objects <*number*>
Specifies the number of MIB objects to be collected.

samples <*number*>
Specifies the number of discrete time intervals over which data is to be saved.

interval <*number*>
Specifies the interval, in seconds, between samples.

owner <*string*>
Specifies the owner of the event; for example, an IP address, machine name or person's name.

status enable|disable
Enables or disables this matrix. The default is enable.

Restrictions

None.

rmon user-history-objects

Purpose

Defines a group of objects (OIDs).

Format

rmon user-history-objects <*groupname*> **variable** <*oid*> **type absolute-value|delta-value** [**status enable|disable**]

Mode

Configure

Description

The **rmon user-history-objects** command defines the group of objects that can be monitored with the **rmon user-history-control** command. This command creates a group with a single OID as a member of the group. To add several objects to the group, you need to issue multiple **user-history-objects** commands. Each object appears as a separate row in the User History control table.

Parameters

<*groupname*>
Is the name of the group of objects.

variable <*oid*>
Specifies the object identifier to be monitored.

type absolute-value|delta-value
Specifies the method of sampling for the selected variable.

interval <*number*>
Specifies the interval, in seconds, between samples.

status enable|disable
Enables or disables this matrix. The default is enable.

Restrictions

None.

Chapter 59

save Command

The **save** command saves the configuration changes you have entered during the current CLI session. You can save the configuration commands in the scratchpad to the active configuration, thus activating changes. You then can save the active changes to the Startup configuration.

Format

save active|startup

Mode

Configure

Note: If you are in Enable mode, you still can save the active configuration changes to the Startup configuration file by entering the **copy active to startup** command.

Description

Saves configuration changes.

- If you use the **active** keyword, uncommitted changes in the scratchpad are *activated*. The X-Pedition accumulates configuration commands in the scratchpad until you activate them, clear them, or reboot. The X-Pedition runs these commands when you activate the changes.
- If you use the **startup** keyword, the X-Pedition saves the *active* configuration to the Startup file. Any non-committed commands in the Scratchpad are ignored.

Parameters

active | startup Specifies the destination for the configuration commands you are saving.

Restrictions

None.

Chapter 60

show Command

Purpose

The **show** command displays the configuration of your running system.

Format

show active| scratchpad| startup| failed

Mode

Configure

Description

The **show** command displays the configuration of your running system as well as any non-committed changes in the scratchpad. Each CLI command is preceded with a number. This number can be used with the **negate** command to negate one or more commands. If you see the character **E** (for Error) immediately following the command number, it means the command did not execute successfully due of an earlier error condition. To get rid of the command in error, you can either negate it or fix the original error condition.

There are three modes for the **show** command: **active**, **scratchpad**, and **startup**. Specifying **active** shows you the configuration that are currently active on the router. Specifying **scratchpad** shows you the configuration currently in the scratchpad but have yet to be applied as active. Specifying **startup** shows the configuration that will be applied to the next bootup. You must specify one of these three modes as a parameter for the show command.

When viewing the active configuration file, the CLI displays the configuration file command lines with the following possible annotations:

- Commands without errors are displayed without any annotation.
- Commands with errors are annotated with an “E”.

- If a particular command has been applied such that it can be expanded on additional interfaces/modules, then it is annotated with a “P”. For example, if you enable STP on all ports in the current system, but the X-Pedition contains only one module, then that particular command will be extended to all modules when they have been added to the X-Pedition.

  A command like **stp enable et.*.*** would be displayed as follows:

  ```
  P: stp enable et.*.*
  ```

  indicating that it is only partially applied. If you add more modules to the X-Pedition at a later date and then update the configuration file to encompass all of the available modules in the X-Pedition, then the “P:” portion of the above command line would disappear when displaying this configuration file.

  If a potentially partial command, which was originally configured to encompass all of the available modules on the X-Pedition, becomes only partially activated (after a hotswap or some such chassis reconfiguration), then the status of that command line will automatically change to indicate a partial completion status, complete with “P:”.

Note: Commands with no annotation or annotated with a “P:” are not in error.

Parameters

active	Specify this parameter to show the configuration that are currently active on the router.
scratchpad	Specify this parameter to show the configuration currently in the scratchpad but have yet to be applied as active.
startup	Specify this parameter to show the configuration that will be applied to the next bootup.
failed	This parameter shows all active configuration lines on the router that are in partial or total error.

Restrictions

None.

Examples

The following command shows the active configuration:

```
xp(config)# show active
Running system configuration:
    !
    ! Last modified from Console on 2000-02-09 13:00:46
    !
 1E: atm create vcl port at.9.1.1.200
    !
 2E: interface create ip pos11 address-netmask 20.11.11.20/24 peer-address 20.11
.11.21 type point-to-point port so.13.1
 3E: interface create ip atm1 address-netmask 12.1.1.1/24 port at.9.1.1.200
 4 : interface add ip en0 address-netmask 134.141.179.147/27
    !
 5 : ip add route 134.141.173.0/24 gateway 134.141.179.129
 6 : ip add route 134.141.176.0/24 gateway 134.141.179.129
 7 : ip add route 134.141.172.0/24 gateway 134.141.179.129
    !
 8 : system set idle-timeout telnet 0
 9 : system set idle-timeout serial 0
```

The following command shows the configuration currently in the scratchpad:

```
xp(config)# show scratchpad

******** Non-committed changes in Scratchpad ********
 1*: atm define service service1 srv-cat cbr pcr 100000
    !
 2*: vlan create vlan1 ip id 5
    !
 3*: ip add route default host gateway 100.0.0.1
```

The following command shows the configuration saved for startup at next bootup:

```
xp(config)# show startup
!
! Startup configuration for the next system reboot
!
! Last modified from Console on 2001-12-28 16:51:19
!
version 3.1
atm create vcl port at.9.1.1.200
interface create ip pos11 address-netmask 20.11.11.20/24 peer-address 20.11.11.2
1 type point-to-point port so.13.1
interface create ip atm1 address-netmask 12.1.1.1/24 port at.9.1.1.200
interface add ip en0 address-netmask 134.141.179.147/27
ip add route 134.141.173.0/24 gateway 134.141.179.129
ip add route 134.141.176.0/24 gateway 134.141.179.129
ip add route 134.141.172.0/24 gateway 134.141.179.129
system set idle-timeout telnet 0
system set idle-timeout serial 0
pos set so.13.1 working protecting so.13.2
```

Chapter 61

smarttrunk Commands

SmartTRUNK ports are groups of ports that have been logically combined to increase throughput and provide link redundancy. The **smarttrunk** commands let you display and set parameters for SmartTRUNK ports. For additional information regarding SmartTRUNKs, see the *Enterasys X-Pedition User Reference Manual*.

Command Summary

Table 47 lists the **smarttrunk** commands. The sections following the table describe the command syntax.

Table 47. smarttrunk commands

smarttrunk add ports <*port list*> **to** <***smarttrunk***>
smarttrunk clear load-distribution <***smarttrunk***>
smarttrunk create <*smarttrunk*> **protocol huntgroup** |**no-protocol** |**lacp [no-llap-ack]**
smarttrunk lacp actor-parameters port <*port_list*> **enable** **[port-key** <*number*>**] [port-priority** <*number*>**] [activity active**| **passive]** **[aggregation aggregatable**| **individual] [timeout long**| **short]**
smarttrunk lacp aggregator <*smarttrunk*> **port-type 10-100-Ethernet**| **Gigabit-Ethernet** **actor-key** <*number*>| **default-10-100**| **default-gig [system-priority** <*number*>**]**
smarttrunk set load-policy on <***smarttrunk***> <*load-policy*>
smarttrunk show <*option*>

smarttrunk add ports

Purpose

Adds physical ports to a SmartTRUNK.

Format

smarttrunk add ports <*port list*> **to** <***smarttrunk***>

Mode

Configure.

Description

The **smarttrunk add ports** command allows you to add the ports specified in <*port_list*> to a SmartTRUNK. Before you can add the ports, you must create a SmartTRUNK with the **smarttrunk create** command and set all SmartTRUNK ports to full duplex. See *smarttrunk create* on page 1041 for information on creating SmartTRUNKs.

Note: The DEC Hunt Group control protocol is limited to 256 ports. On the ER-16, you may configure SmartTRUNKs that use the DEC Hunt Group control protocol on slots 1-7 only.

Parameters

<*port_list*>	The port(s) you will add to an existing SmartTRUNK. All the ports in the SmartTRUNK must be connected to the same destination.
<***smarttrunk***>	The name of the existing SmartTRUNK to which you will add physical ports.

Restrictions

Ports added to a SmartTRUNK must:

- Be set to full duplex
- Be members of the same VLAN
- Share identical properties (e.g., L2 aging, STP state)

Note: Do not use the **add ports** command to add ports to an LACP SmartTRUNK. When you use the **lacp actor-parameters** command to enable LACP, the X-Pedition adds ports to the SmartTRUNK dynamically.

Example

To add ports et.1.1, et.1.2, and et.1.3 to SmartTRUNK st.1:

```
xp(config)# smarttrunk add ports et.1.(1-3) to st.1
```

smarttrunk clear load-distribution

Purpose

Clears load distribution statistics for ports in a SmartTRUNK.

Format

smarttrunk clear load-distribution <***smarttrunk*** *list*> | **all-smarttrunks**

Mode

Enable.

Description

The **smarttrunk clear load-distribution** command allows you to reset load distribution statistics to zero. This command is used in conjunction with the **smarttrunk show distribution** command to gather statistics for the transmitted bytes per second flowing through the SmartTRUNK and each port in it.

Parameters

<***smarttrunk*** *list* >	The name of one or more existing SmartTRUNKs.
all-smarttrunks	Clears load distribution information for all SmartTRUNKs.

Restrictions

None.

Example

To clear load distribution information from SmartTRUNK st.1, enter the following:

```
xp# smarttrunk clear load-distribution st.1
```

smarttrunk create

Purpose

Create a SmartTRUNK and specify a control protocol for it.

Format

smarttrunk create <*smarttrunk*> **protocol huntgroup| no-protocol| lacp [no-llap-ack]**

Mode

Configure.

Description

The **smarttrunk create** command allows you to create a SmartTRUNK logical port. Once you create a SmartTRUNK port, you can add physical ports to it with the **smarttrunk add ports** command.

SmartTRUNKs on the X-Pedition are compatible with the DEC Hunt Groups control protocol. If you connect the SmartTRUNK to another X-Pedition, Enterasys switch, or Digital GIGAswitch/Router, you can specify that the SmartTRUNK use this control protocol. SmartTRUNKing and Hunt Groups are composed of two protocols:

- Logical Link Aging Protocol (LLAP) – Assists in learning and aging
- Physical Link Affinity Protocol (PLAP) – Monitors and maintains the trunking states

SmartTRUNKs are also compatible with devices that do not support the Hunt Groups control protocol, such as those that support Cisco's EtherChannel technology. If you connect a SmartTRUNK to devices that do not support Hunt Groups, *no control protocol is used*. You must specify the **no-protocol** keyword in the **smarttrunk create** command.

Parameters

<***smarttrunk***> The name of the SmartTRUNK to create, in the form st.*x* (e.g., st.1).

- The maximum number of SmartTRUNKs you may configure depends on the router you are using:

  X-Pedition 2000 allows up to 40
  X-Pedition 8000 allows up to 64
  X-Pedition 8600 allows up to 128
  ER-16 allows up to 256

Note: When using firmware version E9.1.0.0 or later, the ER-16 supports SmartTRUNKs that use the lower index ranges of 1-240 only (i.e., st.1–st.240).

huntgroup	Specifies that the DEC Hunt Group control protocol be used. Use this keyword if you connect the SmartTRUNK to another X-Pedition, Enterasys switch, or Digital GIGAswitch/Router.
no-protocol	Specifies that no control protocol be used. Use this keyword if the SmartTRUNK connects to a device that does not support the DEC Hunt Group control protocol.
lacp	This option specifies that the 802.3 Link Aggregation Control Protocol be used. (Use this keyword if you are creating a SmartTRUNK for an aggregator.)
no-llap-ack	By default, the **X-Pedition** sends out extra LLAP ack packets for backward compatibility with some Cabletron products. Select this option to stop these extra packets.

Restrictions

When using firmware version E9.1.0.0 or later, the ER-16 supports SmartTRUNKs that use the lower index ranges of 1-240 only (i.e., st.1–st.240).

Example

The following command creates a SmartTRUNK, st.1, that uses the DEC Hunt Group control protocol.

```
xp(config)# smarttrunk create st.1 protocol huntgroup
```

smarttrunk lacp actor-parameters

Purpose

Enable Link Aggregation Control Protocol (LACP) and set actor parameters on a port to run link aggregation.

Format

smarttrunk lacp actor-parameters port <*port_list*> **enable [port-key** <*number*>**] [port-priority** <*number*>**] [activity active**| **passive] [aggregation aggregatable**| **individual] [timeout long**| **short]**

Mode

Configure.

Description

The **smarttrunk lacp actor-parameters** command allows you to enable LACP and set the actor parameters on a port to run link aggregation with the 802.3ad Link Aggregation Control Protocol (LACP).

Specify Actor Parameters for LACP (Link Aggregation Control Protocol)

To set the actor parameters, do the following:

Enable LACP on a port or series of ports.	**smarttrunk lacp actor-parameters port** <*port_list*> **enable**
Set the administrative key for the port.	**smarttrunk lacp actor-parameters port** <*port_list*> **enable port-key** <*number*>
Set the administrative priority for the port.	**smarttrunk lacp actor-parameters port** <*port_list*> **enable port-priority** <*number*>
Set the administrative LACP activity for the port.	**smarttrunk lacp actor-parameters port** <*port_list*> **enable activity active**| **passive**
Set the administrative aggregation for the port.	**smarttrunk lacp actor-parameters port** <*port_list*> **enable aggregation aggregatable**| **individual**
Set the administrative LACP Timeout for the port.	**smarttrunk lacp actor-parameters port** <*port_list*> **enable timeout long**| **short**

Parameters

port <*port-list*>	The port number for which to set the parameters.
enable	Enables LACP on the specified ports.
port-key <*number*>	Sets the admin key (1-65536) for the port (optional). The default port-key is **1** for 10-100 ports and **2** for gig ports.
port-priority <*number*>	Sets the admin priority (1-65536) for the port (optional). The default priority is **1**.
activity active| **passive**	Sets the admin LACP activity (active or passive) for the port (optional). By default, the port is **active**.
aggregation aggregatable| **individual**	Sets the admin aggregation (aggregate or individual) for the port (optional). By default, the port setting is **aggregate**.
timeout long| **short**	Defines the admin LACP timeout (long or short) for the port (optional). By default, the timeout is **short**.

Note: Any parameter for which you do not specify a value will use its default setting.

Restrictions

None.

smarttrunk lacp aggregator

Purpose

Set properties of the aggregator.

Format

smarttrunk lacp aggregator <*smarttrunk*> **port-type 10-100-Ethernet**| **Gigabit-Ethernet actor-key** <*number*>| **default-10-100**| **default-gig [system-priority** <*number*>**]**

Mode

Configure.

Description

The **smarttrunk lacp aggregator** command allows you to define aggregator properties.

Parameters

aggregator <smarttrunk>

Use this parameter to identify the SmartTRUNK you will configure.

port-type 10-100-Ethernet| **Gigabit-Ethernet**

Defining the **port-type** parameter specifies whether the ports associated with the aggregator are 10/100 Ethernet ports or Gigabit Ethernet ports.

actor-key <*number*>| **default-10-100**| **default-gig**

The **actor-key** parameter specifies the administrative key for the aggregator. To use a default value, enter one of the following: **default-10-100** or **default-gig**. If you define more than one LACP SmartTRUNK on the same router, the SmartTRUNKs cannot share the same key values.

system-priority <*number*>

Sets the priority (1-65536) of the system (optional). The default priority is **1**.

Restrictions

Multiple LACP SmartTRUNKs cannot share the same default values—each SmartTRUNK must use a different default value.

smarttrunk set load-policy

Purpose

Specify traffic distribution among SmartTRUNK ports.

Format

smarttrunk set load-policy round-robin |link-utilization on <*smarttrunk-list*>| all-smarttrunks

Mode

Configure.

Description

The **smarttrunk set load-policy** command lets you specify how a SmartTRUNK distributes traffic among its ports.

Parameters

round-robin	Round-robin (the default) assigns flows to ports on a sequential basis. The first flow goes to the first port in the SmartTRUNK, the second flow to the second port, and so on. This distributes traffic evenly across all ports.
link-utilization	Sends packets to the least-used port in the SmartTRUNK.
<***smarttrunk****-list* >	Sends packets to one or more specific SmartTRUNKs.
all-smarttrunks	Apply the command to all SmartTRUNKs.

Restrictions

None.

Example

To specify that SmartTRUNK st.1 distribute flows sequentially among its component ports:

```
xp(config)# smarttrunk set load-policy on st.1 round-robin
```

smarttrunk show

Purpose

Displays information about SmartTRUNKs on the X-Pedition.

Format

smarttrunk show trunks

smarttrunk show distribution| **protocol-state**| **connections** <***smarttrunk*** *list*>| **all-smarttrunks**

Mode

Enable.

Description

The **smarttrunkshow** command displays statistics about SmartTRUNKs on the X-Pedition.

Parameters

connections	Shows information about the SmartTRUNK connection, including the MAC address of the remote switch, and the module number and port number of each remote port. Connection information is reported only if the Hunt Group protocol is enabled for the SmartTRUNK.
protocol-state	Shows information about the control protocol on a SmartTRUNK.
distribution	Provides statistics on traffic distribution across the ports in a SmartTRUNK.
trunks	Shows information about all SmartTRUNKs, including active and inactive ports, and the control protocol used.
lacp-control	Shows the lacp parameters.
lacp-stats	Shows the lacp statistic counters.
show lags	Shows the lacp LAGs and their members.
<***smarttrunk*** *list*>	The name of one or more specific SmartTRUNK for which you will display statistics.
all-smarttrunks	Apply the command to all SmartTRUNKs.

Restrictions

None.

Examples

To display information about all SmartTRUNKs on the X-Pedition:

```
xp# smarttrunk show trunks

Flags: D - Disabled  I - Inactive

SmartTRUNK   Active Ports       Inactive Ports      Primary     Port     Protocol    Load-Policy    Flags
----------   ------------       --------------      -----------  -------- -----------   -----
st.1         et.3.(7-8)         None                None        RR
```

To show how traffic is distributed across the ports on all SmartTRUNKs:

```
xp# smarttrunk show distribution all-smarttrunks

SmartTRUNK      Member      %TX  Util.      %RX  Util.      Link Status     Grp Status
-----------------  -----------  -------------  -------------  ----------  -----------
st.1            et.2.4      0.00           0.00           Forwarding     Up
st.1            et.2.5      0.00           0.00           Forwarding     Up
st.1            et.2.6      0.00           0.00           Forwarding     Up
```

To show information about the control protocol for SmartTRUNK st.1:

```
xp# smarttrunk show protocol-state st.1

SmartTRUNK     Protocol        State     Port     Port State
----------     --------        -----     ----     ----------
st.1           HuntGroup       Down    et.3.1     Negotiate
                                       et.3.2     Negotiate
```

To show connection information for all SmartTRUNKs:

```
xp# smarttrunk show connections all-smarttrunks
SmartTRUNK    Local Port    Remote Switch          Remote Module    Remote Port    State
----------    ----------    -------------          -------------    -----------    -----
st.1          et.2.1        Enterasys A9:6E:57     3                1              Up
st.1          et.2.2        Enterasys A9:6E:57     3                2              Up
st.1          et.2.3        Enterasys A9:6E:57     3                3              Up
st.1          gi.3.1        Enterasys A9:6E:57     4                5              Up
st.2          et.2.4        --                     --               --             Up
st.2          et.2.5        --                     --               --             Up
st.2          et.2.6        --                     --               --             Up
```

Note: In the example above, SmartTRUNK st.2 has no control protocol enabled, so no connection information is reported.

Chapter 62

snmp Commands

The **snmp** commands let you set and show SNMP parameters including SNMP community names and IP host targets for SNMP traps.

Note: In order to run NetFlow, you must enable SNMP.

Command Summary

Table 48 lists the **snmp** commands. The sections following the table describe the command syntax.

Table 48. snmp Commands

snmp disable trap authentication| link-up-down| frame-relay| ospf| spanning-tree| bgp| vrrp| environmentals
snmp disable port-trap <*port list*>
snmp set chassis-id <*chassis-name*>
snmp set community <*community*> **name** <*community-name*> **security-name** <*security-name*> **engine-id** <*engine-id-string*> **tag** <*tag-string*>
snmp set community-to-group <*security-name*> **to** <*group-name*> **[v1 | v2c]**
snmp set filter <*filter-name*> **subtree** <*OID*> **mask** <*mask-string*> **[category bgp| dot1dbridge| ds3| frame-relay-dte| interfaces| ipx| ospf| rmon| snmp| xp-enterprise| vrrp] [type included| excluded]**
snmp set group <*group-name*> **[v1 | v2c | v3 [auth | noauth | priv]] read** <*readview*> **write** <*writeview*> **notify** <*notifyview*>
snmp set if-alias <*interface-name*> **alias** <*alias-name*>
snmp set mib name <*mib-name*> **status enable| disable**

Table 48. snmp Commands

snmp set notify <*notify-name*> **tag** <*tag-string*> **[type trap | inform]**
snmp set retro-mib-ifspeed
snmp set target [<*target-name*> | <*ip-address*>**] ip-address** <*ip-address*> **param** <*param-name*> **owner** <*owner-name*> **[v1 | v2c | v3 [auth | noauth | priv]] [community** <*community-name*> **| security** <*security-name*>**] port** <*udp-port*> **timeout** <*timeout-value*> **retries** <*count*> **[type traps | informs] notifications** <*notification-list*> **tags** <*tag-list*> **mask** <*mask-string*> **mms** <*mms-value*> **[status enable | disable]**
snmp set target-params <*param-name*> **[v1 | v2c | v3 [auth | noauth | priv]] security-name** <*security-name*> **filter** <*filter-name*>
snmp set trap-source <*IPaddr*>
snmp set user <*username*> **[engine-id** <*id-string*> **| local] [auth md5| sha1] [priv des]**
snmp set user-to-group <*username*> **to** <*groupname*>
snmp set view <*view-name*> **subtree** <*OID*> **mask** <*mask-string*> **[type include| exclude]**
snmp show all| access| chassis-id| tftp| trap| community| statistics| mibs| {engine-id <*IP_address*>| <*port*>**}| statistics| {user** <*user-name*> **[engine-id** <*id-string*>**]| local]| all}| {group** <*group-name*>**| all}| {view** <*view-name*>**| all}| {target-params** <*params-name*>**| all}| {notify** <*notify-name*>**| all}| {filter** <*filter-name*>**| all}**
snmp stop
snmp test trap type ps-failure| ps-recover| vrrpNewMaster| coldStart| linkDown |linkUp

snmp disable trap

Purpose

Disable specific SNMP trap types.

Format

snmp disable trap authentication| link-up-down| frame-relay| ospf| spanning-tree| bgp| vrrp| environmentals

Mode

Configure

Description

The **snmp disable trap** command controls the types of traps the X-Pedition emits based trap type.

Parameters

authentication	Disables authentication traps, which the X-Pedition sends when it receives an invalid SNMP community string.
link-up-down	Disables link-state change traps, which the X-Pedition sends when a port's operational state changes.
frame-relay	DLCI up/down trap.
ospf	Sixteen different OSPF traps.
spanning-tree	NewRoot and topologyChange traps.
bgp	BGPEstablished and bgpBackwardTransistion traps
vrrp	NewMaster and authFailure traps.
environmentals	Temperature, fan, and power supply traps

Restrictions

None.

snmp disable port-trap

Purpose

Disable specific SNMP trap types for a specific port.

Format

snmp disable port-trap *<port list>*

Mode

Configure

Description

The **snmp disable port-trap** command controls the types of traps the X-Pedition emits based trap type on specific ports. You can disable the following trap types on a per-port basis:

- Link-state change – use the **link-up-down** keyword to prevent the X-Pedition from sending a trap each time a port changes operational state.

Parameters

<port list> Specifies the port(s) on which you wish to disable traps.

Restrictions

None.

snmp set chassis-id

Purpose

Set the X-Pedition's chassis ID using SNMP.

Format

snmp set chassis-id <*chassis-name*>

Mode

Configure

Description

The **snmp set chassis-id** command lets you set a string to give the X-Pedition an SNMP identity.

Parameters

<*chassis-name*> Is a string describing the X-Pedition.

Restrictions

None.

snmp set community

Purpose

Set an SNMP community string and specify the access privileges for that string.

Format

snmp set community <*community*> **name** <*community-name*> **security-name** <*security-name*> **engine-id** <*engine-id-string*> **tag** <*tag-string*>

Mode

Configure

Description

The **snmp set community** command sets a community string for SNMP access to the X-Pedition. SNMP management stations that want to access the X-Pedition must supply a community string that is set on the switch. This command also sets the level of access to the X-Pedition to read-only or read-write. Communities that are read-only allow SNMP *GETs* but not SNMP *SETs*. Communities that have read-write access allow both SNMP GETs and SNMP SETs.

Parameters

community <*community*>
Character string for the community string.

name <*community-name*>
Enter a community name for this community entry. If not specified, it is the same as <*community*>.

security-name <*security-name*>
The security name used to map to an access group, which specifies the access rights. This is the security name used in the **snmp set community-to-group** command. If not specified, it is the same as the **<name>** value or **<community>** if name is not specified.

engine-id <*engine-id-string*>
Identifies the SNMP entity for which this community is used. If not present, the local engine ID is used.

tag <*tag-string*>
A tag value for this community entry. If specified, it limits community authentication to messages received from NMS entities whose transport address matches **snmp set target** commands with matching tag in the tag list. See *snmp set target* on page 1069.

Restrictions

None.

Example

To set the SNMP community string to “public,” which has read-only access:

```
xp(config)# snmp set community public privilege read
```

snmp set community-to-group

Purpose

Assigns an SNMP community to an SNMP group.

Format

snmp set community-to-group <*security-name*> **to** <*group-name*> **[v1| v2c]**

Mode

Configure

Description

The **snmp set community-to-group** command allows you to assign an SNMP community to an SNMP group. Related commands include **snmp show**, **snmp set group**, and **snmp set community**.

Parameters

community-to-group <*security-name*>
The security name of the community to assign. See *snmp set community* on page 1054.

to <*group-name*>
Specify the name of the group to which you will assign the community. The group name is specified with the **snmp set group** command. See *snmp set group* on page 1059.

v1 | v2c
Specifies the security model for this community-to-group assignment.

snmp set filter

Purpose

Configures notification filters to include or exclude certain notifications.

Format

snmp set filter <*filter-name*> **subtree** <*OID*> **mask** <*mask-string*> **[category bgp| dot1dbridge| ds3| frame-relay-dte| interfaces| ipx| ospf| rmon| snmp| xp-enterprise| vrrp] [type included| excluded]**

Mode

Configure

Description

The **snmp set filter** command configures notification filters to include or exclude certain notifications.

Also see *snmp show* on page 1079 and *snmp set target-params* on page 1072.

Parameters

filter <*filter-name*>
The name of the notification filter.

subtree <*OID*>
A string containing the OID value of the subtree.

mask <*mask-string*>
A bit string, represented by a hexadecimal representation of 1 to 16 characters, used to specify a mask that modifies which parts of the specified subtree are selected.

category
May be used instead of the subtree and mask options to select notifications by feature name. You may enter one of the following keywords:

bgp	Border Gateway Protocol notifications
dot1dbridge	Dot1d bridging notifications
ds3	DS3/E3 WAN Interface notifications
frame-relay-dte	Frame Relay DTE notifications.
interfaces	Generic interface notifications.
ipx	Novell IPX notifications.
ospf	Open Shortest Path First routing protocol notifications
rmon	RMON alarm notifications.
snmp	SNMP Authentication notifications.

xp-enterprise	X-Pedition-specific enterprise notifications.
vrrp	Virtual Router Redundancy Protocol notifications.

type

Specifies whether or not the selected subtree is included in the filter.

included	The selected subtree is included in the filter (i.e. notifications that are part of the subtree will be filtered out).
excluded	The selected subtree is excluded from the filter (i.e. only notifications that are part of the subtree will be sent).

Restrictions

None.

snmp set group

Purpose

To configure a new SNMP group or table that maps SNMP users to SNMP views.

Format

snmp set group <*group-name*> **[v1 | v2c | v3 [auth | noauth | priv]] read** <*readview*> **write** <*writeview*> **notify** <*notifyview*>

Mode

Configure

Description

The **snmp set group** command allows you to configure a new SNMP group or table that maps SNMP users to SNMP views. When configuring a group, you may specify only one security model per command. To configure a second set of views for the same group using a second security model, add another snmp set group command.

When a community string is configured internally, two groups with the name of the defined community string are automatically generated, one for the v1 security model and the other for the v2c security model. Similarly, deleting a community string will delete a v1 group with the name of the defined community string and a v2c group with the name of the community string.

Also see *snmp show* on page 1079 and *snmp set user* on page 1075.

Parameters

group <*group-name*>
The name of the group

v1
Members of this group authenticated using the SNMPv1 security model will use the read, write, and notify views specified.

v2c
Members of this group authenticated using the SNMPv2c security model will use the read, write, and notify views specified.

v3
Members of this group authenticated using the SNMPv3 security model (USM) will use the read, write, and notify views specified.

auth
Members of this group authenticated using one of the secure authentication security levels will use the read, write, and notify views specified.

noauth
Members of this group authenticated using the plaintext username authentication security level will use the read, write, and notify views specified.

priv
Members of this group authenticated using both secure authentication and privacy will use the read, write, and notify views specified.

read <*readview*>
This option allows you to specify a read view. Enter the name (a string up to 64 characters) of the view that enables you only to view the contents of the agent. By default, this value is the null OID

write <*writeview*>
This option allows you to specify a write view. Enter the name (a string up to 64 characters) of the view that enables you to enter data and configure the contents of the agent. By default, this value is the null OID. Write access must be configured explicitly.

notify <*notifyview*>
This option allows you to specify a notify view. Enter the name (a string up to 64 characters) of the view that enables you to specify a notify, inform, or trap. By default, this value is the null OID. Notify access must be configured explicitly.

Restrictions

None.

snmp set if-alias

Purpose

Used to assign additional identification information to any interface handled by the ifXTable (i.e., physical ports, IP interfaces, IPX interfaces, VLANs, and SmartTRUNKs).

Format

snmp set if-alias <*interface-name*> **alias** <*alias-name*>

Mode

Configure

Description

Used primarily for administrative purposes, the ifAlias SNMP object allows you to assign additional identification information to any interface handled by the ifXTable (i.e., physical ports, IP interfaces, IPX interfaces, VLANs, and SmartTRUNKs). The X-Pedition allows one alias assignment per interface and limits each alias to a maximum length of 64 characters. You may use a remote SNMP manager to view, add, change, or delete an alias.

To assign an alias, enter the following from configuration mode:

```
router(config)# snmp set if-alias <interface-name> alias <alias-name>
```

Note: You cannot remove an interface until you remove any alias assigned to the interface.

Parameters

<*interface-name*>
The name of an existing interface (i.e., physical port, IP interface, IPX interface, VLAN, or SmartTRUNK).

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

<*alias-name*>
The name (up to 64 characters) of the alias you will assign to the interface.

Restrictions

None.

Examples

The following example demonstrates how to assign the alias “spare-port” to the physical port et.4.8.

```
router(config)# snmp set if-alias et.4.8 alias spare-port
```

The example below illustrates how to assign the alias “NEWALIAS” to the newly created “NEWIF” IP interface.

```
router(config)# interface create ip NEWIF address-netmask 10.0.0.1/8 port et.1.1
router(config)# snmp set if-alias NEWIF alias NEWALIAS
```

snmp set mib

Purpose

Enables or disables a given MIB module in the SNMP agent.

Format

snmp set mib name <*mib-name*> **status enable| disable**

Mode

Configure

Description

The **snmp set mib** command allows you to enable or disable a particular MIB module in the SNMP agent.

Parameters

<*mib-name*>

Character string for the MIB module you wish to enable or disable. The following MIB modules are supported by the SNMP agent:

LAG-MIB	IEEE 802.3ad LACP
SNMPv2-MIB	System and snmp group objects
EtherLike-MIB	IEEE 802.3 detailed ethernet statistics
IF-MIB	Interfaces group: ifTable, ifXTable, ifStackTable
IP-MIB	IP group containing global IP statistics
IP-FORWARD-MIB	IP CIDR Route Table
UDP-MIB	UDP statistics group
TCP-MIB	TCP Statistics group
BGP4-MIB	Border Gateway Protocol Version 4 mib
OSPF-MIB	OSPF Version 2 mib
RIPv2-MIB	RIP Version 2 mib
BRIDGE-MIB	Transparent layer 2 bridging protocol mib
FRAME-RELAY-DTE-MIB	Frame Relay mib

PPP-LCP-MIB	Point to Point Link Control Protocol mib
PPP-IP-NCP-MIB	Point to Point IP Network Control Protocol
PPP-BRIDGE-NCP-MIB	Point to Point Bridge Control Protocol
DS1-MIB	Transmission statistics for DS1 serial line protocol
DS3-MIB	Transmission statistics for DS3 serial line protocol
SONET-MIB	Transmission statistics for SONET
ATM-MIB	Transmission statistics for ATM
RADIUS-AUTH-CLIENT-MIB	Radius client protocol statistics
RMON-MIB	Remote Monitoring for Layer 2 traffic
RMON2-MIB	Remote Monitoring for Layer 3/4 traffic
VRRP-MIB	Virtual Router Redundancy Protocol
DVMRP-MIB	Distance Vector Multicast Routing Protocol
IGMP-MIB	Internet Group Membership Protocol MIB, Multicast
MAU-MIB	IEE 802.3 Medium Attachment Units (MIB)
APPLETALK-MIB	AppleTalk MIB II
FDDI-MIB	FDDI MIB
DEC-ELAN-EXT-MIB	DEC FDDI Extensions MIB
NOVELL-RIPSAP-MB	Novell RIPSAP MIB
NOVELL-IPX-MIB	Novell IPX MIB
CDP-MIB	Cabletron Discovery Protocol
POLICY-MIB	Policy Configuration MIB
CONFIG-MIB	Configuration control mib
HARDWARE-MIB	Chassis, environmental and inventory statistics
SERVICE-STATUS-MIB	Operational protocol statistics
CAPACITY-MIB	Device capacity usage statistics
CTRON-MIB2-MIB	Cabletron extension to MIB-II
CTRON-CONTAINER-MIB	Cabletron container MIB
CTRON-CHASSIS-MIB	Cabletron chassis MIB (6SSRM-02 only)
CT-DOWNLOAD-MIB	Cabletron download MIB

status

Specifies whether to enable or disable the MIB module:

enable	Enables the MIB module
disable	Disables the MIB module

Restrictions

None.

Example

To enable the RMON2-MIB in the SNMP agent:

```
xp(config)# snmp set mib rmon2-mib status enable
```

snmp set notify

Purpose

The **snmp set notify** command allows you to associate a specific notification type with one or more targets.

Format

snmp set notify <*notify-name*> **tag** <*tag-string*> **[type trap | inform]**

Mode

Configure

Description

SNMP notifications can be sent as traps or informs. Traps are unreliable because the receiver does not send acknowledgments when it receives traps. The sender cannot determine if the traps were received. However, an SNMP entity that receives an inform request acknowledges the message with an SNMP response PDU. If the sender never receives the response, the inform request can be sent again. Thus, informs are more likely to reach their intended destination.

However, informs consume more resources in the agent and in the network. Unlike a trap, which is discarded as soon as it is sent, an inform request must be held in memory until a response is received or the request times out. Also, traps are sent only once, while an inform may be retried several times. The retries increase traffic and contribute to a higher overhead on the network.

Also see *snmp show* on page 1079 and *snmp set target* on page 1069.

Parameters

notify <*notify-name*>
The name of the notification.

tag <*tag-string*>
A string containing the tag value for this notification. Identifies the targets used when sending the notification.

[type trap | inform]
Specifies the type of notification.

Trap	Send traps.
Inform	Send informs.

Restrictions

None.

snmp set retro-mib-ifspeed

Purpose

Causes the ifSpeed for IP Interface rows to return speed of the first operational port. The default reported value is zero. This allows the ifSpeed to behave as it would in earlier versions of the MIB.

Format

snmp set retro-mib-ifspeed

Mode

Configure

Description

The **snmp set retro-mib-ifspeed** command causes the ifSpeed for IP Interface rows to return the speed of the first operational port.

Parameters

None.

Restrictions

None.

snmp set target

Purpose

Configures the recipient of an SNMP trap operation.

Format

snmp set target [<*target-name*> | <*ip-address*>**] ip-address** <*ip-address*>
param <*param-name*> **owner** <*owner-name*> **[v1 | v2c | v3 [auth | noauth | priv]]**
[community <*community-name*> | **security** <*security-name*>**] port** <*udp-port*>
timeout <*timeout-value*> **retries** <*count*> **[type traps | informs] notifications** <*notification-list*>
tags <*tag-list*> **mask** <*mask-string*> **mms** <*mms-value*> **[status enable | disable]**

Mode

Configure

Description

The **snmp set target** command configures the recipient of an SNMP trap operation (i.e., the IP address of the target server(s) to which you want the X-Pedition to send SNMP traps). You may configure multiple targets with the same IP address and different security strings, but traps will not be sent to a target if the security string specified on the target is not configured. Trap targets are **enabled** by default but you can use the status argument to disable or re-enable a target. Also see *snmp show* on page 1079, *snmp set target-params* on page 1072, and *snmp set notify* on page 1066.

Notes:

- If you do not enter an snmp set target command, no notifications are sent. In order to configure the X-Pedition to send SNMP notifications, you must enter at least one snmp set target command.
- In order to enable multiple targets, you must issue a separate snmp set target command for each target.
- If target-name is not explicitly specified, then the agent will generate a unique name that will be automatically assigned to the target.
- If no security model is specified, either explicitly or through an associated target-parameters entry, SNMPv1 will be used. If no community string is specified, explicitly or through an associated target-parameters entry, “public” will be used.
- If the param option is specified, then the security level and community string are not used. Similarly, if the type option is specified, then the notification list is not used.

Parameters

target [<*target-name*> |<*ip-address*>**]**
String containing the target name.

ip-address <*IP-address*>
The IP Address of the target system. By default, **192.168.1.1**.

Note: The target IP address should be locally attached to the X-Pedition.

param <*param-name*>
String containing the name of the associated target parameters. This can be used to link this target with an snmp set target-params command as a convenient way of grouping multiple snmp set target commands with the same set of parameters used for creating trap or inform messages. The security model, security level, and community are defined by the snmp set target-params command. By default, *param-name* is the same as ***target-name***.

owner <*owner-name*>
The name (a string value) of the creator/owner of the target. By default, the owner-string value is "**monitor**."

[v1 |v2c| v3 [auth| noauth| priv]]

v1 Specifies that the SNMPv1 security model should be used when communicating with this target.

v2c Specifies that the SNMPv2c security model should be used when communicating with this target.

v3 Specifies that the SNMPv3 security model (USM) should be used when communicating with this target.

auth One of the secure authentication protocols will be used when v3 is specified.

noauth Plaintext username authentication will be used when v3 is specified.

priv Encryption will be used, in addition to one of the secure authentication protocols, when communicating with this target.

[community <*community-name*> **security-name** <*security-name*>**]**
The security name to use to authenticate with the target. This can be an SNMP community string or an SNMP user name.

port <*udp-port*>
The destination UDP port on the target to which to send the trap or informs. The default udp port is **162**.

timeout <*time-value*>
The inform response timeout value (in hundredths of seconds). By default, *time-value* = **1500**.

retries <*count*>
The number of retries when there is no response to an inform. The default value is **3**.

[type traps| informs]
The type of notification to send. (not used when *notifications* is specified).

Traps Send traps.

Informs Send informs.

notifications <*notification-list*>
A list of notify tag-string values used to associate a notify entry or a group of notify entries with this target (do not use with the *type* option). By default, this value is **defaultTrap**.

tags <*tag-list*>
The list of tags associated with notifications or communities which reference this target. When specifying multiple tags, use quotes around the tag list and separate each tag name with a space. Use this parameter in place of the “notifications” option.

mask <*mask-string*>
Mask to be used when this target is selected by an SNMPv1 or SNMPv2c community tag. Specify using hexadecimal representation. Contains one bit for each bit in the ip-address and port values (4 bytes of ip-address : 2 bytes for port value).

Example: 0xff:ff:ff:00:ff:ff

For each bit set to one, the target address/port must match the from transport address/port on an incoming message. For each bit set to zero, no match is required. This parameter allows one target to specify multiple transport addresses. If a mask is not specified, a value of all ones is used.

mms <*mms-value*>
The maximum size PDU that can be received from or sent to this target. Enter a value equal to or greater than 484. If not specified, 2048 is used.

status enable | disable
Target status.

disabled Target currently not available for use.

enabled Target enabled.

Restrictions

None.

snmp set target-params

Purpose

The **snmp set target-params** command allows you to configure PDU generation parameters for use by the **snmp set target** command.

Format

snmp set target-params <*param-name*> **[v1 | v2c | v3 [auth | noauth | priv]]**
security-name <*security-name*> **filter** <*filter-name*>

Mode

Configure

Description

Use the **snmp set target-params** configuration command to group multiple targets with the same trap/inform message generation parameters to one configuration statement. The **snmp set target** command need not specify the parameters in the **snmp set target-params** configuration command. Instead, specify the desired parameters using the param option to reference an **snmp set target-params** configuration command. Also see *snmp show* on page 1079, *snmp set target* on page 1069, and *snmp set filter* on page 1057.

Parameters

target-params <*param-name*>
The name of the target parameters.

v1
Specifies that targets associated with this target-params entry will use the SNMPv1 security model.

v2c
Specifies that targets associated with this target-params entry will use the SNMPv2c security model.

v3
Specifies that targets associated with this target-params entry will use the SNMPv3 security model (USM).

auth For SNMPv3, specifies that one of the secure authentication protocols is to be used.

noauth For SNMPv3, specifies that plaintext username authentication is to be used.

priv For SNMPv3, specifies that encryption is to be used in addition to one of the secure authentication protocols.

security-name <*security-name*>
The security name to use to authenticate with the target. This can be an SNMP community string or an SNMP user name.

filter <*filter-name*>
The name of the notification filter to apply when sending notifications to associated targets.

Restrictions

None.

snmp set trap-source

Purpose

Sets the source interface IP address reported in traps sent by the SNMP Agent.

Format

snmp set trap-source <*IPaddr*>

Mode

Configure

Description

The **snmp set trap-source** command configures the IP address reported in traps sent by the SNMP Agent.

Parameters

<*IPaddr*>	The IP address.

Restrictions

None.

snmp set user

Purpose

To configure a new SNMP user.

Format

snmp set user <*username*> **[engine-id** <*id-string*> **| local] [auth md5| sha1] [priv des]**

Mode

Configure

Description

The **snmp set user** command allows you to configure a new SNMP user. When creating a user, use the engine-id option to localize the user for use with a specific SNMP engine and to specify any additional authentication or encryption options to be used with that SNMP engine. Once the user has been created, use the snmp **set user-to-group** command to assign the user to a group.

Related commands include *snmp show* on page 1079, *snmp set user-to-group* on page 1077, and *snmp set group* on page 1059.

Note: When you use the **comment out** command to disable an **snmp set user** command, the X-Pedition will delete the passwords for that user. If you attempt to reactivate the command through the **comment in** command, the **snmp set user** command will fail and you will need to re-enter the command and create a new account for the user. To disable users and prevent them from accessing the X-Pedition through SNMP, **comment out** the user's corresponding **snmp set user-to-group** command. This will prevent you from having to recreate user accounts.

Parameters

user <*username*>
The name of the user.

engine-id <*id-string*>
Identifies a remote SNMP Engine-ID.

local
Specifies the local agent's SNMP Engine-ID.

auth
Configures the user to use one of the secure authentication protocols.

md5
Specifies use of the HMAC-MD5-96 secure authentication protocol.

sha1
Specifies use of the HMAC-SHA-96 secure authentication protocol.

priv
Configures the user to use encryption for privacy, in addition to one of the secure authentication protocols.

des Specifies use of the CBC-DES encryption algorithm. This is the default value.

Restrictions

None.

snmp set user-to-group

Purpose

Assigns an SNMP user to an SNMP group.

Format

snmp set user-to-group <*username*> **to** <*groupname*>

Mode

Configure

Description

The **snmp set user-to-group** command allows you to assign an SNMP user to an SNMP group. Related Commands include *snmp show* on page 1079, *snmp set user* on page 1075, and *snmp set group* on page 1059.

Parameters

user-to-group <*username*>
The name of the user

to <*groupname*>
The name of the group to which you will assign the user.

Restrictions

None.

snmp set view

Purpose

Configure a new view.

Format

snmp set view <*view-name*> **subtree** <*OID*> **mask** <*mask-string*> **[type include| exclude]**

Mode

Configure

Description

The **snmp set view** command allows you to configure a view that may be referenced by a group. Also see *snmp show* on page 1079 and *snmp set group* on page 1059.

Parameters

snmp set view <*view-name*>
The name of the view.

subtree <*OID*>
A string specifying the OID of the subtree.

mask <*mask-string*>
Specifies a mask that modifies which parts of the specified subtree are selected. A bit string containing the mask is a hexadecimal representation of 1 to 16 characters.

type include| exclude
Specifies whether or not the selected subtree is included in the view.

Include Include the selected subtree in the view.

Exclude Exclude the selected subtree from the view.

Restrictions

None.

snmp show

Purpose

Shows SNMP information.

Format

snmp show all| access| chassis-id| tftp| trap| community| statistics| mibs|
{engine-id <*IP_address*>| <*port*>}| **statistics**| **{user** <*user-name*> **[engine-id** <*id-string*>**| local]|**
all}| **{group** <*group-name*>**| all}**| **{view** <*view-name*>**| all}**| **{target-params** <*params-name*>**|**
all}| **{notify** <*notify-name*>**| all}**| **{filter** <*filter-name*>**| all}**| **{target-addr** <*target-name*>**}**

Mode

Enable

Description

The **snmp show** command allows you to display the following SNMP information:

- Five most recent clients to access the X-Pedition
- SNMP name
- Tftp SNMP status
- Trap target related configuration
- Community strings
- SNMP statistics
- MIB registry
- Local SNMP engine and all remote engines configured on the device
- Agent statistics
- Name of user whose SNMP information is displayed
- Engine ID
- Name of group for which SNMP information is displayed
- VACM views
- Trap and inform target parameters
- Notification entries
- Notification filters
- SNMP target information

Parameters

all Displays all SNMP information (equivalent to specifying all the other keywords).

access Displays the last five SNMP clients to access the X-Pedition.

chassis-id Displays the X-Pedition's SNMP name.

tftp Show tftp SNMP status.

trap Displays the IP address of the trap target server.

community Displays the X-Pedition's community string.

statistics Displays SNMP statistics.

mibs Displays the SNMP MIB registry.

{engine-id <*IP_address***>| <***port***>}**

For information about similar commands, see *snmp set user* on page 1075 and *snmp set target* on page 1069.

<*IP_address***>**

Specify an IP Address if you would like to display the remote engine-id of an SNMP entity at a particular IP address. If the SNMP agent already knows the engine-id of the entity at that address, it will display in the table. If the SNMP agent does not know the engine-id, it will attempt to discover the engine-id by sending a discovery packet to the IP address. If the router receives a valid reply to a discover request, it will add the engine-id to an internal table and the following errorlog message will appear:

%SNMP-I-ENGINE_DSCRVD, SNMP has just discovered an engine-id for: 10.136.136.210 Engine-ID = 0x80:00:07:e5:80:09:86:da:7f:92:1b:58:3d

<*port*>

By default, the X-Pedition monitors engine-ids on UDP trap port 162. Use the port option if the SNMP entity is configured to listen to traps on a different UDP port. This will cause the router to search for the IP Address / Port pair in the internal table or to place the specified port in the discovery packet.

statistics View statistics for the SNMP agent.

user <*user-name*> **[engine-id** <*id-string*> **| local]**

<*user-name*> The name of the user whose SNMP information you will display. Use the keyword **all** to view information about all users.

<*id-string*> The host on which the user is defined. Also see *snmp set user* on page 1075.

group <*group-name*>
The name of the group for which you will view the SNMP information. Use the **all** keyword to view information about all groups. Also see *snmp set group* on page 1059.

view <*view-name*>
Display information about a specific SNMP VACM view. Use the **all** keyword to view information about all views. Also see *snmp set view* on page 1078.

target-params <*params-name*>
Display information about a specific set of SNMP trap and inform target parameters. Use the **all** keyword to view information about all target parameters. Also see *snmp set target-params* on page 1072.

notify <*notify-name*>
Display information about a specific SNMP notification entry. Use the **all** keyword to view information about all notification entries. Also see *snmp set notify* on page 1066.

filter <*filter-name*>
Display information about SNMP notification filters. Use the **all** keyword to view information about all notification filters. Also see *snmp set filter* on page 1057.

target-addr <*target-name*>
Displays information about a specific set of SNMP targets. Use the **all** keyword to display information about all targets.

Restrictions

None.

Examples

The following examples depict sample output of each **show** command option.

Example 1

The following **snmp show access** command displays a log of SNMP access to the X-Pedition. The host that accessed the X-Pedition and the X-Pedition system time when the access occurred are listed.

```
xp(config)# snmp show access
SNMP Last 5 Clients:
        10.15.1.2       Wed Feb 7 18:42:59 2001
        10.15.1.2       Wed Feb 7 18:42:55 2001
        10.15.1.2       Wed Feb 7 18:42:56 2001
        10.15.1.2       Wed Feb 7 18:42:57 2001
        10.15.1.2       Wed Feb 7 18:42:58 2001
```

Example 2

To display the SNMP identity of the X-Pedition:

```
xp(config)# snmp show chassis-id

SNMP Chassis Identity:
s/n 123456
```

Example 3

To display the IP address of the trap target server:

```
xp(config)# snmp show trap

Trap Table:
Index    Trap    Target        Addr    Community String    Status
1.               10.15.1.2             public              enabled
2.               1.2.3.4               public123           disabled
3.               5.6.7.8               public20            disabled
```

Example 4

The following example shows 0x80:00:15:F8:03:00:00:1D:5F:00:1E as the local Engine-ID and 0x12:34:56:78:9A:BC:DE:F0:00:00:00:00 as the remote Engine-ID, 171.69.37.61 as the IP address of the remote engine, or copy of SNMP, and 162 as the port on the remote device to which the local device connects.

```
router# snmp show engine-id

Local SNMP engineID: 0x80:00:15:f8:03:00:00:1D:5F:00:1E
Remote Engine ID                        IP-addr          Port
0x12:34:56:78:9A:BC:DE:F0:00:00:00:00   171.69.37.61     162
```

Field Descriptions

Local SNMP engine ID	A string that identifies the copy of SNMP on the local device.
Remote Engine ID	A string that identifies the copy of SNMP on the remote device.
IP-addr	The IP address of the remote device.
Port	The port number on the remote device.

Example 5

To display the statistics for the SNMP agent, use the **snmp show statistics** command.

```
router# snmp show statistics

SNMP Statistics:
        7 packets received
        7 in get objects processed
        0 in get requests
        0 in get responses
        7 get-next requests
        0 in set requests
        0 in total objects set
        0 bad SNMP versions
        0 bad community names
        0 ASN.1 parse errors
        0 PDUs too big
        0 no such names
        0 bad values
        0 in read onlys
        0 in general errors
        0 silent Drops
        0 packets sent
        0 out get requests
        0 get-next responses
        0 out set requests
        0 response PDUs too big
        0 no such name errors
        0 bad values
        0 general errors
        0 version 1 traps sent
        0 traps in queue
        0 traps dropped due to queue overflow
SNMPv3 Engine Stats:
        20 engine boots
        01:24:11 engine time
SNMPv3 Message Processing Stats:
        0 unknown security models
        0 invalid messages
        0 unknown PDU handlers
        0 unavailable contexts
        0 unknown contexts
User Security Model stats:
        0 unsupported security levels
        0 not in time window
        0 unknown users names
        0 unknown engine ids
        0 wrong digests
        0 decryption errors
```

Field Descriptions

Engine boots	The number of times the SNMP engine has re-initialized since initial configuration.
Engine time	The time since engine boots was last incremented.
Unknown security models	The number of packets dropped because they referenced a security model which is not known or not supported.
Invalid messages	The number of packets dropped because there were invalid or inconsistent components in the packet.

Unknown PDU handlers	The number of packets dropped because no application existed to process the PDU.
Unavailable contexts	The number of packets dropped because the context in the message was unavailable.
Unknown contexts	The number of packets dropped because the context in the message was unknown.
Unsupported security levels	The number of packets dropped because they requested a security level that was unknown or unavailable.
Not in time window	The number of packets dropped because they appeared outside of the authoritative SNMP engine's time window.
Unknown users names	The number of packets dropped because they referenced a unknown user.
Unknown engine ids	The number of packets dropped because they referenced an snmpEngineID that was not known.
Wrong digests	The number of packets dropped because they did not contain the expected digest value.
Decryption errors	The number of packets dropped because they could not be decrypted.

Example 6

To display information on SNMP users in the USM user table, use the **snmp show user** command. The following example shows the username as "authuser," the Engine-ID as 0x00:00:00:09:02:00:00:00:0C:02:58:08, the authentication protocol as sha1, no privacy protocol, and the storage-type as nonVolatile:

```
router# snmp show user authuser

user name:    authuser
engine id:    0x00:00:00:09:02:00:00:00:0C:02:58:08
auth type:    sha1
priv type:    none
storage-type: nonVolatile
```

Field Descriptions

User name	A string identifying the name of the SNMP user.
Engine id	A string identifying the name of the copy of SNMP on the device.
Auth type	The authentication protocol the user is configured to use, either MD5, SHA1, or none.

Priv type	The privacy protocol the user is configured to use, either DES or none.
Storage-type	Indicates whether the settings have been set in volatile or temporary memory on the device, or in non-volatile or persistent memory where settings will remain after the device has been turned off and on again.

Example 7

The **snmp show group** example below shows the group name as public, the security model as v1, the security level as noauth the read view name as v1default, no write view, the notify view name as v1notify, and the storage type as volatile:

```
router# snmp show group public

groupname:    public                  security model:  v3 noAuth
readview:      v1default
writeview:
notifyview:    v1notify
storage-type:  volatile
```

Field Descriptions

Groupname	The name of the SNMP group, or collection of users who have a common access policy.
Security model	The security model used by the group, either v1, v2c, or v3.
Readview	A string identifying the read view of the group.
Writeview	A string identifying the write view of the group.
Notifyview	A string identifying the notify view of the group.
Storage-type	Indicates whether the settings have been set in volatile or temporary memory on the device, or in non-volatile or persistent memory where settings will remain after the device has been turned off and on again.

Example 8

The **snmp show view** example below shows the view name as internet, the subtree as 1.3.6.1, no mask, and the storage type as permanent:

```
router# snmp show view internet

viewname:    internet
subtree:     1.3.6.1
mask:
type:        include
storage-type: permanent
```

Field Descriptions

Viewname	The name of the view.
Subtree	The string indentifying the subtree oid.
Mask	A string identifying the mask associated with the subtree.
Type	The type of view: *included* or *excluded*.
Storage-type	Indicates whether the settings have been set in volatile or temporary memory on the device, or in non-volatile or persistent memory where settings will remain after the device has been turned off and on again. You may modify but not remove permanent entries.

Example 9

The **snmp show target-params** example below shows the target parameters name as publicTargets, the security level as v1, the security name as public, and the storage type as nonVolatile:

```
router# snmp show target-params publicTargets

params name:    publicTargets
security level:   v1
security name:   public
storage-type:     nonVolatile
```

Field Descriptions

Params name	The name of the target parameters.
Security level	The security model and security level specified by the target parameters.
Security name	The security name specified by the target parameters. Depending on the security model this will either be a community string or a username.
Storage-type	Indicates whether the settings have been set in volatile or temporary memory on the device, or in non-volatile or persistent memory where settings will remain after the device has been turned off and on again.

Example 10

The **snmp show notify** example below shows the notification entry name as defaultTrap, the notify tag as defaultTrap, the notify type as trap, and the storage type as permanent:

```
router# snmp show notify defaultTrap

notify name:   defaultTrap
notify tag:    defaultTrap
notify type:   trap
storage-type:  permanent
```

Field Descriptions

Notify name	The name of the notification entry.
Notify tag	The tag used to associate targets with the notification entry.
Notify type	The types of notifications, either traps or informs, to be sent to targets associated with the notification entry.
Storage-type	Indicates whether the settings have been set in volatile or temporary memory on the device, or in non-volatile or persistent memory where settings will remain after the device has been turned off and on again.

Example 11

The **snmp show filter** example below shows the notification filter name as noBGP, the subtree as 1.3.6.1.2.1.15, no mask, the filter type as included, and the storage type as volatile:

```
router# snmp show filter noBGP

filter name:   noBGP
subtree:       1.3.6.1.2.1.15
mask:
filter type:   included
storage-type:  volatile
```

Field Descriptions

Filter name	The name of the notification filter.
Subtree	The OID subtree that is included in, or excluded from, the filter.
Mask	The mask used to modify which parts of the subtree are selected.

Filter type	Shows whether the filter includes or excludes the selected subtree. If the subtree is included in the filter, notifications that are part of the subtree will be filtered out. If the subtree is excluded from the filter, only notifications that are part of the subtree will be sent.
Storage-type	Indicates whether the settings have been set in volatile or temporary memory on the device, or in non-volatile or persistent memory where settings will remain after the device has been turned off and on again.

Example 12

The following example displays information about the SNMP target *target1*.

```
xp# snmp show target-addr target1

addr name:          target1
ip addr:            134.141.136.193/0162
timeout:            1500
retry count:        3
tag-list:           mytag defaultTrap
params:             target1
tmask:              0xff:ff:ff:ff:00:00
max msg size:       2048
storage-type:       volatile


Field Descriptions
-----------------------------------------------------------
addr name      |  The name of the target.
ip addr        |  The IP address and UDP port number of the target.
timeout        |  The timeout value used when retrying informs.
retry count    |  The number of times to retry an inform.
tag-list       |  A list of tags associated with notify or community command tags.
params         |  The associated target-params command.
tmask          |  A mask used to determine which bits of the IP address/port are
               |  used when matching a community tag.
max msg size   |  The maximum allowed message size for this target.
storage-type   |  Indicates whether the settings have be set in volatile or
               |  temporary memory on the device, or in non-volatile or persistent
               |  memory where settings will remain after the device has been
               |  reset or turned off and on again.
```

snmp stop

Purpose

Stop SNMP access to the device.

Format

snmp stop

Mode

Configure

Description

The **snmp stop** command stops SNMP access to the X-Pedition. The X-Pedition will still finish all active requests but will then disregard future requests. When you issue this command, UDP port 161 is closed.

Parameters

None.

Restrictions

None.

snmp test trap

Purpose

Tests SNMPv1 notifications to currently configured managers.

Format

snmp test trap type ps-failure| ps-recover| vrrpNewMaster| coldStart| linkDown |linkUp

Mode

Enable

Description

The **snmp test trap** command allows you to test the SNMPv1 notifications to the managers currently configured.

Parameters

ps-failure	Tests the power supply failure trap notification.
ps-recover	Tests the power supply recover trap notification.
vrrpNewMaster	Tests the Virtual Router Redundancy New Master Trap.
coldStart	Send coldStart trap to manager.
linkDown	Send link down for ifIndex 1 to manager.
linkUp	Send link up for ifIndex 1 to manager.

Restrictions

None.

Chapter 63

sonet Commands

The **sonet** commands allow you to configure and display various parameters for Synchronous Optical Network (SONET) encapsulation. These commands allow you to accommodate Packet-over-SONET (POS) and ATM (asynchronous transfer mode) transmission using the X-Pedition.

Packet-over-SONET technology provides the ability to transmit IP packets and ATM cells over a SONET backbone by encapsulating them into a SONET frame. In reference to the OSI Layer model, the SONET layer rests right beneath the IP layer or the ATM layer. Based on the transmission mechanism of SONET frames, the result is larger traffic bandwidth and faster line speed (OC-3), accommodating QoS guarantees and the ability to deliver voice/video data over an internetwork.

SONET frames carry a large amount of data stored as overhead. This overhead information provide the information for OAM&P (operation, administration, management, and provisioning) capabilities, such as performance monitoring, automatic protection switching, and path tracing.

Enterasys' SONET technology features Automatic Protection Switching, performance monitoring capabilities, as well as commercial circuit identification.

Command Summary

Table 49 lists the **sonet** commands. The sections following the table describe the command syntax.

Notes:

- The X-Pedition does *not* support PVST over POS. However, the router *will* support STP over POS.
- A hardware limitation allows FDDI and SONET modules to increment only the *ifInUcastPkts* and *ifOutUcastPkts* ifMib counters. Non-unicast packet counters (i.e., *ifInNUcastPkts* and *ifOutNUcastPkts*) do not increment and will remain 0.

  On gigabit and 10-Gigabit modules, all OCMAC counters increment correctly.

Table 49. sonet commands

sonet set <*SONET ports*> **C2** <*num*>
sonet set <*SONET ports*> **circuit-id** <*string*>
sonet set <*SONET ports*> **fcs-16-bit**
sonet set <*SONET ports*> **framing sonet| sdh**
sonet set <*SONET ports*> **J0** <*num*>
sonet set <*SONET ports*> **loopback none| line-facility| serial-terminal| parallel**
sonet set <*SONET ports*> **path-trace** <*string*>
sonet set <*SONET ports*> **payload-scramble on| off**
sonet set <*SONET ports*> **protected-by** <*SONET port*>
sonet set <*SONET ports*> **protection 1+1**
sonet set <*SONET ports*> **protection-switch lockoutprot| forced| manual**
sonet set <*SONET ports*> **revertive off|on**
sonet set <*SONET ports*> **S1S0** <*num*>
sonet set <*SONET ports*> **sd-ber** <*num*>
sonet set <*SONET ports*> **sf-ber** <*num*>
sonet set <*SONET ports*> **sts-stream-scramble on|off**
sonet set <*SONET ports*> **WTR-timer** <*num*>
sonet show aps <*SONET ports*>

Table 49. sonet commands

sonet show loopback <*SONET ports*>
sonet show medium <*SONET ports*>
sonet show pathtrace <*SONET ports*>

sonet set C2

Purpose

Sets a value for the C2 flag.

Format

sonet set <*SONET ports*> **C2** <*num*>

Mode

Configure

Description

The **sonet set C2** command allows you to specify a value for the C2 flag. SONET frames carry overhead for path, section and line for easier multiplexing and better OAM&P (operation, administration, management, and provisioning) capabilities.

The SONET frame overhead information is stored in separate bytes, or flags.

There are nine bytes allocated for section overhead labeled A1, A2, B1, D1, D2, D3, E1, F1, J0/Z0.

There are 18 bytes allocated for line overhead labeled H1, H2, H3, B2, K1, K2, D4, D5, D6, D7, D8, D9, D10, D11, D12, S1/Z1, M0/M1, and E2.

There are nine bytes allocated for path overhead labeled J1, B3, C2, H4, G1, P2, Z3, Z4, and Z5.

The **sonet set C2** command set the C2 flag. The C2 flag is the path signal label byte used to indicate the contents of the synchronous payload envelope.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).
<*num*>	Specifies the value of the C2 flag. Specify any number between 0 and 255.

Restrictions

None.

Example

To set the C2 flag to 16 on port so.2.1:

```
xp(config)# sonet set so.2.1 C2 16
```

sonet set circuit-id

Purpose

Sets a circuit identifier.

Format

sonet set <*SONET ports*> **circuit-id** <*string*>

Mode

Configure

Description

The **sonet set circuit-id** command allows you to set a circuit identifier on a specified SONET port. This command is for administrative purposes, used to identify this line and associate it with a certain customer circuit. Primarily used for service level management.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).
<*string*>	Specifies the circuit identifier. The maximum length must be 64 bytes or less.

Restrictions

None.

Example

To identify the circuit line on port so.2.1 to as ‘customer1’:

```
xp(config)# sonet set so.2.1 circuit-id customer1
```

sonet set fcs-16-bit

Purpose

Sets the frame check sequence to 16 bits.

Format

sonet set <*SONET ports*> **fcs-16-bit**

Mode

Configure

Description

The **sonet set fcs-16-bit** command allows you to set the frame check sequence (FCS) field length of the SONET frame to 16 bits. By default, this field length is set to 32 bits (4 octets). Using this command, you can set the frame check sequence field length to 16 bits (2 octets) instead.

The FCS field is used as an error check mechanism during frame transmission. An FCS value is calculated before transmission based on destination address, source address, and other data inside the frame. The FCS field inside the SONET frame carries this value. After the frame arrives to the destination, the FCS value is calculated again and compared with the value in the FCS field. This is done to ensure that there was no errors during transmission.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).

Restrictions

None.

Example

To set the frame check sequence on port so.2.1 to 16 bits:

```
xp(config)# sonet set so.2.1 fcs-16-bit
```

sonet set framing

Purpose

Sets optical framing for SONET or SDH.

Format

sonet set <*SONET ports*> **framing sonet| sdh**

Mode

Configure

Description

The **sonet set framing** command allows you to specify the SONET frame type for mapping the data. The two options are SONET (Synchronous Optical Network) which is an ANSI standard, or SDH (Synchronous Digital Hierarchy) which is an ITU standard.

There are minor differences between the two standards. One such difference is that SONET has a basic transmission rate of OC-1 (51.84 Mbps), whereas SDH has a basic transmission rate of OC-3 (155.52 Mbps).

Parameters

<*SONET ports*>	Specifies the SONET port name(s).
sonet	Sets the optical framing standard to SONET.
sdh	Sets the optical framing standard to SDH.

Restrictions

None.

Example

To set optical framing on port so.2.1 to SONET:

```
xp(config)# sonet set so.2.1 framing sonet
```

sonet set J0

Purpose

Sets a value for the J0 flag.

Format

sonet set <*SONET ports*> **J0**<*num*>

Mode

Configure

Description

The **sonet set J0** command allows you to specify a value for the J0 flag. SONET frames carry overhead for path, section and line for easier multiplexing and better OAM&P (operation, administration, management, and provisioning) capabilities.

The SONET frame overhead information is stored in separate bytes, or flags.

There are nine bytes allocated for section overhead labeled A1, A2, B1, D1, D2, D3, E1, F1, J0/Z0.

There are 18 bytes allocated for line overhead labeled H1, H2, H3, B2, K1, K2, D4, D5, D6, D7, D8, D9, D10, D11, D12, S1/Z1, M0/M1, and E2.

There are nine bytes allocated for path overhead labeled J1, B3, C2, H4, G1, P2, Z3, Z4, and Z5.

The **sonet set J0** command set the J0 flag. The J0 flag is the section trace byte.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).
<*num*>	Specifies the value of the J0 flag. Specify any number between 0 and 255.

Restrictions

None.

Example

To set the J0 flag to 16 on port so.2.1:

```
xp(config)# sonet set so.2.1 J0 16
```

sonet set loopback

Purpose

Exercises loopback functionality.

Format

sonet set <*SONET ports*> **loopback none| line-facility| serial-terminal| parallel**

Mode

Configure

Description

The **sonet set loopback** command allows you to exercise loopback functionality on a specified SONET port. Loopback is used to verify connectivity between two devices.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).
none	Disables loopback functionality. Loopback is disabled by default.
line-facility	Line or facility loopback connects high speed receive data to high speed transmit data.
serial-terminal	Serial or terminal loopback connects high speed transmit to high speed receive data.
parallel	Parallel loopback connects byte wide transmit to receive processor.

Restrictions

None.

Example

To connect high speed receive data to high speed transmit on port so.2.1:

```
xp(config)# sonet set so.2.1 loopback line-facility
```

sonet set path-trace

Purpose

Sets a path trace message.

Format

sonet set <*SONET ports*> **path-trace** <*string*>

Mode

Configure

Description

The **sonet set path-trace** command allows you to set a message in a buffer to be sent as a path-trace message.

The path trace message is part of the path overhead of the transport overhead in every SONET frame. This path trace byte is a 64 byte (or less) message string that is used by the destination and source to notify each other that they are connected within a path.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).
<*string*>	Specifies a character string. The maximum length must be 64 bytes or less.

Restrictions

None.

Example

To send the path trace message ‘tracer’ on port so.2.1:

```
xp(config)# sonet set so.2.1 path-trace tracer
```

sonet set payload-scramble

Purpose

Enables scrambling and descrambling of the STS (synchronous transfer signal) payload.

Format

sonet set <*SONET ports*> **payload-scramble on| off**

Mode

Configure

Description

The **sonet set payload-scramble** command allows you to enable scrambling or descrambling of the payload encapsulated in the STS frame. Scrambling the STS payload is important in optimizing the transmission density of the data stream. Since all SONET transmission use the same source clock for timing, scrambling the payload using a random number generator converts the data stream to a more random sequence. This ensures optimal transmission density of the data stream.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).
on	Enables scrambling and descrambling of the STS payload.
off	Disables scrambling and descrambling of the STS payload.

Restrictions

None.

Example

To enable scrambling on port so.2.1:

```
xp(config)# sonet set so.2.1 payload-scramble on
```

sonet set protected-by

Purpose

Configures an APS protecting port.

Format

sonet set <*SONET ports*> **protected-by** <*SONET port*>

Mode

Configure

Description

The **sonet set protecting** command allows you to specify a protecting port for Automatic Protection Switching (APS). APS is used to provide redundancy for transmission between two SONET devices. This ensures that if a link goes down, traffic can be automatically switched to a secondary backup link and the connection remains operational.

With APS, there is a **working** (primary) port and a **protecting** (backup) port. APS automatically switches all traffic over from the **working** to the **protecting** port in case of signal degradation or failure in receive on the working port.

This command is used in conjunction with the **sonet set protection** command.

Parameters

<*SONET ports*> Specifies the SONET port name(s).

protected-by <*SONET port*>
Specifies the APS protecting port. This must be a single port. Only valid for Packet-over-SONET ports.

Restrictions

None.

Example

To set so.1.1 as the APS protecting port for so.2.1:

```
xp(config)# sonet set so.2.1 protection 1+1 protected by so.1.1
```

sonet set protection

Purpose

Configures an APS working port.

Format

sonet set <*SONET ports*> **protection 1+1**

Mode

Configure

Description

The **sonet set protection 1+1** command allows you to configure a working port for Automatic Protection Switching (APS). This working port will be protected by the protecting port. APS is used to provide redundancy for transmission between two SONET devices. This ensures that if a link goes down, traffic can be automatically switched to a secondary backup link and the connection remains operational.

With APS, there is a **working** (primary) port and a **protecting** (backup) port. APS automatically switches all traffic over from the **working** to the **protecting** port in case of signal degradation or failure in receive on the working port.

This command is used in conjunction with the sonet set **protected-by** option.

Negate this command to disable APS on the SONET port.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).
1+1	Specifies the 1+1 APS scheme, where one working port is matched with one protecting port.

Restrictions

None.

Example

To configure so.2.1 as an APS working port protected by so.1.1:

```
xp(config)# sonet set so.2.1 protection 1+1 protected-by so.1.1
```

sonet set protection-switch

Purpose

Configures protection switching parameters.

Format

sonet set <*SONET ports*> **protection-switch lockoutprot| forced| manual**

Mode

Configure

Description

The **sonet set protection-switch** command allows you to configure SONET Automatic Protection Switching (APS) on a SONET port. With APS, there is a **working** (primary) port and a **protecting** (backup) port. APS automatically switches all traffic over from the **working** to the **protecting** port in case of signal degradation or failure in receive on the working port.

Use this command to configure the APS switching characteristics for a SONET port.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).
lockoutprot	Prevents APS switching from a **working** port to a **protecting** port in the case of signal failure or signal degrade. This command is configured only on the protecting port.
forced	Allows protection switching to occur. Switches service from this port to the other port, even when there are errors on the other port.
manual	Allows you to manually switch service from a port to the other APS port. This is provided that there are no errors on the protecting port. This command can be configured on either the working port or the protecting port.

Restrictions

None.

Example

To configure APS switching for the working port so.2.1:

```
xp(config)# sonet set so.2.1 protection-switch forced
```

sonet set revertive

Purpose

Sets SONET protection switching to revertive or nonrevertive mode.

Format

sonet set <*SONET ports*> **revertive off|on**

Mode

Configure

Description

The **sonet set revertive** command allows you to select whether traffic will be switched back to the working port from the protecting port after the signal degrade or failure condition has been corrected. Once the condition has been corrected, APS waits for a specified time period (WTR-timer) before switching back to the working port.

With APS, there is a **working** (primary) port and a **protecting** (backup) port. APS automatically switches all traffic over from the **working** to the **protecting** port in case of signal degradation or failure in receive on the working port.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).
off	Prevents automatic switch back to the **working** port from the **protecting** port after the signal degrade or failure condition has been corrected.
on	Allows traffic to switch back from the **protecting** port to the **working** port after the signal degrade or failure condition has been corrected and after the Wait-to-Restore timer has expired.

Restrictions

None.

Example

To set APS switching to revertive mode for the protecting port so.2.1:

```
xp(config)# sonet set so.2.1 revertive on
```

sonet set S1S0

Purpose

Sets a value for the S1/S0 flag.

Format

sonet set <*SONET ports*> **S1S0** <*num*>

Mode

Configure

Description

The **sonet set S1S0** command allows you to specify a value for the S1/S0 flag. SONET frames carry overhead for path, section and line for easier multiplexing and better OAM&P (operation, administration, management, and provisioning) capabilities.

The SONET frame overhead information is stored in separate bytes, or flags.

There are nine bytes allocated for section overhead labeled A1, A2, B1, D1, D2, D3, E1, F1, J0/Z0.

There are 18 bytes allocated for line overhead labeled H1, H2, H3, B2, K1, K2, D4, D5, D6, D7, D8, D9, D10, D11, D12, S1/Z1, M0/M1, and E2.

There are nine bytes allocated for path overhead labeled J1, B3, C2, H4, G1, P2, Z3, Z4, and Z5.

The **sonet set S1S0** command set the S1/S0 flag. The S1/S0 flag is the line synchronization status byte used to indicate synchronization state of the line terminating devices.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).
<*num*>	Specifies the value of the S1/S0 flag. Specify any number between 0 and 3.

Restrictions

None.

Example

To set the S1/S0 flag to 1 on port so.2.1:

```
xp(config)# sonet set so.2.1 S1S0 1
```

sonet set sd-ber

Purpose

Sets the Bit Error Rate (BER) signal degrade threshold level.

Format

sonet set <*SONET ports*> **sd-ber** <*num*>

Mode

Configure

Description

The **sonet set sd-ber** command allows you to specify a signal degrade threshold level. There are two threshold levels based on the Bit Error Rate: signal degrade and signal failure. These two threshold levels act as a two stage alarm system, where the signal degrade threshold is always met first before the signal failure threshold.

Once the BER reaches the signal degrade threshold level, then a signal degrade alarm occurs and the receive is considered to be in signal degrade condition. Based upon the APS configuration, all traffic is switched from the working port to the protecting port.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).
<*num*>	Specifies the Bit Error Rate signal degrade threshold level in 10^{-n}. Specify any number for n between 5 to 9. The default is 6, indicating a threshold level of 10^{-6}. This means that a signal degrade alarm occurs if the Bit Error Rate rises past the 1/1000000 level.

Restrictions

None.

Example

To set the BER signal degrade threshold level to 10^{-6} or 1/1000000:

```
xp(config)# sonet set so.2.1 sd-ber 6
```

sonet set sf-ber

Purpose

Sets the Bit Error Rate (BER) signal failure threshold level.

Format

sonet set <*SONET ports*> **sf-ber** <*num*>

Mode

Configure

Description

The **sonet set sf-ber** command allows you to specify a signal failure threshold level. There are two threshold levels based on the Bit Error Rate: signal degrade and signal failure. These two threshold levels act as a two-stage alarm system, whereby the signal degrade threshold is always met first before the signal failure threshold.

Once the BER reaches the signal failure threshold level, then a signal failure alarm occurs and the receive is considered to be in signal failure condition. Based upon the APS configurations, all traffic is switched from the working port to the protecting port.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).
<*num*>	Specifies the Bit Error Rate signal failure threshold level in 10^{-n}. Specify any number for n between 3 to 5. The default is 3, indicating a threshold level of 10^{-3}. This means that a signal failure alarm occurs if the Bit Error Rate rises past the 1/1000 level.

Restrictions

None.

Example

To set the BER signal failure threshold level to 10^{-3} or 1/1000:

```
xp(config)# sonet set so.2.1 sf-ber 3
```

sonet set sts-stream-scramble

Purpose

Enables scrambling or descrambling of the STS (synchronous transfer signal) stream.

Format

sonet set <*SONET ports*> **sts-stream-scramble on| off**

Mode

Configure

Description

The **sonet set sts-stream-scramble** command allows you to enable scrambling and descrambling of the STS stream. Scrambling the STS stream is important in optimizing the transmission density of the data stream. Since all STS transmission use the same source clock for timing. Scrambling the payload using a random number generator converts the data stream to a more random sequence. This ensures optimal transmission density of the data stream.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).
on	Enables scrambling and descrambling of the STS stream.
off	Disables scrambling and descrambling of the STS stream.

Restrictions

None.

Example

To enable scrambling on port so.2.1:

```
xp(config)# sonet set so.2.1 sts-stream-scramble on
```

sonet set WTR-timer

Purpose

Sets the Wait-to-Restore timer.

Format

sonet set <*SONET ports*> **WTR-timer** <*num*>

Mode

Configure

Description

The **sonet set WTR-timer** command allows you to set the Wait-to-Restore timer. The WTR-timer specifies a time period that must expire before traffic is switched back to the working port from the protecting port. Once the signal degrade or failure condition has been corrected, APS waits until the WTR-timer expires before switching back to the working port.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).
<*num*>	Specifies the WTR timer (in minutes). Specify any number between 5 and 12 minutes. The **default** is 5 minutes.

Restrictions

None.

Example

To set the WTR-timer to 6 minutes on port so.2.1:

```
xp(config)# sonet set so.2.1 WTR-timer 6
```

sonet show aps

Purpose

Displays APS status.

Format

sonet show aps <*SONET ports*>

Mode

Enable

Description

The **sonet show aps** command allows you to display APS (automatic protection switching) status. This command allows you to display such APS parameters as protection level, working or protecting port, directionality, and switch status.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).

Restrictions

None.

Example

To display the APS status for port so.2.1:

```
xp# sonet show aps so.2.1
```

sonet show loopback

Purpose

Displays loopback status.

Format

sonet show loopback <*SONET ports*>

Mode

Enable

Description

The **sonet show loopback** command allows you to display loopback status for a specified SONET port. Loopback is used to verify connectivity between two devices.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).

Restrictions

None.

Example

To display the loopback status for port so.2.1:

```
xp# sonet show loopback so.2.1
```

sonet show medium

Purpose

Displays SONET optical line values.

Format

sonet show medium <*SONET ports*>

Mode

Enable

Description

The **sonet show medium** command allows you to display the various SONET optical line values associated with a SONET port. This command will allow you to display values such as framing status, line type, and administrator-specified circuit identifier.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).

Restrictions

None.

Example

To display optical line values for port so.2.1:

```
xp# sonet show medium so.2.1
```

sonet show pathtrace

Purpose

Displays received path trace messages.

Format

sonet show pathtrace <*SONET ports*>

Mode

Enable

Description

The **sonet show pathtrace** command allows you to display path trace messages received on a specified SONET port.

Parameters

<*SONET ports*>	Specifies the SONET port name(s).

Restrictions

None.

Example

To display the path trace messages for port so.2.1:

```
xp# sonet show pathtrace so.2.1
```

Chapter 64

ssh Commands

Secure Shell (**ssh**) is a "secure" replacement for Telnet. SSH provides the same remote access to the X-Pedition that Telnet provides, but does so securely by encrypting all session data—including passwords.

Note: When you enable the SSH server, the X-Pedition automatically disables Telnet access.

Command Summary

Table 50 lists the secure shell (**ssh**) commands. The sections following the table describe the command syntax.

Table 50. secure shell commands

ssh <*host*> **[encryption-preference** <*encryption-algorithm-list*>**]** **[escape** <*escape-character*>**| none] [login-as** <*username*>**]** **[mac-preference** <*mac-algorithm-list*>**] [no-compression] [port** <*tcp-port*>**]** **[protocol-version-preference** <*version-list*>**] [ssh1-encryption 3des| blowfish]**
ssh-client clear-known-hosts
ssh-client delete-host-key <*hostname*> **[dsa| rsa| rsa1]**
ssh-client import-host-keys <*filename*>
ssh-client set [encryption-preference <*cipher1*> [<*cipher2*>] …**]|** **[escape** <*escape-character*>**| none]| [mac-preference <***mac1*> [<*mac2*>] …**]| [no-compression]| [port** <*tcp-port*>**]| [protocol-version-preference** [**ssh1**] [**ssh2**]**]|** **[ssh1-encryption 3des|blowfish]| [strict-host-key-checking]| [username** <*name*>**]**
ssh-client set software-version-string <*version-string*>
ssh-server enable
ssh-server generate-host-key <*type*> **bits** <*bits*>
ssh-server set auth-grace-timeout <*timeout*>

Table 50. secure shell commands

ssh-server set encryption <*cipher1*> <*cipher2*> ...
ssh-server set listen-port <*port*>
ssh-server set mac <*mac1*> <*mac2*> ...
ssh-server set max-sessions <*limit*>
ssh-server set protocol-version <*version*>
ssh-server set server-key-lifetime <*time*>
ssh-server set software-version-string <*version-string*>
ssh-server show public-host-key <*type*> **fingerprint-format** <*format*>

ssh

Format

ssh <*host*> **[encryption-preference** <*encryption-algorithm-list*>**]**
[escape <*escape-character*>**| none] [login-as** <*username*>**]**
[mac-preference <*mac-algorithm-list*>**] [no-compression] [port** <*tcp-port*>**]**
[protocol-version-preference <*version-list*>**] [ssh1-encryption 3des| blowfish]**

Mode

Enable.

Description

The **ssh** command invokes the Secure Shell client. Any options specified on the command-line will override any defaults and configuration settings saved in the active configuration. No configuration is required in order to use the **ssh** command.

Note: SSH client requires firmware version E9.1.0.0 or later.

Parameters

<*host*>

The host name or IP address of the remote SSH server with which to connect.

encryption-preference <*encryption-algorithm-list*>

Specify a comma-separated list of SSH-2 encryption algorithms or *ciphers* to attempt, in order of preference. By default, the algorithms attempted are (in order of preference) AES, Triple-DES, Blowfish, CAST-128, and ARCFOUR. Valid ciphers and their corresponding names are shown in the following table.

Cipher	Cipher Name
Triple-DES	3des-cbc
AES	aes128-cbc
ARCFOUR	arcfour
Blowfish	blowfish-cbc
CAST-128	cast128-cbc

Note: Cipher names are case sensitive and, when formulating the comma-separated list, there should be **no** spaces.

escape <*escape-character*>**| none**

Specify a single character to use as the escape-sequence initiator. Use the circumflex character (^) as a prefix to designate a control character. Select the **none** option if you do not want the router to recognize escape sequences.

login-as <*username*>

Specify the name to use for authentication on the remote host. The username cannot exceed 32 characters in length.

mac-preference <*mac-algorithm-list*>

Specify a comma-separated list of SSH-2 Message Authentication Code algorithms or *MACs* to attempt, in order of preference. By default, the algorithms attempted are (in order of preference) HMAC-MD5, HMAC-SHA1, HMAC-RIPEMD160, HMAC-SHA1-96, and HMAC-MD5-96. Valid MACs and their corresponding names are shown in the following table.

MAC	MAC Name
HMAC-MD5	hmac-md5
HMAC-MD5-96	hmac-md5-96
HMAC-RIPEMD160	hmac-ripemd160
HMAC-SHA1	hmac-sha1
HMAC-SHA1-96	hmac-sha1-96

Note: MAC names are case sensitive and, when formulating the comma-separated list, there should be **no** spaces.

no-compression

Specifying this option will disable compression of session data which, by default, is compressed. When compression is enabled, the router compresses data prior to its encryption. Generally speaking, leaving compression *enabled* will result in better performance—it is far more taxing on system resources to encrypt data than to compress it.

port <*tcp-port*>

Specify an alternate TCP port (from 1 to 65,535 inclusive) to connect to on the remote host. Typically, SSH servers listen for incoming SSH connection requests on TCP port 22.

protocol-version-preference <*version-list*>

Specify a comma-separated list of which protocol versions to allow, in order of preference. By default, the allowed protocol versions are (in order of preference) SSH-2 and SSH-1. Valid protocol versions and their corresponding names are shown in the following table.

Protocol Version	Protocol Version Name
SSH-1	ssh1
SSH-2	ssh2

Note: Protocol version names are case-sensitive and, when formulating the comma-separated list, there should be **no** spaces.

ssh1-encryption 3des| blowfish

Specify a cipher to use for an SSH-1 session. Enter **3des** to use the Triple-DES cipher and **blowfish** to use the Blowfish cipher.

Restrictions

- If a PCMCIA flash card is not present in the router, some SSH client security features will be disabled. Enterasys Networks recommends that you use PCMCIA flash cards in all routers that will run the SSH client.
- SSH client requires firmware version E9.1.0.0 or later.

ssh-client clear-known-hosts

Format

ssh-client clear-known-hosts

Mode

Enable.

Description

The Known Hosts database stores host keys belonging to all known SSH servers and is used to verify the identity of a server each time an SSH connection is made. If unauthorized alterations are made to the Known Hosts database or if database tampering is otherwise detected, the router will not allow any new outbound SSH sessions. In such an event, entering the **ssh-client clear-known-hosts** command from the CLI will reset the Known Hosts database, allowing the router to rebuild data and restore access to new SSH sessions.

Note: For added security, only users with configuration-level privileges or knowledge of the configuration mode password (if enabled) may execute this command.

Parameters

None.

Restrictions

- This command requires that a PCMCIA flash card be present in the router.
- Only users with configuration-level privileges or knowledge of the configuration mode password (if enabled) may execute this command.
- SSH client requires firmware version E9.1.0.0 or later.

ssh-client delete-host-key

Format

ssh-client delete-host-key <*hostname*> **[dsa| rsa| rsa1]**

Mode

Enable.

Description

In the event that a particular host key is compromised or becomes outdated, you can delete the key with the **ssh-client delete-host-key** command. When you specify the optional key-type, the router will remove only keys of that type from the host—otherwise, all keys that belong to the specified host will be deleted.

Note: For added security, only users with configuration-level privileges or knowledge of the configuration mode password (if enabled) may execute this command.

Parameters

<*hostname*> [**dsa**|**rsa**|**rsa1**]

The name or IP address of the host whose key(s) you want to delete. To delete keys of a specific type from the host, enter one of the following:

dsa	Delete only DSA keys.
rsa	Delete only RSA keys.
rsa1	Delete only RSA1 keys.

Restrictions

- This command requires that a PCMCIA flash card be present in the system chassis.
- Only users with configuration-level privileges or knowledge of the configuration mode password (if enabled) may execute this command.
- SSH client requires firmware version E9.1.0.0 or later.

ssh-client import-host-keys

Format

ssh-client import-host-keys <*filename***>**

Mode

Enable.

Description

The **ssh-client import-host-keys** command is useful for initially populating the Known Hosts database on a number of X-Pedition routers. The X-Pedition allows you to create a text file that contains a list of the hosts you wish to add to the Known Hosts databases, then upload the file to each router you wish to configure. Executing this command on each router will import the host keys from the uploaded file and will create an identical Known Hosts database list on each router.

The text-file used to import keys must adhere to the following:

- Each host key must occupy a single line.
- No line should exceed 8,192 characters in length.
- Each line should be formatted as follows: *[hostname,]ip-address public-host-key*

  Note: Although *hostname* is optional, it should be followed by a comma if used and there should be no spaces in or between *hostname* and *ip-address*. *Public-host-key* should be the ASCII representation of the DSA, RSA, or RSA1 public host key. The ASCII representation of these keys is obtained by entering the **ssh-server show public-host-key** command from the CLI. The fingerprint should not be included—only the part of the key displayed after "**Key:**".

Parameters

<filename>

The name of the file on the local system that contains the population list for the Known Hosts database.

Restrictions

- This command requires that a PCMCIA flash card be present in the system chassis.
- For added security, only users with configuration-level privileges or knowledge of the configuration mode password (if enabled) may execute this command.
- SSH client requires firmware version E9.1.0.0 or later.

Example

The following example demonstrates how to import host keys from the text file "**host_keys**" located on a remote TFTP server::

```
xp# copy tftp://192.168.1.1/host_keys to host_keys
xp# ssh-client import-host-keys bootflash:host_keys
```

ssh-client set

Format

ssh-client set [encryption-preference <*cipher1*> [<*cipher2*>] …**]| [escape** <*escape-character*>| **none]| [mac-preference <***mac1*> [<*mac2*>] …**]| [no-compression]| [port** <*tcp-port*>**]| [protocol-version-preference** [**ssh1**] [**ssh2**]**]| [ssh1-encryption 3des**|**blowfish]| [strict-host-key-checking]| [username** <*name*>**]**

Mode

Configure.

Description

Customizes default values for SSH client sessions. Configured values will be used by all sessions unless explicitly overridden by an alternate value on the SSH command-line.

Parameters

encryption-preference <*cipher1*> [<*cipher2*>] …

Sets the default SSH-2 encryption algorithms (i.e., *ciphers*) and priority for all future SSH-2 client sessions. The SSH client will attempt only the specified ciphers and in the order they are listed below. Available options for the ciphers appear in the following table:

Option	Cipher
3des-cbc	Triple-DES
aes128-cbc	AES
arcfour	ARCFOUR
blowfish-cbc	Blowfish
cast128-cbc	CAST-128

escape <*escape-character*>**| none**

Specify a single character to use as the escape-sequence initiator. Use the circumflex character (^) as a prefix to designate a control character. Select the **none** option if you do not want the router to recognize escape sequences.

mac-preference <*mac1*> [<*mac2*>] …

Sets the default SSH-2 Message Authentication Code or *MAC* algorithms and priority for all future SSH-2 client sessions. The SSH client will attempt only the specified MACs in the order listed below. Available options for the MACs appear in the following table.

Option	MAC
hmac-md5	HMAC-MD5
hmac-md5-96	HMAC-MD5-96
hmac-ripemd160	HMAC-RIPEMD160
hmac-sha1	HMAC-SHA1
hmac-sha1-96	HMAC-SHA1-96

no-compression

Specifying this option will disable compression of session data for all future SSH sessions which, by default, is compressed. When compression is enabled, the router compresses data prior to its encryption. Generally speaking, leaving compression *enabled* will result in better performance—it is far more taxing on system resources to encrypt data than to compress it.

port <*tcp-port*>

Specify an alternate TCP port (from 1 to 65,535 inclusive) to connect to on the remote host for all future SSH sessions. Typically, SSH servers listen for incoming SSH connection requests on TCP port 22.

protocol-version-preference [**ssh1**] [**ssh2**]

Sets the default protocol versions and priority for all future SSH sessions. Only the specified version(s) will be attempted, and in the order listed below.

Protocol Version	Protocol Version Name
SSH-1	ssh1
SSH-2	ssh2

Note: Protocol version names are case-sensitive and, when formulating the comma-separated list, there should be **no** spaces.

ssh1-encryption 3des| blowfish

Specify a cipher to use for all future SSH-1 client sessions. Enter **3des** to use the Triple-DES cipher and **blowfish** to use the Blowfish cipher.

strict-host-key-checking

Under normal circumstances, when an ordinary user connects an SSH session to an unknown host, the user receives a warning that the host is unknown. If the user elects to continue, the new host key is added automatically to the Known Hosts database.

The **ssh-client set strict-host-key-checking** configuration command changes the default behavior by preventing users from automatically adding new host keys to the Known Hosts database. With this option enabled, users may add new keys to the Known Hosts database via the **ssh-client import-host-keys** command only. This provides a higher level of security by allowing only users with configuration-level privileges to add new keys to the Known Hosts database.

Note: This parameter requires that a PCMCIA flash card be present in the system chassis—otherwise, the router will not permit any outbound SSH client sessions.

username <*name***>**

The default username to use for all future SSH client sessions. At the start of a client session, the router sends this username to the remote host for authentication. The user name should not exceed 32 characters in length.

Restrictions

- The **strict-host-key-checking** option requires that a PCMCIA flash card be present in the system chassis—otherwise, the router will not permit any outbound SSH client sessions.
- SSH client requires firmware version E9.1.0.0 or later.

ssh-client set software-version-string

Format

ssh-client set software-version-string <*version-string*>

Mode

Configure

Description

This command allows users to change the built-in software version string sent to SSH servers—useful in cases where incompatibilities exist between the X-Pedition SSH client and a third-party server.

Parameter

software-version-string <*version-string*>
Enter the version string (up to 63 characters in length). By default, this string is XPSSH.

Restrictions

SSH client requires firmware version E9.1.0.0 or later.

Examples

The following example uses a the software version string, “OpenSSH_2.9.9p2.”

```
xp (config)# ssh-server set software-version-string OpenSSH_2.9.9p2
```

ssh-server enable

Purpose

Start the secure shell server.

Format

ssh-server enable

Mode

Configuration.

Description

Launches the secure shell server. In order for the server to start, at least one host key must exist and the key must be compatible with the configured protocol-version.

Parameters

None.

Restrictions

When you enable the SSH server, the X-Pedition automatically disables Telnet access.

Example

To start the secure shell server, enter the following:

```
ssh-server enable
```

ssh-server generate-host-key

Purpose

Generate host key pairs.

Format

ssh-server generate-host-key <*type*> **bits** <*bits*>

Mode

Enable.

Description

Generates asymmetric host key pairs. The host key is used to uniquely and securely identify the SSH server to the SSH client. In other words, the host key makes it possible for the SSH client to guarantee that it is connected to the intended host, not an imposter.

Parameters

<*type*> There are three key types of host key pairs—*rsa1*, *rsa*, and *dsa*.

- **rsa1** An RSA1 key must be generated in order to interoperate with SSH-1 clients. RSA1 keys use the RSA public-key encryption algorithm.
- **rsa** An RSA key can be used to interoperate with SSH-2 clients. RSA keys use the RSA public key encryption algorithm.
- **dsa** A DSA key can be used to interoperate with SSH-2 clients. DSA keys use the DSA public key encryption algorithm.
- **all** Generates one key of each key type.

<*bits*> Specifies the bit length (512–4096) of the keys to generate. In general, 1,024 bits (the default) is considered very secure. Lengths greater than 1,024 bits are not considered to provide much additional security and will slow down cryptographic operations. For example, keys that are 1,024 bits or less in size take only a few minutes to generate. In contrast, keys larger than 1,024 bits may take several hours to generate (e.g., 4,096 bit keys may require several hours).

Restrictions

None.

Example

To generate a 1,024-bit DSA host key pair, enter the following:

```
ssh-server generate-host-key dsa bits 1024
```

ssh-server set auth-grace-timeout

Purpose

Sets the authentication time limit for connecting clients.

Format

ssh-server set auth-grace-timeout <*timeout*>

Mode

Configuration.

Description

Sets the time limit used to authenticate and connect clients. Clients that take longer than this to connect and successfully authenticate will be disconnected.

Parameter

<*timeout*> Number of seconds (10-120) given to clients to successfully authenticate. By default, this value is 60.

Restrictions

None.

Example

To set a 90-second authentication time limit, enter the following:

```
xp(config)# ssh-server set auth-grace-timeout 90
```

ssh-server set encryption

Purpose

Enable various encryption algorithms that will “scramble” and protect data.

Format

ssh-server set encryption <*cipher1*> <*cipher2*> ...

Mode

Configuration.

Description

Specifies which encryption algorithms the X-Pedition will support. Encryption algorithms provide privacy of session data by “scrambling” the contents of the message so that only the intended recipient will be able to “unscramble” the data.

Parameters

cipher(n) The name of the cipher encryption algorithm to support. To select multiple ciphers, enter the name of each cipher or enter the keyword **all** (the default) to support all encryption algorithms.

- **aes128-cbc** 128-bit AES encryption operates in cipher-block-chaining mode. AES is relatively fast, but has not as yet seen as much real-world use as some of the other options. AES was formerly known as Rijndael.
- **3des-cbc** Triple-DES encryption utilizes three separate 56-bit keys operating in cipher-block-chaining mode. This encryption is considered by many to be the most secure widely-available bulk cipher because of its lengthy record of real-world use. Very slow compared to the other available algorithms. Key generation, in particular, is very slow and may result in a noticeable delay when generating new session keys.
- **blowfish-cbc** 128-bit Blowfish encryption operates in cipher-block-chaining mode. Blowfish has received much scrutiny, but has so far proved secure. Its speed is comparable to that of AES encryption.
- **cast128-cbc** CAST-128 encryption (128-bit) operates in cipher-block-chaining mode. Slightly slower than Blowfish, CAST-128 is perhaps more widely implemented due to its standardization in RFC-2144.

arcfour	128-bit ARCFOUR encryption is a “stream” cipher. ARCFOUR is the only non-cipher-block-chaining cipher widely used with SSH implementations. Although it has received little scrutiny, ARCFOUR is considered secure by many. ACRFOUR is the fastest of the available options.

Restrictions

None.

Example

To set Triple-DES, 128-bit Blowfish, and 128-bit ARCFOUR encryption, enter the following:

```
xp(config)# ssh-server set encryption 3des-cbc blowfish-cbc arcfour
```

ssh-server set listen-port

Purpose

Set the TCP port on which secure shell server will listen.

Format

ssh-server set listen-port <*port*>

Mode

Configuration.

Description

Secure shell servers normally listen on TCP port 22. This command allows you to change the TCP port.

Parameter

<*port*> The TCP port number on which to listen. By default, the TCP port is 22.

Restrictions

None.

Example

To assign secure shell server to listen on TCP port ten, enter the following:

```
xp(config)# ssh-server set listen-port 10
```

ssh-server set mac

Purpose

Enable support for MAC algorithms under SSH-2 to provide additional authentication for session data.

Format

ssh-server set mac <*mac1*> <*mac2*> ...

Mode

Configuration.

Description

Specifies which Message Authentication Code (MAC) algorithms the X-Pedition will support. MAC algorithms provide authenticity for session data by digitally "signing" each message. The digital signature prevents a third party from altering or falsifying any session data. Applies to SSH-2 only.

Parameters

mac(n) The name of the MAC to enable. To specify multiple MACs, enter the name of each MAC or specify the keyword **all** (the default) to select all MACs. When a client connects, the first MAC listed that is supported by both the client and the server will be used for the session.

hmac-sha1	The HMAC-SHA1 MAC algorithm based on the SHA-1 message digest algorithm.
hmac-md5	The HMAC-MD5 MAC algorithm based on the MD5 message digest algorithm.
hmac-ripemd160	The HMAC-RIPEMD160 MAC algorithm based on the RIPEMD-160 message digest algorithm.
hmac-sha1-96	The HMAC-SHA1-96 MAC algorithm based on the SHA-1 message digest algorithm.
hmac-md5-96	The HMAC-MD5-96 MAC algorithm based on the MD5 message digest algorithm.

Restrictions

None.

Example

To support the HMAC-SHA1, HMAC-MD5, and HMAC-MD5-96 MAC algorithms under SSH-2, enter the following:

```
xp(config)# ssh-server set mac hmac-sha1 hmac-md5 hmac-md5-96
```

ssh-server set max-sessions

Purpose

Set the maximum allowed number of simultaneous secure-shell sessions.

Format

ssh-server set max-sessions <*limit*>

Mode

Configuration.

Description

This command allows you to limit the number of secure shell sessions that can be active simultaneously. Limiting this to fewer than four sessions can be useful in preventing multiple users from modifying the configuration at the same time. This command is also useful for limiting the amount of CPU and memory resources consumed by active SSH sessions, since the cryptographic operations performed by active sessions can be somewhat resource intensive.

Parameter

<*limit*>	The maximum number (1-4) of secure-shell sessions that can be active simultaneously. The default is 4.

Restrictions

None.

Example

To allow three simultaneous secure shell sessions, enter the following:

```
xp(config)# ssh-server set max-sessions 3
```

ssh-server set protocol-version

Purpose

Select which Secure Shell protocol version(s) to support.

Format

ssh-server set protocol-version <*version*>

Mode

Configuration.

Description

Specifies which SSH protocol version(s) to support. Secure shell protocol versions include SSH-1 and SSH-2—SSH-2 is considered to be more secure than SSH-1. By default, the X-Pedition enables both SSH protocols.

Parameters

<*version*> Specifies which Secure Shell protocol version(s) connecting clients will be allowed to use.

- **ssh1** Clients will be allowed to connect using the SSH-1 protocol only.
- **ssh2** Clients will be allowed to connect using the SSH-2 protocol only.
- **both** Clients will be allowed to connect using either SSH-1 or SSH-2 protocols. The X-Pedition uses this as the default.

Restrictions

None.

Example

To enable support for both versions of the secure shell protocol, enter the following from configuration mode:

```
xp(config)# ssh-server set protocol-version both
```

ssh-server set server-key-lifetime

Purpose

Set the regeneration period for the server key.

Format

ssh-server set server-key-lifetime <*time*>

Mode

Configuration.

Description

SSH-1 uses the server key to provide “perfect forward secrecy” for SSH-1 sessions. The **ssh-server set server-key-lifetime** command determines how often the server key regenerates. More frequent regeneration may provide slightly increased security, but comes at the cost of increased CPU utilization. The server key always generates a 768-bit RSA1 key and is not used for SSH-2. For additional information regarding perfect forward security refer to the *Enterasys X-Pedition User Reference Manual*.

Parameter

<*time*> The number of minutes (5–480) that will transpire before the server key regenerates. By default, this value is 60 minutes.

Restrictions

None.

Example

To set a delay time of 2 hours (120 minutes) before regenerating the server key, enter the following from configuration mode:

```
xp(config)# ssh-server set server-key-lifetime 120
```

ssh-server set software-version-string

Format

ssh-sever set software-version-string <*version-string*>

Mode

Configure

Description

This command allows users to change the built-in software version string sent to SSH clients—useful in cases where incompatibilities exist between the X-Pedition SSH server and a third-party client.

Parameter

software-version-string <*version-string*>
Enter the version string (up to 63 characters in length). By default, this string is XPSSH.

Restrictions

None

Examples

The following example uses a the software version string, “OpenSSH_2.9.9p2.”

```
xp (config)# ssh-server set software-version-string OpenSSH_2.9.9p2
```

ssh-server show public-host-key

Purpose

Shows the public component and fingerprint of the specified host key pair.

Format

ssh-server show public-host-key <*type*> **fingerprint-format** <*format*>

Mode

Enable.

Description

Shows the public component and fingerprint of a specific host key pair. The fingerprint is often useful for determining the authenticity of the host when connecting with a client for the first time. If necessary, you can use this command to copy the public key.

Parameters

<*type*> Specifies which of the host keys to show.

rsa1 Shows the RSA1 public key and fingerprint.

rsa Shows the RSA public key and fingerprint.

dsa Shows the DSA public key and fingerprint.

<*format*> The format in which to display the fingerprint (optional). The default format is hexadecimal.

bubble-babble
Bubble babble is a method of representing the fingerprint as a string of “real" words to make the fingerprint easier to remember. The “words" are not necessarily real words, but they look more like words than a string of hexadecimal characters. Bubble-babble may be more useful than hex when verbally communicating the fingerprint to a user.

hex Shows the exact fingerprint in hexadecimal format. Although this format is a more concise way to display the fingerprint, it may be harder to remember and difficult to communicate verbally without transcription errors.

Restrictions

None.

Example

To show the RSA public key and fingerprint in bubble babble format, enter the following from enable mode:

```
xp(config)# ssh-server show public-host-key rsa fingerprint-format bubble-babble
```

Chapter 65

statistics Commands

The **statistics** commands allow the user to display statistics for various X-Pedition features. You also can clear some statistics.

Command Summary

Table 51 lists the statistics commands. The sections following the table describe the command syntax.

Table 51. statistics commands

statistics clear <*statistic-type*>
statistics show appletalk-atp| appletalk-ddp| appletalk-echo| appletalk-interface| appletalk-nbp| appletalk-routing| appletalk-zip
statistics show arp <*Interface Name*>**| all**
statistics show icmp
statistics show ip
statistics show ip-interface <*string*>**|all [packets] [bytes] [errors] [input] [output]|verbose**
statistics show ip-routing
statistics show ipx
statistics show ipx-interface <*string*>**|all packets bytes errors input output verbose**
statistics show ipx-routing
statistics show multicast
statistics show framer <*port-list*>

Table 51. statistics commands (Continued)

statistics show port-errors <*port/*SmartTRUNK-*list*>**|all-ports**
statistics show port-packets <*port-list*>**|all-ports**
statistics show port-stats <*port/*SmartTRUNK-*list*>**|all-ports**
statistics show rarp <*string*>**|all**
statistics show summary-stats
statistics show tcp
statistics show udp
statistics show most-active
statistics show vlan all|<*string*>

statistics clear

Purpose

Clear statistics.

Format

statistics clear <*statistic-type*>

Mode

Enable

Description

The **statistics clear** command clears port statistics, error statistics, or RMON statistics. When you clear statistics, the X-Pedition sets the counters for the cleared statistics to 0, then begins accumulating the statistics again.

Parameters

<*statistic-type*>

Type of statistics you want to clear. Specify one of the following:

ip Clears all IP statistics.

ipx Clears all IPX statistics.

appletalk Clears all AppleTalk statistics. When you clear statistics, the X-Pedition sets the counters for the cleared statistics to 0, then begins accumulating the statistics again.

icmp Clears all ICMP statistics.

- **all** Clear all statistics.
- **input** Clear input statistics.
- **output** Clear output statistics.
- **disabled** Clear disabled statistics.

port-errors Clears error statistics for the specified LAN port, WAN module, or SmartTRUNK. Specify **all-ports** to clear error statistics for all ports.

port-packets Clears all port packet statistics for a specified POS module or list of POS modules. Specify **all-ports** to clear packet statistics for all ports.

port-stats Clears all normal (non-error) statistics for the specified port. Specify **all-ports** to clear port statistics for all ports.

vlan all|<*string*>

Specify the keyword **all** to reset all counters used for per-VLAN packet accounting or enter the name of a specific VLAN.

Restrictions

None.

statistics show appletalk

Purpose

Displays various AppleTalk statistics.

Format

statistics show appletalk-atp|-ddp|-echo|-interface|-nbp|-routing|-zip

Mode

Enable

Parameters

atp	Displays statistics for the AppleTalk Transaction Protocol.
ddp	Displays statistics for the Datagram Delivery Protocol
echo	Displays statistics for the Echo Protocol.
interface	Displays interface statistics.
nbp	Displays statistics for the Name Binding Protocol (NBP).
routing	Displays statistics for the Routing Table Maintenance Protocol (RTMP).
zip	Displays statistics for the Zone Information Protocol (ZIP).

Restrictions

None.

statistics show arp

Purpose

Display address resolution protocol (ARP) statistics.

Format

statistics show arp <*Interface Name*>**| all**

Mode

Enable

Description

The **arp show statistics** command displays ARP statistics, such as the total number of ARP requests and replies.

Parameters

<*Interface Name*> Displays ARP statistics for the specified interface.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

all Displays ARP statistics for all router interfaces.

Restrictions

None.

Example

To display ARP statistics on interface ‘en0’:

```
xp# statistics show arp en0

Interface en0:
      1 requests sent
      19 replies sent
      0 proxy replies sent
      Last 5 Requests Sent
      ----- no arp requests sent -----
      Last 5 Replies Sent
      134.141.179.129  | XP1     16:BF:21      |2000-04-17 13:12:49
      134.141.179.129  | XP1     16:BF:21      |2000-04-17 13:50:15
      134.141.179.129  | XP1     16:BF:21      |2000-04-17 15:32:32
      134.141.179.129  | XP1     16:BF:21      |2000-04-17 16:17:19
      134.141.179.129  | XP1     16:BF:21      |2000-04-17 11:12:44

      Last 5 ARP packets received on wrong interface
      ----- no arp packets received on wrong interface -----
```

Field Definitions

Field	Description
requests sent	Displays how many ARP requests have been sent out to an ARP server for address resolution.
replies sent	Displays how many ARP replies have been sent out to an ARP client in response to request packets.
proxy replies sent	Displays how many proxy ARP replies have been sent out in response to request packets. A proxy router serving as a gateway to a subnet would respond with a proxy reply.
Last 5 Requests sent	Displays the last five ARP requests sent, including the following information: target MAC address, date and time sent.
Last 5 Replies sent	Displays the last five ARP replies sent, including the following information: target IP address, date and time sent.
Last 5 ARP packets received on wrong interface	Displays the last five ARP packets that has been received on the wrong interface.

statistics show icmp

Purpose

Display internet control message protocol (ICMP) statistics.

Format

statistics show icmp

Mode

Enable

Parameters

None.

Restrictions

None.

Example

To display ICMP statistics:

```
xp# statistics show icmp
icmp:
      0 messages with bad code fields
      0 messages smaller than minimum length
      0 bad checksums
      0 messages with bad length
      0 message responses generated
```

Field Definitions

Field	Description
messages with bad code fields	Displays the number of ICMP messages processed by the router with a bad code field. The code field within the ICMP header uses a number to specify the message content of the ICMP message. An invalid number within the code field would show in this statistic parameter.
messages smaller than minimum length	Displays the number of ICMP messages processed by the router that didn't meet a minimum length requirement.
bad checksums	Displays the number of ICMP messages processed by the router with bad checksums. The checksum field within the ICMP header is used to verify that the message was transmitted error-free. A bad checksum indicates an ICMP message with errors.
messages with bad length	Displays the number of ICMP messages processed by the router with bad or invalid length.
message responses generated	Displays the number of ICMP responses that have been generated by the router in response to ICMP messages.

statistics show ip

Purpose

Display Internet Protocol (IP) statistics for all packets received or sent by the router software.

Format

statistics show ip

Mode

Enable

Parameters

None.

Restrictions

IP statistics on hardware-routed IP flows are not included in the statistics displayed by this command. To see statistics collected by router hardware, use the **statistics show ip-interface** and **statistics show port-packets** commands.

Example

To display IP statistics:

```
xp# statistics show ip
ip:
        78564 total packets received
        0 bad header checksums
        0 packets with size smaller than minimum
        0 packets with data size < data length
        0 packets with header length < data size
        0 packets with data length < header length
        0 packets with bad options
        0 packets with incorrect version number
        0 fragments received
        0 fragments dropped (dup or out of space)
        0 fragments dropped after timeout
        0 packets reassembled ok
        2984 packets for this host
        0 packets for unknown/unsupported protocol
        0 packets forwarded
        75580 packets not forwardable
        0 redirects sent
        2120 packets sent from this host
        0 packets sent with fabricated ip header
        0 output packets dropped due to no bufs, etc.
        0 output packets discarded due to no route
        0 output datagrams fragmented
        0 fragments created
        0 datagrams that can't be fragmented
```

Field Definitions

Field	Description
total packets received	The total number of IP packets forwarded and dropped by the router.
bad header checksums	The number of IP packets received with bad checksums. The checksum field within the IP header is used to verify that the packet was transmitted error-free. A bad checksum indicates an IP packet with errors.
packets w/size smaller than minimum	The number of IP packets received that didn’t meet a minimum length requirement.
packets w/data size < data length	The number of IP packets received that contain a data size smaller than the data length specified in the IP header.

Field	Description
packets w/header length < data size	The number of IP packets received that contain an IP header length smaller than the data size within the packet.
packets w/data length < header length	The number of IP packets received that contain a data length smaller than the IP header length.
packets w/incorrect version number	The number of IP packets received with an incorrect IP version number. The IP version number field in the IP header is used to specify whether the packet is formatted for IPv4 or IPv6.
fragments received	The number of datagram fragments received by the router.
fragments dropped	The number of datagram fragments dropped by the router.
fragments dropped after timeout	The number of datagram fragments dropped by the router after a timeout.
packets reassembled ok	The number of IP packets containing fragmented datagrams that were reassembled successfully by the router.
packets for this host	The total number of IP packets received that were intended for the router as the destination.
packets for unknown protocol	The number of IP packets received that are of an unknown or unsupported routed protocol.
packets forwarded	The number of IP packets received that were forwarded on to another host.
packets not forwardable	The total number of IP packets received that the router could not forward on to another host.
redirects sent	The number of redirects sent by the router.
packets sent from this host	The total number of IP packets sent by the router.
packets sent w/fabricated ip header	The number of IP packets sent after attaching an IP header to the packet.
output packets dropped due to no bufs	The number of IP packets dropped before being sent due to a lack of output buffer space.
output packets discarded due to no route	The number of IP packets dropped before being sent due to a lack of IP routing information.

Field	Description
output datagrams fragmented	The number of datagrams that were fragmented into two or more IP packets before being sent out by the router.
fragments created	The number of datagram fragments created.
datagrams that can’t be fragmented	The number of datagrams that were not successfully fragmented into two or more IP packets.

statistics show ip-interface

Purpose

Display IP interface statistics.

Note: Interface statistics originate from hardware counters on a port basis. Therefore, two interfaces on the same physical port will have identical statistics.

Format

statistics show ip-interface <*string*>**|all [packets] [bytes] [errors] [input] [output]| verbose**

Mode

Enable

Parameters

<*string*>**|all**

Specifies the name of an interface. Specify **all** to display IP statistics for all interfaces.

Note: The **statistics show ip-interface** command cannot display statics for any interface that contains an ATM port. If a user attempts to display this information, the router will display “n/a” on the console.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

packets

Specify this optional parameter to display the number of packets that have passed through the interface.

bytes

Specify this optional parameter to display the number of bytes that have passed through the interface.

errors

Specify this optional parameter to display the number of packets with errors detected through the interface.

input

Specify this optional parameter to display interface statistics for the input side.

output

Specify this optional parameter to display interface statistics for the output side.

verbose
Specify this optional parameter to display statistics on the number of packets, bytes, and errors on both the input and output sides of the interface.

Restrictions

The **statistics show ip-interface** command cannot display statics for any interface that contains an ATM port. If a user attempts to display this information, the router will display “n/a” on the console.

Example

To display interface statistics on interface ‘en0’:

```
xp# statistics show ip-interface en0 verbose

Name In-frames Out-frames   In-bytes  Out-bytes  In-errors Out-errors
en0 0          0          0         0          0         0
```

Field Definitions

Field	Description
In-frames	Displays the number of packets that have entered the interface.
Out-frames	Displays the number of packets that have exited the interface.
In-bytes	Displays the number of bytes that have entered the interface.
Out-bytes	Displays the number of bytes that have exited the interface.
In-errors	Displays the number of packets with errors detected entering the interface.
Out-errors	Displays the number of packets with errors detected exiting the interface.

Note: Interface statistics originate from hardware counters on a port basis. Therefore, two interfaces on the same physical port will have identical statistics.

statistics show ip-routing

Purpose

Display unicast IP routing statistics.

Format

statistics show ip-routing

Mode

Enable

Parameters

None.

Restrictions

None.

Example

To display routing statistics:

```
xp# statistics show ip-routing
routing:
        0 bad routing redirects
        0 dynamically created routes
        0 new gateways due to redirects
        1141 destinations found unreachable
        0 uses of a wildcard route
```

Field Definitions

Field	Description
bad routing redirects	Displays the number of bad redirects have occurred. A redirect occurs in the case where the destination interface is the same as the source interface.
dynamically created routes	Displays the number of IP routes have been created using a routing protocol, as opposed to static routes which are user-defined.
new gateways due to redirects	Displays the number of new gateways have been added into the routing table due to redirects.
destinations found unreachable	Displays the number of destination addresses that have been found to be unreachable in the routing table. A destination may be unreachable due to the route being expired or being unavailable due to network changes.
uses of a wildcard route	Displays the number of times that a wildcard route has been used to forward a packet onto the next-hop destination.

statistics show ipx

Purpose

Display internetwork packet exchange (IPX) statistics.

Format

statistics show ipx

Mode

Enable

Parameters

None.

Restrictions

If you configure multiple protocol types (e.g., IP and IPX) on the same port(s), interface statistics collected for the port(s) will not be accurate—both protocols will gather port statistics.

Example

To display IPX statistics:

```
xp# statistics show ipx
ipx:
        0 total packets received
        0 packets with bad checksums
        0 packets smaller than advertised
        0 packets smaller than a header
        0 packets forwarded
        0 packets not forwardable
        0 packets for this host
        0 packets sent from this host
        0 packets dropped due to no bufs, etc.
        0 packets discarded due to no route
        0 packets too big
        0 packets with too many hops
        0 packets of type 20
        0 packets discarded due to infiltering
        0 packets discarded due to outfiltering
        0 packets with misc protocol errors
        0 rip packets discarded due to socket buffer full
        0 sap packets discarded due to socket buffer full
        0 rip req packets discarded due to socket buffer full
        0 sap gns packets discarded due to socket buffer full
        0 packets discarded due to port of entry zero
        0 packets discarded due to sourced by us
```

Field Definitions

Field	Description
total packets received	Displays the total number of IPX packets received by the router, including all forwarded and dropped packets.
bad header checksums	Displays the number of IPX packets received by the router with bad checksums. The checksum field within the IPX header is used to verify that the packet was transmitted error-free. A bad checksum indicates an IPX packet with errors.
packets smaller than advertised	Displays the number of IPX packets received by the router that are smaller than what the header indicates as the size.
packets smaller than a header	Displays the number of IPX packets received by the router that are smaller than the IPX header.

Field	Description
packets forwarded	Displays the number of IPX packets received by the router that have been forwarded onto the next-hop destination.
packets not forwardable	Displays the total number of IPX packets received by the router that could not be forwarded onto another host.
packets for this host	Displays the total number of IPX packets received that were intended for the router as the destination.
packets sent from this host	Displays the total number of IPX packets sent out by the router.
packets dropped due to no bufs	Displays the total number of IPX packets dropped before being sent out by the router because of lack of buffer space.
packets discarded due to no route	Displays the total number of IPX packets dropped before being sent out by the router because of no IPX routing information.
packets too big	Displays the total number of IPX packets that exceed a size threshold.
packets with too many hops	Displays the total number of IPX packets that exceed a number of hops threshold.
packets of type 20	Displays the total number of NetBIOS packets.
packets discarded due to infiltering	Displays the total number of incoming IPX packets that have been discarded due to filtering. Filtering is based upon various access control lists (ACL) such as IPX ACL, SAP ACL, and RIP ACL.
packets discarded due to outfiltering	Displays the total number of outgoing IPX packets that have been discarded due to filtering. Filtering is based upon various access control lists (ACL) such as IPX ACL, SAP ACL, and RIP ACL.
packets with misc protocol errors	Displays the total number of IPX packets containing routing protocol errors.
rip packets discarded	Displays the total number of Routing Information Protocol (RIP) packets that have been discarded due to the socket buffer being full.
sap packets discarded	Displays the total number of Server Advertisement Protocol (SAP) packets that have been discarded due to the socket buffer being full.

Field	Description
rip req packets discarded	Displays the total number of Routing Information Protocol (RIP) request packets that have been discarded due to the socket buffer being full.
sap gns packets discarded	Displays the total number of Service Advertisement Protocol (SAP) Get Nearest Server (GNS) packets that have been discarded due to the socket buffer being full.
packets discarded due to port of entry zero	Displays the total number of received IPX packets that were discarded because of a poe value of 0 in the packet header.
packets discarded due to sourced by us	Displays the total number of received IPX packets that have been discarded because they were sent by us.

statistics show ipx-interface

Purpose

Display IPX interface statistics.

Note: Interface statistics originate from hardware counters on a port basis. Therefore, two interfaces on the same physical port will have identical statistics.

Format

statistics show ipx-interface <*string*>**|all [packets] [bytes] [errors] [input] [output]|verbose**

Mode

Enable

Parameters

<*string*>**|all** Specifies the name of an interface. Specify **all** to display IPX statistics for all interfaces.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

packets Specify this optional parameter to display the number of packets that have passed through the interface.

bytes Specify this optional parameter to display the number of bytes that have passed through the interface.

errors Specify this optional parameter to display the number of packets with errors detected through the interface.

input Specify this optional parameter to display interface statistics for the input side.

output Specify this optional parameter to display interface statistics for the output side.

verbose Specify this optional parameter to display statistics on the number of packets, bytes, and errors on both the input and output sides of the interface.

Restrictions

None.

Example

To display interface statistics on interface ‘en0’:

```
xp# statistics show ipx-interface en0 verbose

Name In-frames Out-frames   In-bytes  Out-bytes  In-errors Out-errors
en0 0          0          0         0          0
```

Field Definitions

Field	Description
In-frames	Displays the number of packets that have entered the interface.
Out-frames	Displays the number of packets that have exited the interface.
In-bytes	Displays the number of bytes that have entered the interface.
Out-bytes	Displays the number of bytes that have exited the interface.
In-errors	Displays the number of packets with errors detected entering the interface.
Out-errors	Displays the number of packets with errors detected exiting the interface.

Note: Interface statistics originate from hardware counters on a port basis. Therefore, two interfaces on the same physical port will have identical statistics.

statistics show ipx-routing

Purpose

Display IPX routing statistics.

Format

statistics show ipx-routing

Mode

Enable

Parameters

None.

Restrictions

None.

Example

To display routing statistics:

```
xp# statistics show ipx-routing
routing:
        0 bad routing redirects
        0 dynamically created routes
        0 new gateways due to redirects
        1141 destinations found unreachable
        0 uses of a wildcard route
```

Field Definitions

Field	Description
bad routing redirects	Displays the number of bad redirects have occurred. A redirect occurs in the case where the destination interface is the same as the source interface.
dynamically created routes	Displays the number of IPX routes have been created using a routing protocol, as opposed to static routes which are user-defined.
new gateways due to redirects	Displays the number of new gateways have been added into the routing table due to redirects.
destinations found unreachable	Displays the number of destination addresses that have been found to be unreachable in the routing table. A destination may be unreachable due to the route being expired or being unavailable due to network changes.
uses of a wildcard route	Displays the number of times that a wildcard route has been used to forward a packet onto the next-hop destination.

statistics show multicast

Purpose

Display multicast statistics.

Format

statistics show multicast

Mode

Enable

Parameters

None.

Restrictions

None.

Example

To display multicast statistics:

```
xp# statistics show multicast
multicast forwarding:
        0 multicast forwarding cache lookups
        0 multicast forwarding cache misses
        0 upcalls to mrouted
        0 upcall queue overflows
        0 upcalls dropped due to full socket buffer
        0 cache cleanups
        0 datagrams with no route for origin
        0 datagrams arrived with bad tunneling
        0 datagrams could not be tunneled
        0 datagrams arrived on wrong interface
        0 datagrams selectively dropped
        0 datagrams dropped due to queue overflow
        0 datagrams dropped for being too large
```

Field Definitions

Field	Description
multicast forwarding cache lookups	This counter increments whenever a multicast packet does a “route” lookup in software. If a multicast packet hits the CPU and a forwarding decision already exists, the X-Pedition increments this counter.
multicast forwarding cache misses	This counter increments whenever the CPU receives an unlearned multicast packet. A cache miss can also result in an upcall to DVMRP or PIM.
upcalls to mrouted	The number of multicast packets sent to mrouted (DVMRP) for learning.
upcall queue overflows	The number of times a packet was unsuccessfully queued for learning. Usually a result of learning many flows simultaneously.
upcalls dropped due to full socket buffer	DVMRP and PIM use a routing socket to communicate with the kernel. If the X-Pedition drops any packets queued for learning as the result of insufficient buffer space on that socket, this counter increments.
cache cleanups	The number of upcalls that timeout before servicing the upcall.
datagrams with no route for origin	The RPF check on this source address yielded no known route. In unicast this would produce an “ICMP host unreachable” message.
datagrams arrived with bad tunneling	When using IP in IP tunneling with DVMRP enabled, this field indicates that the tunneled packet was either corrupted or improperly encapsulated.
datagrams could not be tunneled	The number of packets that could not be tunneled. This is usually the result of the DF bit being set and the MTU of the tunnel interface being smaller than the size of the datagram + encapsulating IP header.
datagrams arrived on wrong interface	The number of packets received on an interface that is not the RPF upstream interface for the flow.
datagrams selectively dropped	
datagrams dropped due to queue overflow	The number of packets dropped as the result of the CPU’s forwarding queue being full.
datagrams dropped for being too large	The number of packets dropped because they were larger than the MTU on the outbound interface, and the DF bit was set.

statistics show framer

Purpose

Display framer statistics.

Format

statistics show framer <*port list*>

Mode

Enable

Parameters

<*port list*>	Specifies the port or group of ports.

Restrictions

None.

statistics show port-errors

Purpose

Display port error statistics.

Format

statistics show port-errors <*port/*SmartTRUNK-*list*>|**all-ports**

Mode

Enable

Parameters

<*port/*SmartTRUNK-*list*>
Specifies a specific port or SmartTRUNK list.

all-ports
Display port error statistics for all physical and logical ports.

Restrictions

None.

Example

To display port error statistics on port et.2.1:

```
xp# statistics show port-errors et.2.1

Port: et.2.1
----
Error Stats                          Error Stats

-----------                          -----------

CRC errors                    0      Carrier sense errors          0

Single collision (tx OK)      0      Many collisions (tx OK)       0

Many collisions (drop)        0      Late collisions               0

Long frames  >1518 bytes      0      Invalid long frames           0

Short frames <64 bytes        0      Alignment errors              0

Deferred transmissions        0      Transmit underruns            0

IP - bad version              0      IP - bad checksum             0

IP - bad header               0      IP - small datagram           0

IP - expand TTL ring          0      IPX - bad header              0

Non-IP/IPX protocol           0      Invalid MAC encap.            0

Internal frame tx error       0      Internal frame rx error       0

Input buffer overflow         0      Packet request overflow       0

Out buffer (low) ovflow       0      Out buffer (med) ovflow       0

Out buffer (high) ovflow      0      Out buffer (ctrl) ovflow      0

Input VLAN drop frame         0
Error stats cleared   * Never Cleared *
```

Field Definitions

Field	Description
CRC errors	Displays the total frames received that are an integral number of octets in length but failed the FCS check.
Single collision (tx OK)	Displays the total number of frames that successfully transmitted after only one collision.
Many collisions (drop)	Displays the total number of frames dropped after more than one collision.
Long frames >1518 bytes	Displays the total number of frames received that exceeded the maximum permitted frame size (1518 bytes) but were otherwise acceptable because they passed FCS checks and were an integral number of octets in length.
Short frames <64 bytes	Displays the total number of frames that received that were less than 64 bytes in length.
Deferred transmissions	Displays the total number of frames for which the first transmission attempt was delayed because the medium was busy. This count does not include frames involved in collisions.
IP - bad version	Displays the total number of IP packets dropped because the IP version was not equal to 4.
IP - bad header	Displays the total number of IP packets dropped because the header length was less than 20 bytes.
IP - expand TTL ring	Displays the total number of IP packets dropped due to expanding TTL.
Non-IP/IPX protocol	Displays the total number of packets dropped due to an unknown or bad Layer-3 protocol.
Internal frame tx error	Displays the total number of transmit frames dropped due to an OWB resync or internal transmit error.
Input buffer overflow	Displays the total number of frames dropped due to the IPP Interface Buffer full condition.
Out buffer (low) ovflow	Displays the total number of frames dropped because the low priority Output Packet Manager was full.
Out buffer (high) ovflow	Displays the total number of frames dropped because the high priority Output Packet Manager was full.
Input VLAN drop frame	Displays the total number of frames dropped due to a VLAN table.

Field	Description
Carrier sense errors	Displays the number of times that the carrier sense condition was lost or never asserted when attempting to transmit a frame on a particular interface. This count increments at most once per transmission attempt, even if the carrier sense condition fluctuates during the attempt.
Many collisions (tx OK)	Displays the total number of frames successfully transmitted after more than one collision.
Late collisions	Displays the total number of times a collision was detected on a particular interface later than 512 bit-times into the transmission of a packet.
Invalid long frames	Displays the total number of frames received that exceeded the 1518 byte size limit and were dropped because they were not an integral number of octets in length and/or failed FCS checks.
Alignment errors	Displays the total number of frames received that are not an integral number of octets in length and do not pass the FCS check.
Transmit underruns	Displays the total number of frames dropped due to transmission underruns. This is a normal, self-correcting condition, but large numbers of underruns may indicate a problem.
IP - bad checksum	Displays the total number of IP packets received with a bad checksum.
IP - small datagram	Displays the total number of IP packets received without a full header (payload too small).
IPX - bad header	Displays the total number of IPX packets received with too small of a header length.
Invalid MAC encap.	Displays the total number of frames dropped due to a bad or unknown MAC encapsulation.
Internal frame rx error	Displays the total number of received frames dropped due to reasons not accounted for in other counts.
Packet request overflow	Displays the total number of packet request overflows.
Out buffer (med) ovflow	Displays the total number of frames dropped because the medium priority Output Packet Manager was full.
Out buffer (ctrl) ovflow	Displays the total number of frames dropped due to the control priority Output Packet Manager being full.
Error stats cleared	Shows the date and time when the port-error stats were last cleared.

statistics show port-packets

Purpose

Display port packet statistics.

Format

statistics show port-packets <*port-list*>|**all-ports**

Mode

Enable

Parameters

<*port-list*>|**all-ports**

Specifies the port. Specify **all-ports** to display port packet statistics for all physical and logical ports.

Restrictions

None.

Example

To display port packet statistics on port et.2.1:

```
xp# statistics show port-packets et.2.1

Port: et.2.1
----
RMON Stats              Received            Transmitted

----------              --------            -----------

Unicast frames             0                    0

Multicast frames           0                    0

Broadcast frames           0                    0

64 byte frames             0                    0

65-127 byte frames         0                    0

128-255 byte frames        0                    0

256-511 byte frames        0                    0

512-1023 byte frames       0                    0

1024-1518 byte frames      0                    0

RMON stats cleared   * Never Cleared *
```

statistics show port-stats

Purpose

Display normal (non-error) port statistics.

Format

statistics show port-stats <*port/*SmartTRUNK-*list*>|**all-ports**

Mode

Enable

Parameters

<*port/*SmartTRUNK-*list*>|**all-ports**

Specifies a specific port or SmartTRUNK list.

all-ports Display port statistics for all physical and logical ports.

Note: For additional information on gathering statistics on SmartTRUNKs, see *smarttrunk show on page 1047*.

Restrictions

None.

Example

Port Statistics

To display port statistics on port et.2.1:

```
xp# statistics show port-stats et.2.1

Port: et.2.1
-------------
Port Stats                      Received          Transmitted

----------                      --------          -----------

Frames/Packets                     0                 0

. Switched frames (bridging)       0                 0

. Local frames (bridging)          0                 N/A
. Routed packets                   0                 0

  . Switched (data)                0                 N/A
  . Consumed by CPU                0                 N/A
Bytes                              0                 0

. Bridged bytes                    0                 0

. Routed bytes                     0                 0

L2 table misses                    0                 N/A
IP table misses                    0                 N/A
IPX table misses                   0                 N/A
IP TTL expirations                 0                 N/A
IPX TC expirations                 0                 N/A
1 minute traffic rates
. Average bits/sec                 0                 0

. Packet discards                  0                 0

. Packet errors                    0                 0

. Unicast packets                  0                 0

. Multicast packets                0                 0

. Broadcast packets                0                 0

Port stats cleared  * Never Cleared *
```

Port Statistics Field Definitions

Field	Description
Frames/Packets	Shows the total number of frames received/transmitted on this port.
Switched frames	Shows the number of frames that have been bridged or forwarded.
Local frames	Shows the number of local frames (frames destined for a port that is the same as the port of entry) that was dropped.
Routed packets	Shows the total number of frames routed on this port.
Switched (data)	Shows the number of packets that was forwarded by the hardware.
Consumed by CPU	Shows the number of packets that was sent to the control module to be forwarded.
Bytes	Shows the total number of bytes received/transmitted on this port.
Bridged bytes	Shows the number of total bytes that has been bridged.
Routed bytes	Shows the number of total bytes that has been routed.
L2 table misses	Shows the number of times that a Layer-2 frame could not be resolved by the L2 Table.
IP table misses	Shows the number of times that an IP packet could not be resolved by the IP Routing Table.
IPX table misses	Shows the number of times that an IPX packet could not be resolved by the IPX Routing Table.
IP TTL expirations	Shows the number of IP packets that have been received by the port with a Time-to-Live (TTL) header with a value of 1. The IP packet will then be expired at this point.
IPX TC expirations	Shows the number of IPX packets that have been received by the port with a TC header with a value of 1. The IPX packet will then be expired at this point.
Average bits/sec	Shows an average traffic rate in bits/second for a one-minute time period for a port.
Packet discards	Shows the number of packets discarded by a port within a one-minute time period.
Packet errors	Shows the number of packets containing errors that was seen by the port within a one-minute time period.

Field	Description
Unicast packets	Shows the number of unicast packets that was seen by the port within a one-minute time period.
Multicast packets	Shows the number of multicast packets that was seen by the port within a one-minute time period.
Broadcast packets	Shows the number of broadcast packets that was seen by the port within a one-minute time period.
Port stats Cleared	Shows the date and time when the port stats were last cleared.

statistics show rarp

Purpose

Display reverse ARP statistics.

Format

statistics show rarp <*string*>

Mode

Enable

Parameters

<*string*>|**all**
Specifies the interface name. Specify **all** to display reverse ARP statistics for all interfaces.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

None.

Example

To display reverse ARP statistics on interface ‘en0’:

```
xp# statistics show rarp en0

Interface en0:
      0 requests received
      0 replies sent
      0 requests received on interface with rarpd disabled
      0 requests received that failed sanity check
      0 requests received that did not result in a match
      Last 5 Requests Received
      ----- no rarp requests received -----
      Last 5 Replies Sent
      ----- no rarp replies sent -----
```

statistics show summary-stats

Purpose

Display recent traffic summary statistics.

Format

statistics show summary-stats

Mode

Enable

Parameters

None.

Restrictions

None.

statistics show tcp

Purpose

Display Transmission Control Protocol (TCP) statistics for all packets received or sent by the router software.

Format

statistics show tcp

Mode

Enable

Parameters

None.

Restrictions

IP statistics on hardware-routed IP flows are not included in the statistics displayed by this command. To see statistics collected by router hardware, use the **statistics show ip-interface** and **statistics show port-packets** commands.

Example

To display TCP statistics:

```
xp# statistics show tcp
tcp:
        235 packets sent
                232 data packets (22777 bytes)
                1 data packet (494 bytes) retransmitted
                0 resends initiated by MTU discovery
                2 ack-only packets (5 packets delayed)
                0 URG only packets
                0 window probe packets
                0 window update packets
                0 control packets
        320 packets received
                227 acks (for 22776 bytes)
                3 duplicate acks
                0 acks for unsent data
                158 packets (185 bytes) received in-sequence
                0 completely duplicate packets (0 bytes)
                0 old duplicate packets
                0 packets with some dup. data (0 bytes duped)
                0 out-of-order packets (0 bytes)
                0 packets (0 bytes) of data after window
                0 window probes
                0 window update packets
                0 packets received after close
                0 discarded for bad checksums
                0 discarded for bad header offset fields
                0 discarded because packets too short
        0 connection requests
        1 connection accept
        1 bad connection attempt
        0 listen queue overflows
        1 connection established (including accepts)
        0 connections closed (including 0 drops)
                0 connections updated cached RTT on close
                0 connections updated cached RTT variance on close
                0 connections updated cached ssthresh on close
        0 embryonic connections dropped
        226 segments updated rtt (of 228 attempts)
        0 retransmit timeouts
                0 connections dropped by rexmit timeout
        0 persist timeouts
                0 connections dropped by persist timeout
        0 keepalive timeouts
                0 keepalive probes sent
                0 connections dropped by keepalive
        0 correct ACK header predictions
        88 correct data packet header predictions
```

Field Definitions

Field	Description
packets sent	Total number of TCP packets sent by the router.
data packets (bytes)	Number of data packets (bytes) sent.
data packet (bytes) retransmitted	Number of data packets (bytes) retransmitted due to congestion.
resends initiated by MTU discovery	Number of packets resent due to MTU discovery.
ack-only packets (5 packets delayed)	Number of ACK-only packets sent (Number of delayed ACK packets sent)
URG only packets	Number of URG-only packets sent. The URG (Urgent Pointer) bit indicates that urgent data can be found in the TCP header.
window probe packets	Number of packets sent for probing TCP window size.
window update packets	Number of packets sent for updating TCP window size.
control packets	Number of control (SYN|FIN|RST) packets sent.
packets received	Total number of TCP packets received by the router.
acks (bytes)	Number of ACK packets (bytes) received.
duplicate acks	Number of duplicate ACK packets received.
acks for unsent data	Number of ACK packets received for unsent data.
packets (bytes) received in-sequence	Number of data packets (bytes) received in sequence.
completely duplicate packets (bytes)	Number of completely duplicate packets (bytes) received.
old duplicate packets	Number of old duplicate packets dropped by PAWS (*Protect Against Wrapped Sequence number*) mechanism as described in RFC1323.
packets with some dup. data (bytes duped)	Number of packets (bytes) with partially duplicate data.
out-of-order packets (bytes)	Number of out-of-order packets (bytes) received.
packets (bytes) of data after window	Number of packets (bytes) received with data after window.

Field	Description
window probes	Number of packets received for probing TCP window size.
window update packets	Number of packets received for updating TCP window size.
packets received after close	Number of packets received after the TCP connection closed.
discarded for bad checksums	Number of packets received that were dropped due to checksum errors.
discarded for bad header offset fields	Number of packets received that were dropped because of bad header offsets.
discarded because packets too short	Number of packets received that were dropped because they were too short.
connection requests	Number of TCP connection requested.
connection accept	Number of TCP connection accepted.
bad connection attempt	Number of bad connection attempts (e.g., those with premature acknowledgments).
listen queue overflows	Number of connection requests dropped due to listen queue overflows.
connection established (including accepts)	Number of TCP connections established.
connections closed (including drops)	Number of TCP connections closed (including number of connections dropped).
connections updated cached RTT on close	Number of times the cached RTT (round-trip time) was updated.
connections updated cached RTT variance on close	Number of times the cached RTT variance was updated.
connections updated cached ssthresh on close	Number of times the cached ssthresh variable was updated.
embryonic connections dropped	Number of embryonic connections dropped.
segments updated rtt (of attempts)	Number of successful RTT updates from segments (number of attempted updates from segments).
retransmit timeouts	Number of retransmission timeouts.
connections dropped by rexmit timeout	Number of connections dropped due to retransmission timeouts.
persist timeouts	Number of persistence timeouts.

Field	Description
connections dropped by persist timeout	Number of connections dropped due to persistence timeouts.
keepalive timeouts	Number of keepalive timeouts.
keepalive probes sent	Number of keepalive probes sent.
connections dropped by keepalive	Number of connections dropped due to keepalive timeout
correct ACK header predictions	Number of correct header predictions made by the router for ACK packets.
correct data packet header predictions	Number of correct header predictions made by the router for data packets.

statistics show udp

Purpose

Display User Datagram Protocol (UDP) statistics for all packets received or sent by the router software.

Format

statistics show udp

Mode

Enable

Parameters

None.

Restrictions

IP statistics on hardware-routed IP flows are not included in the statistics displayed by this command. To see statistics collected by router hardware, use the **statistics show ip-interface** and **statistics show port-packets** commands.

Example

To display UDP statistics:

```
xp# statistics show udp
udp:
        0 datagrams received
        0 datagrams with incomplete header
        0 datagrams with bad data length field
        0 datagrams with bad checksum
        0 datagrams dropped due to no socket
        0 broadcast/multicast datagrams dropped due to no socket
        0 datagrams dropped due to full socket buffers
        0 datagrams not for hashed pcb
        0 delivered
        0 datagrams output
```

Field Definitions

Field	Description
datagrams received	Total number of UDP datagrams received.
datagrams with incomplete header	Number of datagrams dropped due to incomplete header.
datagrams with bad data length field	Number of datagrams dropped due to bad data lengths.
datagrams with bad checksum	Number of datagrams dropped due to checksum errors.
datagrams dropped due to no socket	Number of datagrams dropped because there was no socket.
broadcast/multicast datagrams dropped due to no socket	Number or broadcast/multicast datagrams dropped because there was no socket.
datagrams dropped due to full socket buffers	Number of datagrams dropped because of full socket buffers.
datagrams not for hashed pcb	Number of input datagrams received that are not for hashed pcb.
delivered	Total number of datagrams received that were not dropped.
datagrams output	Total number of datagrams sent.

statistics show most-active

Purpose

Display active tasks.

Format

statistics show most-active

Mode

Enable

Parameters

None.

Restrictions

None.

Example

To display active tasks:

```
xp# statistics show most-active

Timestamp: 2000-04-25 17:56:32
CPU Idle : 98%   (since system startup 441751425.0 sec ago)
NAME            USAGE %      RELATIVE %
--------        -------       ----------
STP_T            0.2           47.65
PHY_POLL         0.0           17.57
L2_AGE_T         0.0           7.90
L3_AGE_T         0.0           7.10
IPC              0.0           4.60
CONS_T           0.0           4.25
STATS_T          0.0           3.96
TNTASK           0.0           2.41
SYSTEM H         0.0           0.88
HBT_T            0.0           0.82
SNMP             0.0           0.67
GATED            0.0           0.58
IPXROUTE         0.0           0.48
CONS2T           0.0           0.33
LOWEST           0.0           0.25
PPP_TASK         0.0           0.24
PINGER_T         0.0           0.11
L2_LRN_T         0.0           0.07
CDP_T            0.0           0.02
LGRP_T           0.0           0.00
MPS              0.0           0.00
TNETD            0.0           0.00
ETHH             0.0           0.00
NI H             0.0           0.00
ARP_T            0.0           0.00
HSWAP            0.0           0.00
IPRED_T          0.0           0.00
SYS_TK           0.0           0.00
SNMP_CF          0.0           0.00
WAN_TOD_         0.0           0.00
DHCP             0.0           0.00
BOUNCE           0.0           0.00
IP_T             0.0           0.00
IPX_T            0.0           0.00
PHX_T            0.0           0.00
NTP              0.0           0.00
ERROR_LO         0.0           0.00
L3_ACL_T         0.0           0.00
MCAST            0.0           0.00
PROFILE          0.0           0.00
PRI_L3MD         0.0           0.00
L3_RL_T          0.0           0.00
```

statistics show vlan

Purpose

Display the per-VLAN-packet statistics for IP-unicast traffic.

Format

statistics show vlan all|<*string*>

Mode

Enable

Description

The **statistics show vlan** command shows IP-unicast packet statistics for all VLANs in the router or for a specific VLAN.

Parameters

all
Display statistics for all VLANs.

<string>
Display statistics for a specific VLAN.

Restrictions

L4-bridging and RMON Pro must be enabled on the VLAN and the VLAN ports in order to collect per-VLAN-statistics. “N/A” will appear if the VLAN does not satisfy these restrictions.

Example

To display UDP statistics:

```
xp# statistics show vlan

  VID    VLAN Name      Total Packets      Total Bytes
  ---     ---------      -------------      -----------
  1       DEFAULT        N/A                N/A
  5       red            1007114            81576234
  6       blue           1004947            81400
```

Chapter 66

stp Commands

The **stp** commands let you display and change settings for the default Spanning Tree.

Command Summary

Table 52 lists the **stp** commands. The sections following the table describe the command syntax.

Table 52. stp commands

stp enable port <*port-list*>
stp set bridging [**forward-delay** <*num*>**] [hello-time** <*num*>**] [max-age** <*num*>**] [priority** <*num*>**]**
stp set port <*port-list*> **priority** <*num*> **port-cost** <*num*> **point-to-point [ForceTrue| ForceFalse| Auto] edge-port [True| False]**
stp show bridging-info
stp reset-rstp port <*port-list*> **|all-ports**
stp set protocol-version rstp
stp filter-bpdu <*port-list*> **|all-ports**

Note: The X-Pedition supports STP over POS, but does *not* support PVST over POS.

stp enable port

Purpose

Enable STP on one or more ports.

Format

stp enable port <*port-list*>

Mode

Configure

Description

The **stp enable port** command enables STP on the specified ports.

Parameters

<*port-list*> The ports on which you are enabling STP. You can specify a single port or a comma-separated list of ports. Example: et.1.3,et.(1-3).(4,6-8).

Restrictions

None

stp set bridging

Purpose

Set STP bridging parameters.

Format

stp set bridging [**forward-delay** <*num*>**] [hello-time** <*num*>**] [max-age** <*num*>**]**
[priority <*num*>**]**

Mode

Configure

Description

The **stp set bridging** command lets you configure the following STP parameters:

- Bridging priority
- Hello time
- Maximum age
- Forward delay

Parameters

forward-delay <*num*>
Sets the STP forward delay for the X-Pedition. The forward delay is measured in seconds. Specify a number from 4–30. The default is 15.

hello-time <*num*>
Sets the STP hello time for the X-Pedition. The hello time is measured in seconds. Specify a number from 1–10. The default is 2.

max-age <*num*>
Sets the STP maximum age for the X-Pedition. Specify a number from 6–40. The default is 20.

priority <*num*>
Sets the STP bridging priority for the X-Pedition. Specify a number from 0–65535. The default is 32768.

Restrictions

None.

Examples

To set the bridging priority of Spanning Tree for the entire X-Pedition to 1:

```
xp(config)# stp set bridging priority 1
```

stp set port

Purpose

Set STP port priority and port cost for ports.

Format

stp set port <*port-list*> **priority** <*num*> **port-cost** <*num*>
point-to-point [ForceTrue| ForceFalse| Auto] edge-port [True| False]

Mode

Configure

Description

The **stp set port** command sets the STP priority and port cost for individual ports.

Parameters

port <*port-list*>
The port(s) for which you are setting STP parameters. You can specify a single port or a comma-separated list of ports. Example: et.1.3,et.(1-3).(4,6-8).

priority <*num*>
The priority you are assigning to the port(s). Specify a number from 0– 16 inclusive. The default is 8.

port-cost <*num*>
The STP cost you are assigning to the port(s). Specify a number from 1– 65535. The default depends on the port speed: 1 for Gigabit (100-Mbps) ports, 10 for 100-Mbps ports, and 100 for 10-Mbps ports.

point-to-point [ForceTrue|ForceFalse|Auto]
Specify a point-to-point or a non-point-to-point link administratively. The default setting is ‘Auto.’

edge-port [True|False]
Specify whether the port(s) should be initialized as a edge port or a non-edge port. The default is ‘False.’

Restrictions

With the introduction of the ER16, an X-Pedition router can support up to 480 ports—this exceeds the 256-port limit allowed by the 8-bit port number field specified in the IEEE 802.1D-1998 standard. To accommodate the increase in the number of supported ports, Enterasys extended the

port field to a 12-bit value and decreased the port priority field to a 4-bit value. As a result, the X-Pedition allows STP or PVST port configurations with a priority of 0 to 15 only. In spite of these changes, the X-Pedition remains compatible with other switches.

stp show bridging-info

Purpose

Display STP bridging information.

Format

stp show bridging-info

Mode

Enable

Description

The **stp show bridging-info** command displays STP bridging information for the X-Pedition.

Parameters

None.

Restrictions

None.

stp reset-rstp

Purpose

Reset RSTP.

Format

stp reset-rstp port <*port_list*> **|all ports**

Mode

Enable

Description

The **stp reset-rstp** command resets the point-to-point and edge port parameters to the user-specified values and forces the specified ports to send RSTP BPDU's until a version 0 STP BPDU is received.

Parameters

<*port_list*>	Specifies the ports for which you want to reset RSTP.
all-ports	The all-ports keyword resets RSTP for all the X-Pedition ports.

Restrictions

None.

stp set protocol-version rstp

Purpose

Enable rapid reconfiguration on default spanning tree.

Format

stp set protocol-version rstp

Mode

Configure.

Description

The **stp set protocol-version** command changes the STP version from “STP compatible” (version 0) to “Rapid Configuration” (version 2).

Restrictions

STP cannot be enabled on any non-LAN ports when running RSTP.

stp filter-bpdu

Purpose

Filter out BPDU on a port when STP is disabled.

Format

stp filter-bpdu <*port-list*> |**all-ports**

Mode

Configure.

Description

The **stp filter-bpdu** command sets up a filter on the specified port for BPDU's when STP is disabled.

Parameters

<*port-list*>	List of ports to which you will apply the filter.
all-ports	All ports.

Restrictions

Can be used only when STP is disabled.

Chapter 67

system Commands

The **system** commands let you display and change system parameters.

Command Summary

Table 53 lists the **system** commands. The sections following the table describe the command syntax.

Table 53. system commands

system are-promimage upgrade module <*number*> **[tftp-server** <*IPaddr-or-hostname*> **filename** <*filename*>**]	[tftp-url** <*URL*>**]**
system disable inputportlevel-rate-limiting slot <*numbers*>
system enable aggregate-rate-limiting slot <*number*>
system failover master-cm
system hotswap out|in slot <*number*>
system image add {tftp-server <*IPaddr-or-hostname*> **filename** <*filename*>| **tftp-url** <*URL*>**} [destination {backup-cm| primary-cm| all}]**
system image choose <*filename*> **[backup-cm| primary-cm| all]**
system image delete <*filename*> **[backup-cm| primary-cm| all]**
system image list [primary-cm| backup-cm| all]
system kill ssh-session <*session-id*>
system kill telnet-session <*session-id*>
system l3-deep-buckets module <*num*> **set on**

Table 53. system commands (Continued)

system promimage upgrade tftp-server <*IPaddr-or-hostname*> **file-name** <*filename*> **{destination primary-cm| backup-cm}|** **tftp-url** <*URL*> **{destination primary-cm | backup-cm}|** **file-name {tftp-server** <*IPaddr-or-hostname*>**| destination primary-cm| backup-cm}|** **destination primary-cm | backup-cm {tftp-server** <*IPaddr-or-hostname*> **filename** <*filename*>**}| {filename** <*filename*> **tftp-server** <*IPaddr-or-hostname*>**}| {tftp-url** <*URL*>**}**
system set backup-cm-timeout seconds <*seconds*>
system set bootprom netaddr <*IPaddr*> **netmask** <*IPnetmask*> **tftp-server** <*IPaddr*> **[tftp-gateway** <*IPaddr*>**]**
system set buffs-in-normal-mode low <*number*> **medium** <*number*> **high** <*number*>
system set buffs-in-recv-ctrl-mode low <*number*> **medium** <*number*> **high** <*number*> **dynamic**
system set cntrl-only-mode-count-per-min <*number*>
system set cntrl-pkts-threshold <*number*>
system set console level <*level*> **use-syslog-levels**
system set contact <*system-contact*>
system set cpu-utilization-trap min-threshold <*value*> **max-threshold** <*value*>
system set data-pkts-threshold <*number*>
system set date year <*year*> **month** <*month*> **day** <*day*> **hour** <*hour*> **min** <*min*> **second** <*sec*>
system set dns server <*IPaddr*>**[,**<*IPaddr*>**[,**<*IPaddr*>**]]** domain <*name*>
system set dst-changing [s-wk <*num*>**] [s-dow** <*num*>**] [s-mo** <*num*>**] [s-hr** <*num*>**] [s-min** <*num*>**] [e-wk** <*num*>**] [e-dow** <*num*>**] [e-mo** <*num*>**] [e-hr** <*num*>**] [e-min** <*num*>**]|dst-fixed [s-mo** <*num*>**] [s-day** <*num*>**] [s-hr** <*num*>**] [s-min** <*num*>**] [e-mo** <*num*>**] [e-day** <*num*>**] [e-hr** <*num*>**] [e-min** <*num*>**]|dst-manual**
system set dst-fixed [s-mo <*num*>**] [s-day** <*num*>**] [s-hr** <*num*>**] [s-min** <*num*>**] [e-mo** <*num*>**] [e-day** <*num*>**] [e-hr** <*num*>**] [e-min** <*num*>**]|dst-manual**
system set dst-manual
system set extended-debug [**inhibit-master-reboot**] | [**enable-intr-monitor** <*num*>] | [**enable-pkt-capture**]
system set high-priority-pad <*number*>
system set idle-timeout serial| telnet| ssh <*timeout*>
system set ifqlen-to-xmit-pkts <*number*>
system set ip-wakeup-intvl <*seconds*>

Table 53. system commands (Continued)

system set ipx-wakeup-intvl <*seconds*>
system set lgrp-pkts-threshold <*number*>
system set location <*location*>
system set login-banner <*string*>|**none**|**file-name** <*string*>
system set low-priority-pad <*number*>
system set malloc
system set max-packets-per-interrupt <*number*>
system set max-pkts-in-recv-ctrl-only <*number*>
system set med-priority-pad <*number*>
system set name <*system-name*>
system set ni-driver-debug
system set password <*mode*> <*string*>|**none**
system set password-policy auth-grace-timeout <*grace*>| **change-after-first-login {on|off}**| **expire-warning** <*warn*>| **history-size** <*size*>| **lifetime** <*time*>| **login-failure-grace-time** <*fail-time*>| **maximum-failed-logins** <*fails*>| **minimum-length** <*minimum*> **verification {on | off}**
system set poweron-selftest [on| quick]
system set show-config alphabetical
system set spooler-memory-limit <*memory-limit*>
system set stp-pkts-threshold <*number*>
system set syslog [**server** <*hostname-or-IPaddr*>] [**local**] [**level** <*level-type*>] [**facility** <*syslog-facility-type*>] [**source** <*source-IPaddr*>] [**buffer-size** <*size*>]
system set syslog-levels <*facility*> **level** <*level*>
system set terminal baud <*baud-rate*>| **columns** <*number*> **rows** <*number*>
system set tftpsource <*IP address*>
system set timezone <*timezone*>|<*minutes*>
system set user <*username*> <*new-password*> **password-option [never-expires| normal] privilege-level [login| enable] status [always-enabled | disabled| enabled]**
system show <*system-parm*>
system show capacity all | chassis | task | cpu | memory
system show syslog levels

system are-promimage upgrade

Purpose

Upgrade the boot prom on a specific ARE module.

Format

system are-promimage upgrade module <*number*> **[tftp-server** <*IPaddr-or-hostname*> **filename** <*filename*>**] | [tftp-url** <*URL*>**]**

Mode

Enable

Description

The **system set data-pkts-threshold** command allows you to upgrade the boot prom on a specific ARE module.

Parameters

module <*number*> The module number of the ARE module to upgrade.

<*IPaddr-or-hostname*>
The IP address or host name of the TFTP server. The tftp-url is not allowed when using this parameter.

<*filename*>
The name of the software image file—required when using the tftp-server option.

<URL>
The TFTP URL (e.g., tftp://10.1.2.3/images/img.tar.gz). The tftp-server is not allowed when using this parameter.

Restrictions

None.

Examples

To use the tftp-server parameter to download the boot prom image file “are.tar.gz” from the TFTP server 10.1.2.3 to a specific ARE module:

```
xp# system are-promimage upgrade module 3 tftp-server 10.1.2.3 filename are.tar.gz
```

To use the tftp-url parameter to download the boot prom image file “are.tar.gz” from the TFTP server 10.1.2.3 to a specific ARE module:

```
xp# system are-promimage upgrade module 3 tftp-url tftp://10.1.2.3/images/are.tar.gz
```

system disable inputportlevel-rate-limiting slot

Purpose

Disables Input Port Level Rate Limiting on a specific slot(s) and allows aggregate rate limiting policies to use the credit buckets reserved for port-level rate limiting policies.

Format

system disable inputportlevel-rate-limiting slot <*numbers*>

Mode

Configure

Description

The **system disable inputportlevel-rate-limiting** command disables Input Port Level Rate Limiting on a specific slot(s) and makes the credit buckets reserved for port-level rate limiting policies available for aggregate rate limiting policies.

Parameters

slot <*numbers*> The occupied slot or list of slots.

Restrictions

None.

system enable aggregate-rate-limiting

Purpose

Enables Input Port Level and Aggregate Rate Limiting.

Format

system enable aggregate-rate-limiting slot <*numbers*>

Mode

Configure

Description

The **system enable aggregate-rate-limiting** command enables port level and aggregate rate limiting features on the router. There are two modes of operation for rate limiting available on the X-Pedition: per-flow rate limiting and aggregate rate limiting. By default, the per-flow rate limiting mode is enabled.

By using this command, you are disabling per-flow rate limiting and enabling aggregate rate limiting and port level rate limiting.

To revert back to per-flow rate limiting, negate this command.

Parameters

slot <*numbers*> The slot numbers you wish to disable.

Restrictions

Aggregate and flow-aggregate rate limiting are not supported on 802.1q trunk ports.

Example

To enable aggregate rate limiting:

```
xp# system enable aggregate-rate-limiting slot 1
```

system failover master-cm

Purpose

Force a failover from the master to backup Control Module.

Format

system failover master-cm

Mode

Enable

Description

The system failover master-cm command allows you to force a failover from the master CM to the backup CM. With the backup CM acting as the master CM, you can upgrade the boot firmware. Refer to the *Enterasys X-Pedition User Reference Manual* for details.

Note: In a dual Control Module configuration, the MAC address of the Primary Control Module in slot “CM/0” is used for both Control Modules after the system is booted. If the Control Module in slot “CM/0” is removed and not replaced after a fail-over, or if it is replaced with a new Control Module and the system is rebooted, the system will use the MAC address of the Control Module in slot 1 (i.e., the new Control Module).

Parameters

None

Restrictions

None

system hotswap

Purpose

Activates or deactivates a line card.

Format

system hotswap out|in slot <*number*>

Mode

Enable

Description

The **system hotswap out** command deactivates a line card in a specified slot on the X-Pedition, causing it to go offline. The command performs the same function as if you had pressed the Hot Swap button on the line card.

The **system hotswap in** command causes a line card that was deactivated with the **system hotswap out** command to go online again. The command performs the same function as if you had removed the card from its slot and inserted it again.

See the *Enterasys X-Pedition User Reference Manual* for more information on hot swapping line cards.

Parameters

out
Causes the line card in the specified slot to be deactivated.

in
Causes an inactive line card in the specified slot to be reactivated.

Note: The **system hotswap in** command works only on a line card that was deactivated with the **system hotswap out** command.

slot <*number*>
Is the slot where the line card resides. Specify any number between 1-16.

Restrictions

None.

Example

To deactivate the line card in slot 7 on the X-Pedition:

```
xp# system hotswap out slot 7
```

system image add

Purpose

Copy a system software image to the X-Pedition.

Format

system image add {tftp-server <*IPaddr-or-hostname*> **filename** <*filename*>| **tftp-url** <*URL*>**} [destination {backup-cm| primary-cm| all}]**

Mode

Enable

Description

The **system image add** command copies a system software image from a TFTP server into the PCMCIA Flash Module on the Control Module. By default, if the X-Pedition has two Control Modules, the system software image is copied to both Control Modules.

Note: The X-Pedition supports PCMCIA Flash Modules obtained from Enterasys Networks only. For information regarding the PCMCIA Virtual File systems VFS1 and VFS2, see the *Enterasys X-Pedition User Reference Manual*.

Parameters

<*IPaddr-or-hostname*>
The IP address or host name of the TFTP server. The tftp-url is not allowed when using this parameter.

<*filename*>
The name of the system software image file—required when using the tftp-server option.

<*URL*>
The TFTP URL (e.g., tftp://10.1.2.3/images/img.tar.gz). The tftp-server is not allowed when using this parameter.

destination {backup-cm| primary-cm| all}

Note: When a user selects the **primary-cm** or **backup-cm** option, the router will prompt the user about adding the image to *both* CMs.

primary-cm
Copies the system software image to the *Primary* Control Module only.

backup-cm
Copies the system software image to the *Backup* Control Module only.

all
Select this option to add the image to all CMs.

Restrictions

None.

Examples

In the following examples, the image file img.tar.gz is located in a folder in the root directory named images on the TFTP server 10.1.2.3.

To use the *tftp-server* parameter to download the software image file "img.tar.gz" from the TFTP server 10.1.2.3 to the both control modules:

```
xp# system image add tftp-server 10.1.2.3 filename img.tar.gz destination primary-cm
xp# system image add tftp-server 10.1.2.3 filename img.tar.gz destination backup-cm
xp# system image add tftp-server 10.1.2.3 filename img.tar.gz
```

To use the *tftp-url* parameter to download the software image file "images/img.tar.gz" from the TFTP server 10.1.2.3 to both control modules:

```
xp# system image add tftp-url tftp://10.1.2.3/images/img.tar.gz destination primary-cm
xp# system image add tftp-url tftp://10.1.2.3/images/img.tar.gz destination backup-cm
xp# system image add tftp-url tftp://10.1.2.3/images/img.tar.gz
```

system image choose

Purpose

Select a system software image file.

Format

system image choose <*filename*> **[backup-cm| primary-cm| all]**

Mode

Enable

Description

The **system image choose** command specifies the system software image file on the PCMCIA Flash Module that you want the X-Pedition to use the next time the system reboots.

Note: The X-Pedition supports PCMCIA Flash Modules obtained from Enterasys Networks only. For information regarding the PCMCIA Virtual File systems VFS1 and VFS2, see the *Enterasys X-Pedition User Reference Manual*.

Parameters

<*filename*>	The name of the system software image file.
primary-cm	This parameter specifies that the image file is chosen for the primary control module.
backup-cm	This parameter specifies that the image file is chosen for the backup control module.
all	Select this option to use the image file specified as the next boot image on both Control Modules.

Restrictions

None.

system image delete

Purpose

Deletes a system software image file from the PCMCIA Flash Module.

Format

system image delete <*filename*> **[backup-cm| primary-cm| all]**

Mode

Enable

Description

The **system image delete** command deletes a system software image file from the PCMCIA Flash Module on the Control Module.

Note: The X-Pedition supports PCMCIA Flash Modules obtained from Enterasys Networks only. For information regarding the PCMCIA Virtual File systems VFS1 and VFS2, see the *Enterasys X-Pedition User Reference Manual*.

Parameters

<*filename*>	The name of the system software image file you want to delete.
primary-cm	This parameter deletes the image file from the *Primary* Control Module only.
backup-cm	This parameter deletes the image file from the *Backup* Control Module only.
all	Select this option to delete the image file from both Control Modules. The router will prompt users before removing the image.

Restrictions

None.

system image list

Purpose

Lists the system software image files on the PCMCIA Flash Module.

Format

system image list [primary-cm| backup-cm| all]

Mode

Enable

Description

The **system image list** command lists the system software image files contained on the PCMCIA Flash Module on the Control Module.

Note: The X-Pedition supports PCMCIA Flash Modules obtained from Enterasys Networks only. For information regarding the PCMCIA Virtual File systems VFS1 and VFS2, see the *Enterasys X-Pedition User Reference Manual*.

Parameters

primary-cm	This parameter lists the image files on the *Primary* Control Module.
backup-cm	This parameter lists the image files on the *Backup* Control Module.
all	Select this option to display all the images on both Control Modules. If you do not specify any of these options, this command defaults to **all**.

Restrictions

None.

system kill ssh-session

Purpose

Terminates an active Secure Shell session.

Format

system kill ssh-session <*session-id*>

Mode

Enable.

Description

Terminates the active SSH session specified. Use the **system show users** command to get the ID of the session you want to terminate.

Parameter

<*session-id*> The ID (0-3) of the session to terminate.

Restrictions

None.

Example

To terminate SSH session 2, enter the following:

```
system kill ssh-session 2
```

system kill telnet-session

Purpose

Kills a specified Telnet session.

Format

system kill telnet-session <*session-id*>

Mode

Enable

Description

The **system kill telnet-session** command kills the Telnet session specified by the session ID. Use the **system show users** command to display the list of current Telnet users and session IDs.

Parameters

<*session-id*>
The Telnet connection slot number, which can be 0, 1, 2, or 3. The **system show users** command displays the session ID number in the first column. You can only specify one session ID per **system kill telnet-session** command.

Restrictions

None.

Example

To show the active Telnet sessions.

```
xp# system show users
Current Terminal User List:
# Login ID      Mode          From            Login Timestamp
- --------      ----          ----            ---------------
              enabled       console          Thu Feb 22 13:07:412001
0             enabled       10.9.0.1         Thu Feb 22 13:07:592001
2             login-prompt  10.9.0.1
3             login-prompt  10.9.0.1
```

Then, to kill Telnet session 2:

```
xp# system kill telnet-session 2
Telnet session 2 (from 10.9.0.1) killed
```

system l3-deep-buckets

Purpose

Enables deep hashing on a specified module.

Format

system l3-deep-buckets module <*num*> **set on**

Mode

Configure

Description

Use the **system l3-deep-buckets** command to enable deep hashing on a specified module.

Deep hashing allows for more than four hash buckets (levels within a particular entry for a hash value) within an entry in the L3 lookup table. Although hashing should provide an even distribution across the lookup table, there is still a possibility that more than four flows may end up at a particular entry in the lookup table.

Allowing for more than four entries through deep hashing will prevent thrashing, but may cause less-than-wirespeed performance due to the extra amount of entries. This is because thrashing will reduce performance to a greater extent than deep hashing. But although deep hashing may result in less-than-wirespeed performance, it still performs much better than if it were thrashing.

Parameters

module <*num*>**|all**
Is a slot number on the X-Pedition. Specify any number between 1 and 16. The hashing algorithm change affects all ports on the line card in the slot. Specify **all** to enable deep hashing on all slots.

on
Enables deep hashing on the module. Negate this command from active configuration to disable l3 deep hashing

Restrictions

None.

Example

To enable deep hashing on slot 7:

```
xp(config)# system l3-deep-buckets module 7 set on
```

system promimage upgrade

Purpose

Upgrades the boot PROM software on primary and secondary Control Modules.

Format

system promimage upgrade tftp-server <*IPaddr-or-hostname*> **file-name** <*filename*> **{destination primary-cm| backup-cm}|**
tftp-url <*URL*> **{destination primary-cm | backup-cm}|**
file-name {tftp-server <*IPaddr-or-hostname*>**| destination primary-cm| backup-cm}|**
destination primary-cm | backup-cm {tftp-server <*IPaddr-or-hostname*> **filename** <*filename*>**}| {filename** <*filename*> **tftp-server** <*IPaddr-or-hostname*>**}| {tftp-url** <*URL*>**}**

Mode

Enable

Description

The **system promimage upgrade** command copies and installs a boot PROM software image from a TFTP server onto the internal memory on the Primary and Backup Control Module. By default the system copies the bootprom image to both the primary and secondary control modules. The boot PROM software image is loaded when you power on the X-Pedition and in turn loads the system software image file.

Parameters

<*IPaddr-or-hostname*>
The IP address or host name of the TFTP server. The tftp-url is not allowed when using this parameter.

<*filename*>
The name of the boot PROM software image file—required when using the tftp-server option.

<*URL*>
The TFTP URL (e.g., tftp://10.1.2.3/images/img.tar.gz). The tftp-server is not allowed when using this parameter.

primary-cm
Copies the system software image only to the primary Control Module.

backup-cm
Copies the system software image only to the backup Control Module.

Note: If you do not specify a control module, the X-Pedition will load the Boot Firmware onto both Control Modules simultaneously.

Restrictions

None.

Example

In the following examples, the boot-prom image named prom-image is located in a folder in the root directory named images on the TFTP server 10.50.89.88.

To load a new boot PROM image onto the Backup Control Module only, enter the following command from Enable mode:

Note: If you do not specify a control module, the XP will load the Boot Firmware onto both control modules simultaneously. To load a new boot PROM image onto both Control Modules using the tftp-url option, enter the following:

```
xp# system promimage upgrade tftp-url tftp://10.50.89.88/qa/prom-upgrade destination primary-cm

Downloading image ‘qa/prom-upgrade’ from host ‘10.50.89.88’
tftp complete
checksum valid. Ready to program.
flash found at 0xbfc00000
erasing...
programming...
verifying...
programming successful.
Programming complete.
system promimage upgrade tftp://10.50.89.88/qa/prom-upgrade destination backup- cm
system promimage upgrade tftp://10.50.89.88/qa/prom-upgrade
```

To load a new boot PROM image onto both Control Modules using the tftp-server option, enter the following:

```
xp# system promimage upgrade tftp-server 10.50.89.88 filename prom-upgrade destination primary-
cm

Downloading image ‘prom-upgrade’ from host ‘10.50.89.88’
tftp complete
checksum valid. Ready to program.
flash found at 0xbfc00000
erasing...
programming...
verifying...
programming successful.
Programming complete.
system promimage upgrade tftp-server 10.50.89.88 filename prom-upgrade
destination backup-cm
system promimage upgrade tftp-server 10.50.89.88 filename prom-upgrade
```

To upgrade the PROM image from a URL:

```
xp# system promimage upgrade 10.136.2.9 /qa/prom-upgrade
xp# system promimage upgrade 10.136.2.9 /qa/prom-upgrade primary-cm
xp# system promimage upgrade 10.136.2.9 /qa/prom-upgrade backup-cm
```

```
xp# system promimage upgrade 10.136.2.9 bp3200

Downloading image 'bp3200' from host '10.136.2.9'
image is a prom upgrade to version 'prom-E3.2.0.0'
%SYS-I-PRIMARY_CM_MSG, TFTP Complete.
%SYS-I-PRIMARY_CM_MSG, Checksum valid. Ready to program.
%HBT-E-NOBACKUPCP, There is no backup module present
%SYS-W-PRIMARY_CM, Warning from Primary CM: Failed to upgrade PROM on backup CM.
%SYS-I-PRIMARY_CM_MSG, Flash Found.
%SYS-I-PRIMARY_CM_MSG, Erasing.
%SYS-I-PRIMARY_CM_MSG, Programming.
%SYS-I-PRIMARY_CM_MSG, Verifying.
%SYS-I-PRIMARY_CM_MSG, Programming Successful.
%SYS-I-PRIMARY_CM_MSG, Programming Complete.
```

system set backup-cm-timeout

Purpose

Set backup-CM timeout value.

Format

system set backup-cm-timeout seconds <*seconds*>

Mode

Configure

Description

The **system set backup-cm-timeout** command sets the amount of time the backup Control Module will use to determine failure of the primary Control Module. If the secondary Control Module does not receive a heartbeat from the primary Control Module for a time equal to or greater than the time-out value, the secondary Control Module takes over as the primary Control Module. Typically, the primary Control Module sends heartbeats to the secondary Control Module at specific intervals. If the primary Control Module becomes too busy to send heartbeats to the secondary Control Module, you can change this interval and extend the timeout.

Parameters

<*seconds*> The number of seconds (4-1000) the backup Control Module waits without receiving a heartbeat from the primary Control Module before taking over as the primary control module. By default, this value is 4 seconds.

Restrictions

None.

system set bootprom

Purpose

Sets parameters for the boot PROM.

Format

system set bootprom netaddr <*IPaddr*> **netmask** <*IPnetmask*>
tftp-server <*IPaddr*> **[tftp-gateway** <*Ipaddr*>**]**

Mode

Configure

Description

The **system set bootprom** command sets parameters to aid in booting the X-Pedition's system software image remotely over the network. You can use this command to set the X-Pedition's IP address, subnet mask, TFTP boot server address, and gateway address.

Note: These parameters apply only to the Control Module's en0 Ethernet interface (labeled "10/100 Mgmt"). This port is a management port only, and is not intended to perform routing.

Parameters

netaddr <*IPaddr*>
The IP address the X-Pedition uses during the boot exchange with the TFTP boot server.

netmask <*IPnetmask*>
The subnet mask the X-Pedition uses during the boot exchange.

tftp-server <*IPaddr*>
The TFTP boot server's IP address.

tftp-gateway <*Ipaddr*>
The gateway that connects the X-Pedition to the TFTP boot server.

Restrictions

None.

Example

The command in the following example configures the X-Pedition to use IP address 10.50.88.2 to boot over the network from TFTP boot server 10.50.89.88.

```
xp(config)# system set bootprom netaddr 10.50.88.2 netmask 255.255.0.0 tftp-server 10.50.89.88
```

system set buffs-in-normal-mode

Purpose

Set the percentage distribution in normal mode.

Format

system set buffs-in-normal-mode low <*number*> **medium** <*number*> **high** <*number*>

Mode

Configure (diagnostic mode)

Description

The **system set buffs-in-normal-mode** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

low <*number*>	Percentage of low buffers.
medium <*number*>	Percentage of medium buffers.
high <*number*>	Percentage of high buffers.

Restrictions

None.

system set buffs-in-recv-ctrl-mode

Purpose

Set the percentage distribution in control receive mode.

Format

system set buffs-in-recv-ctrl-mode low <*number*> **medium** <*number*> **high** <*number*> **dynamic**

Mode

Configure (diagnostic mode)

Description

The **system set buffs-in-recv-ctrl-mode** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

low <*number*>	The percentage of low buffers.
medium <*number*>	The percentage of medium buffers.
high <*number*>	The percentage of high buffers
dynamic	Set the thresholds dynamically.

Restrictions

None.

system set cntrl-only-mode-count-per-min

Purpose

Sets a threshold for readjusting the buffer thresholds, based on the number of control priority switches per minute.

Format

system set cntrl-only-mode-count-per-min <*number*>

Mode

Configure (diagnostic mode)

Description

The **system set cntrl-only-mode-count-per-min** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

<*number*>	The number of control priority switches per minute.

Restrictions

None.

system set cntrl-pkts-threshold

Purpose

Set maximum number of cntrl packets processed before relinquishing the CPU.

Format

system set cntrl-pkts-threshold <*number*>

Mode

Configure (diagnostic mode)

Description

The **system set cntrl-pkts-threshold** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

<*number*>	The maximum number of cntrl packets.

Restrictions

None.

system set console level

Purpose

This command allows users to select the minimum error message severity level to display to the console.

Format

system set console level <*level*> **use-syslog-levels**

Mode

Configure.

Parameters

<*level*> The minimum console message level.

fatal	Display fatal messages only.
error	Display fatal and error messages only.
warning	Display fatal, error, and warning messages only.
audit	Display fatal, error, audit, and warning messages only.
info	Display all messages.

use-syslog-levels

Select this option to apply the level as the default console message severity level and override the level defined by the **system set syslog-levels** command.

Restrictions

None.

system set contact

Purpose

Set the contact name and information for this X-Pedition.

Format

system set contact <*system-contact*>

Mode

Configure

Description

The **system set contact** command sets the name and contact information for the network administrator responsible for this X-Pedition.

Parameters

<*system-contact*>
A string listing the name and contact information for the network administrator responsible for this X-Pedition. If the string contains blanks or commas, you must use the quotation marks around the string. (Example: **"Jane Doe, janed@corp.com, 408-555-5555 ext. 555"**.)

Restrictions

None.

system set cpu-utilization-trap

Purpose

Configure the threshold parameters for sending a CPU threshold exceeded trap.

Format

system set cpu-utilization-trap min-threshold <*value*> **max-threshold** <*value*>

Mode

Configure

Description

The **system set cpu-utilization** command allows you to configure the threshold values used to control the sending of the CPU threshold exceeded trap. The max-threshold value controls the utilization percentage at which a trap is sent. The min-threshold value controls the utilization percentage where the trap sending logic will be armed. When the CPU utilization exceeds the max-threshold value, one trap is sent—no more traps are sent until the CPU utilization falls below the min-threshold value and exceeds the max-threshold value again. If either value is zero, no trap is generated.

Parameters

min-threshold The percentage value (0-99) to reach before *arming* the trap.

max-threshold The utilization percentage value (0-99) to exceed before *generating* the trap.

Restrictions

None

system set data-pkts-threshold

Purpose

Set maximum number of data packets processed before relinquishing the CPU.

Format

system set data-pkts-threshold <*number*>

Mode

Configure (diagnostic mode)

Description

The **system set data-pkts-threshold** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

<*number*>	The maximum number of data packets.

Restrictions

None.

system set date

Purpose

Set the system time and date.

Format

system set date year <*year*> **month** <*month*> **day** <*day*> **hour** <*hour*> **min** <*min*> **second** <*sec*>

Mode

Enable

Description

The **system set date** command sets the system time and date for the X-Pedition. The X-Pedition keeps the time in a battery-backed realtime clock. To display the time and date, enter the **system show date** command.

Parameters

year <*number*>
Four-digit number for the year. (Example: **2001**)

month <*month-name*>
Name of the month. You must spell out the month name. (Example: **March**)

day <*day*>
Number from 1 – 31 for the day.

hour <*hour*>
Number from 0 – 23 for the hour. (The number **0** means midnight.)

minute <*minute*>
Number from 0 – 59 for the hour.

second <*second*>
Number from 0 – 59 for the second.

Restrictions

None.

system set dns

Purpose

Configure the X-Pedition to reach up to three DNS servers.

Format

system set dns server [“]<*IPaddr*> [<*IPaddr*>] [<*IPaddr*>][“] **domain** <*name*>

Mode

Configure

Description

The **system set dns** command configures the X-Pedition to reach up to three DNS servers. You also can specify the domain name to use for each DNS query.

Parameters

[“]<*IPaddr*> [<*IPaddr*>] [<*IPaddr*>][“]
IP address of the DNS server. Specify the address in dotted-decimal notation. You can specify up to three DNS servers separated by single spaces in the command line.

Note: If you specify more than one IP address, you must surround the IP address specification with a set of quotes.

<*domain-name*>
Domain name for which the server is an authority.

Restrictions

None.

Examples

To configure a single DNS server and configure the X-Pedition’s DNS domain name to “mrb.com”:

```
xp(config)# system set dns server 10.1.2.3 domain mrb.com
```

To configure three DNS servers and configure the X-Pedition’s DNS domain name to “mrb.com”:

```
xp(config)# system set dns server “10.1.2.3 10.2.10.12 10.3.4.5” domain mrb.com
```

system set dst-changing

Purpose

Sets Daylight Saving Time according to specific days.

Format

system set dst-changing [s-wk <*num*>**] [s-dow** <*num*>**] [s-mo** <*num*>**] [s-hr** <*num*>**]**
[s-min <*num*>**] [e-wk** <*num*>**] [e-dow** <*num*>**] [e-mo** <*num*>**] [e-hr** <*num*>**] [e-min** <*num*>**]**

Mode

Configure

Description

If Daylight Saving Time is in effect in your local time zone, use one of the **system set dst-** commands to enable it on the X-Pedition (see *system set dst-fixed* on page 1247 and *system set dst-manual* on page 1249). When you enable automatic DST settings, the settings do not affect the system until the time change arrives. When Daylight Saving Time starts (s-mo, s-hr, etc.), the system time will automatically advance one hour. At the end of Daylight Saving Time (e-mo, e-hr, etc.), the system clock will subtract one hour is. To disable Daylight Saving Time settings on the X-Pedition, negate this command. (The UCT offset stays the same during all of this.)

Parameters

s-wk <*num*> This optional parameter specifies the starting week of the month. Specify a number between 1 and 5. The following is a description of the values: 1-first week, 2-second week, 3-third week, 4-fourth week, 5-last week. The default value is 1.

s-dow <*num*> This optional parameter specifies the starting day of the week. Specify a number between 1 and 7. The following is a description of the values: 1-Sunday, 2-Monday, 3-Tuesday, 4-Wednesday, 5-Thursday, 6-Friday, 7-Saturday. The default value is 1.

s-mo <*num*> This optional parameter specifies the starting month of the year. Specify a number between 1 and 12. The following is a description of the values: 1-January, 2-February, 3-March, 4-April, 5-May, 6-June, 7-July, 8-August, 9-September, 10-October, 11-November, 12-December. The default value is 1.

s-hr <*num*> This optional parameter specifies the starting hour of the day. Specify a number between 0 and 23. This is based upon a 24-hour day, where 0-beginning of the first hour and 23-beginning of the last hour for that day. The default value is 0.

s-min <*num*> This optional parameter specifies the starting minute of the hour. Specify a number between 0 and 59. This is based upon a 60-minute hour, where 0-beginning of the first minute and 59-beginning of the last minute for that hour. The default value is 0.

e-wk <*num*> This optional parameter specifies the ending week of the month. Specify a number between 1 and 5. The following is a description of the values: 1-first week, 2-second week, 3-third week, 4-fourth week, 5-last week. The default value is 1.

e-dow <*num*> This optional parameter specifies the ending day of the week. Specify a number between 1 and 7. The following is a description of the values: 1-Sunday, 2-Monday, 3-Tuesday, 4-Wednesday, 5-Thursday, 6-Friday, 7-Saturday. The default value is 1.

e-mo <*num*> This optional parameter specifies the ending month of the year. Specify a number between 1 and 12. The following is a description of the values: 1-January, 2-February, 3-March, 4-April, 5-May, 6-June, 7-July, 8-August, 9-September, 10-October, 11-November, 12-December. The default value is 1.

e-hr <*num*> This optional parameter specifies the ending hour of the day. Specify a number between 0 and 23. This is based upon a 24-hour day, where 0-beginning of the first hour and 23-beginning of the last hour for that day. The default value is 0.

e-min <*num*> This optional parameter specifies the ending minute of the hour. Specify a number between 0 and 59. This is based upon a 60-minute hour, where 0-beginning of the first minute and 59-beginning of the last minute for that hour. The default value is 0.

Restrictions

None.

Examples

To set Daylight Saving Time to start at midnight on the last Sunday of March and end at 2:00 A.M. on the first Saturday of October every year:

```
xp(config)# system set dst-changing s-wk 5 s-dow 1 s-mo 3 e-wk 1 e-dow 7 e-mo 10 e-hr 2
```

system set dst-fixed

Purpose

Sets Daylight Saving Time automatically according to specific dates.

Format

system set dst-fixed [s-mo <*num*>**] [s-day** <*num*>**] [s-hr** <*num*>**] [s-min** <*num*>**] [e-mo** <*num*>**] [e-day** <*num*>**] [e-hr** <*num*>**] [e-min** <*num*>**]**

Mode

Configure

Description

If Daylight Saving Time is in effect in your local time zone, use one of the **system set dst-** commands to enable it on the X-Pedition (see *system set dst-changing* on page 1245 and *system set dst-manual* on page 1249). When you enable automatic DST settings, the settings do not affect the system until the time change arrives. When Daylight Saving Time starts (s-mo, s-hr, etc.), the system time will automatically advance one hour. At the end of Daylight Saving Time (e-mo, e-hr, etc.), the system clock will subtract one hour is. To disable Daylight Saving Time settings on the X-Pedition, negate this command. (The UCT offset stays the same during all of this.)

Parameters

s-mo <*num*> This optional parameter specifies the starting month of the year. Specify a number between 1 and 12. The following is a description of the values: 1-January, 2-February, 3-March, 4-April, 5-May, 6-June, 7-July, 8-August, 9-September, 10-October, 11-November, 12-December. The default value is 1.

s-day <*num*> This optional parameter specifies the starting day of the month. Specify a number between 1 and 31. This is based upon a 31-day month, where 1-first day and 31-thirty first day for that month. The default value is 1.

s-hr <*num*> This optional parameter specifies the starting hour of the day. Specify a number between 0 and 23. This is based upon a 24-hour day, where 0-beginning of the first hour and 23-beginning of the last hour for that day. The default value is 0.

s-min <*num*> This optional parameter specifies the starting minute of the hour. Specify a number between 0 and 59. This is based upon a 60-minute hour, where 0-beginning of the first minute and 59-beginning of the last minute for that hour. The default value is 0.

e-mo <*num*> This optional parameter specifies the ending month of the year. Specify a number between 1 and 12. The following is a description of the values: 1-January, 2-February, 3-March, 4-April, 5-May, 6-June, 7-July, 8-August, 9-September, 10-October, 11-November, 12-December. The default value is 1.

e-day <*num*> This optional parameter specifies the ending day of the month. Specify a number between 1 and 31. This is based upon a 31-day month, where 1-first day and 31-thirty first day for that month. The default value is 1.

e-hr <*num*> This optional parameter specifies the ending hour of the day. Specify a number between 0 and 23. This is based upon a 24-hour day, where 0-beginning of the first hour and 23-beginning of the last hour for that day. The default value is 0.

e-min <*num*> This optional parameter specifies the ending minute of the hour. Specify a number between 0 and 59. This is based upon a 60-minute hour, where 0-beginning of the first minute and 59-beginning of the last minute for that hour. The default value is 0.

Restrictions

None.

Examples

To set Daylight Saving Time to start at 3:00 a.m. on April 1st and end at midnight on the 15th of September every year:

```
xp(config)# system set dst-fixed s-mo 4 s-day 1 s-hr 3 e-mo 9 e-day 15
```

system set dst-manual

Purpose

Allows you to set the system time forward by one hour after you save the command into active configuration. Negating this command will set the system time back one hour.

Format

system set dst-manual

Mode

Configure

Description

If Daylight Saving Time is in effect in your local time zone, use one of the **system set dst-** commands to enable it on the X-Pedition (see *system set dst-changing* on page 1245, and *system set dst-fixed* on page 1247). When you enable automatic DST settings, the settings do not affect the system until the time change arrives. When Daylight Saving Time starts (s-mo, s-hr, etc.), the system time will automatically advance one hour. At the end of Daylight Saving Time (e-mo, e-hr, etc.), the system clock will subtract one hour is. To disable Daylight Saving Time settings on the X-Pedition, negate this command. (The UCT offset stays the same during all of this.)

Parameters

None.

Restrictions

None.

system set extended-debug

Purpose

Sets various runtime debug extensions.

Format

system set extended-debug [**inhibit-master-reboot**] | [**enable-intr-monitor** <*num*>] | [**enable-pkt-capture**]

Mode

Configure (diagnostic mode)

Description

The **system set extended-debug** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

inhibit-master-reboot
This optional parameter prevents the former Master CPU from rebooting during a redundant CPU failover.

enable-intr-monitor <*num*>
This optional parameter allows you to set the amount of time for which to disable interrupts. If system interrupts exceed this value, the router will display an error. By default, this duration is 1000 milliseconds. This duration also includes the amount of time that splnet is on.

enable-pkt-capture
Enabling this optional parameter will capture and save the last 10 packets.

Restrictions

None.

system set high-priority-pad

Purpose

Sets the nia receive queue threshold padding for high priority packets.

Format

system set high-priority-pad <*number*>

Mode

Configure (diagnostic mode)

Description

The **system set high-priority-pad** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

<*number*>	The receive queue threshold padding.

Restrictions

None.

system set idle-timeout

Purpose

Set the console idle timeout value.

Format

system set idle-timeout serial| telnet| ssh <*timeout*>

Mode

Configuration.

Description

Use the **system set idle-timeout** command to define the amount of time (in minutes) to remain idle before the control module terminates the communication session.

Parameters

serial
Use this parameter to set the timeout value for a *serial* console connection.

telnet
Use this parameter to set the timeout value for a *telnet* console connection.

ssh
Use this parameter to set the timeout value for a *secure shell* console connection.

<*timeout*>
The amount of time to remain idle (0-60 minutes) before disconnecting a communication session. By default, this value is 5 minutes. To disable the timeout, enter a value of 0.

Restrictions

None.

Example

To set a secure shell timeout of 30 minutes, enter the following from configuration mode:

```
xp(config)# system set idle-timeout ssh 30
```

system set ifqlen-to-xmit-pkts

Purpose

Sets the length of the interface transmit packet queue.

Format

system set ifqlen-to-xmit-pkts <*number*>

Mode

Configure (diagnostic mode)

Description

The **system set ifqlen-to-xmit-pkts** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

<*number*>	The interface transmit packet queue length.

Restrictions

None.

system set ip-wakeup-intvl

Purpose

Sets wake-up interval for the IP task.

Format

system set ip-wakeup-intvl <*seconds*>

Mode

Configure (diagnostic mode)

Description

The **system set ip-wakeup-intvl** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

<*seconds*>	The duration of the wake-up interval.

Restrictions

None.

system set ipx-wakeup-intvl

Purpose

Sets wake-up interval for the IPX task.

Format

system set ipx-wakeup-intvl <*seconds*>

Mode

Configure (diagnostic mode)

Description

The **system set ipx-wakeup-intvl** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

<*seconds*>	The duration of the wake-up interval.

Restrictions

None.

system set lgrp-pkts-threshold

Purpose

Set maximum number of packets processed for a port that is part of a link group before entering receive-control-only mode.

Format

system set lgrp-pkts-threshold <*number*>

Mode

Configure (diagnostic mode)

Description

The **system set lgrp-pkts-threshold** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

<*number*>	The maximum number of packets processed.

Restrictions

None.

system set location

Purpose

Set the system location.

Format

system set location <*location*>

Mode

Configure

Description

The **system set location** command adds a string describing the location of the X-Pedition. The system name and location can be accessed by SNMP managers.

Parameters

<*location*> A string describing the location of the X-Pedition. If the string contains blanks or commas, you must use quotation marks around the string.
(Example: **“Bldg C, network control room”**.)

Restrictions

None.

system set login-banner

Purpose

Set the system login banner.

Format

system set login-banner <*string*>|**none**|**file-name name** <*string*>

Mode

Configure

Description

The **system set login-banner** command configures the initial login banner that one sees when logging into the X-Pedition. The banner may span multiple lines by adding line-feed characters in the string, "**\n**".

Parameters

<*string*> Is the text of the login banner for the X-Pedition. Banners that include more than one word must be inclosed in quotation marks (i.e., "This is a multi-word banner"). When you include the new line ("**\n**") command in the banner, the banner may span multiple lines. You may also use the tab ("**\t**") command to include tabs in the banner. It is not common to use both the new line and tab commands in a banner.

none Specifies that no login-banner be used on the X-Pedition.

file-name name <*string*>
Specifies the name of the file containing the login banner.

Restrictions

None.

Example

The following example configures a multi-line login banner:

```
xp(config)# system set login-banner "Core Router #1\nUnauthorized Access Prohibited"
```

The next person to log into the X-Pedition would see the following:

```
Core Router #1
Unauthorized Access Prohibited

Press RETURN to activate console...
```

To use a login banner from a file, enter the following:

```
xp(config)# system set login-banner file-name name the_banner_file
```

If you do not want any login-banner at all, enter the following:

```
xp(config)# system set login-banner none
```

system set low-priority-pad

Purpose

Sets the nia receive queue threshold padding for low priority packets.

Format

system set low-priority-pad <*number*>

Mode

Configure (diagnostic mode)

Description

The **system set low-priority-pad** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

<*number*>	The receive queue threshold padding.

Restrictions

None.

system set malloc

Purpose

Sets the caller trace for the system malloc functionality.

Format

system set malloc debug

Mode

Configure (diagnostic mode)

Description

The **system set malloc** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

None.

Restrictions

None.

system set max-packets-per-interrupt

Purpose

Set maximum number of packets per interrupt.

Format

system set max-packets-per-interrupt <*number*>

Mode

Configure (diagnostic mode)

Description

The **system set max-package-per-interrupt** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

<*number*>	The maximum number of packets.

Restrictions

None.

system set max-pkts-in-recv-ctrl-only

Purpose

Set maximum number of packets for control in receive-control-only mode.

Format

system set max-pkts-in-recv-ctrl-only <*number*>

Mode

Configure (diagnostic mode)

Description

The **system set max-pkts-in-recv-ctrl-only** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

<*number*>	The maximum number of packets.

Restrictions

None.

system set med-priority-pad

Purpose

Sets the nia receive queue threshold padding for medium priority packets.

Format

system set med-priority-pad <*number*>

Mode

Configure (diagnostic mode)

Description

The **system set med-priority-pad** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

<*number*>	The receive queue threshold padding.

Restrictions

None.

system set name

Purpose

Set the system name.

Format

system set name <*system-name*>

Mode

Configure

Description

The **system set name** command configures the name of the X-Pedition. The X-Pedition name will use the name as part of the command prompt.

Parameters

<*system-name*> The hostname of the X-Pedition. If the string contains blanks or commas, you must use quotation marks around the string (e.g., **"Mega-Corp** X-Pedition **#27"**.)

Restrictions

None.

system set ni-driver-debug

Purpose

Sets the nia driver debugging mode.

Format

system set ni-driver-debug

Mode

Configure (diagnostic mode)

Description

The **system set ni-driver-debug** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

None.

Restrictions

None.

system set password

Purpose

Set passwords for various CLI access modes.

Format

system set password <*mode*> <*string*>|**none**

Mode

Configure

Description

The **system set password** command sets or changes the passwords for the Login, Enable, and Configure access modes.

Note: If a password is configured for the Enable mode, the X-Pedition prompts for the password when you enter the **enable** command. Otherwise, the X-Pedition displays a message advising you to configure an Enable password, then enters the Enable mode. From the Enable mode, you can access the Configure mode to make configuration changes. Configuration mode access may require a password.

Parameters

<*mode*>
The access mode for which you are setting a password. Specify one of the following:

login The password required to start a CLI session. The X-Pedition prompts for this password when the system finishes booting.

enable The password for entering the Enable mode.

configure
The password for entering Configure mode.

<*string*>|**none**
The password. If you specify **none**, no password is required.

Note: You cannot use the string “none” as a password.

Restrictions

The X-Pedition stores passwords in the Startup configuration file. If you copy a configuration file from one X-Pedition to another, the passwords in the file also are copied and will be required on the new X-Pedition.

When you activate a new password by copying the password set command to the active configuration, the X-Pedition replaces the command with a **system set hashed-password** command, which hides the password text in the configuration file so that the password is not visible to others if they examine the configuration file.

To remove a password, enter the following command while in Configure mode:

```
xp(config)# system set password <mode> none
```

system set password-policy

Purpose

This command allows you to configure the optional password-related selections. These include the minimum password length, the number of login attempts allowed, and selecting an aging time. For added security, passwords entered into the system appear as asterisks and a password history prevents the use of the 5 previous passwords.

Format

system set password-policy auth-grace-timeout <*grace*>**| change-after-first-login {on|off}| expire-warning** <*warn*>**| history-size** <*size*>**| lifetime** <*time*>**| login-failure-grace-time** <*fail-time*>**| maximum-failed-logins** <*fails*>**| minimum-length** <*minimum*> **verification {on | off}**

Mode

Configure.

Parameters

auth-grace-timeout <*grace*>

Specifies the amount of time, in seconds, a user has to attempt to successfully log in. After this period expires, the user is disconnected (by default, 60 seconds). This value overrides any value configured by **ssh-server set auth-grace-timeout**. For details on this command, see *ssh-server set auth-grace-timeout* on page 1137.

change-after-first-login [on | off]

Select **on** to require users to change their password after their first login. By default, this option is **off**.

expire-warning <*warn*>

The number of days prior to password expiration to warn users that their passwords will expire. By default, 14 days.

history-size <*size*>

The number of passwords to keep in user password histories. By default, 5.

Note: To prevent a user from cycling through passwords to reuse an old one, deny the user access to configuration mode—this will prevent the user from resetting the password until the current one expires.

lifetime <*time*>

The duration (in days) that the password will remain valid. By default, 90 days.

login-failure-grace-time <*fail-time*>

The amount of time to wait once a user reaches the maximum number of failed login

attempts before the counter resets and the user is allowed to try again. By default, the router will wait 60 minutes.

maximum-failed-logins <*fails*>
The number of login failures allowed before disabling a user's account—by default, this value is 6. Entering 0 allows users an unlimited number of attempts.

minimum-length <*minimum*>
The minimum allowable length for user passwords (the default value is 8). Enter 0 if you do not require a minimum length.

verification [on| off]
Enables prompt to confirm password entry when adding new users or when changing passwords (the default is **on**).

Restrictions

None.

system set poweron-selftest

Purpose

Specify the type of Power-On-Self-Test (POST) to perform during system bootup.

Format

system set poweron-selftest [on|quick]

Mode

Configure

Description

The **system set poweron-selftest** command configures the type of Power-On-Self-Test (POST) the X-Pedition should perform during the next system bootup. By default, no POST is performed during system bootup. To perform POST, you must use this command to specify which type of test to run, **quick** or **full**. Once POST enabled, to turn off POST, you simply negate this command (using the **negate** command).

Parameters

on The X-Pedition will perform a **full** test during the next system bootup.

quick The X-Pedition will perform a **quick** test during the next system bootup.

Restrictions

None.

system set show-config

Purpose

Specify how configuration commands should be displayed.

Format

system set show-config alphabetical

Mode

Configure

Description

The **show** and **system show active-config** commands normally display the configuration commands in the order that they are executed. The **system set show-config** command changes the way the configuration commands are shown.

Parameters

alphabetical Shows the configuration commands in alphabetical order.

Restrictions

None.

Example

To display the configuration commands in alphabetical order:

```
xp(config)# system set show-config alphabetical
```

system set spooler-memory-limit

Purpose

Increase the amount of memory allocated to the system spooler.

Format

system set spooler-memory-limit <*memory-limit*>

Mode

Configure

Description

The system spooler buffers CLI output before sending it to a display device. If the %SYS-W-SPOOLOVERFLOW message appears, use the **system set spooler-memory-limit** command to increase the amount of memory allocated to the system spooler. Use caution when increasing the spooler memory limit—increasing the limit to a very high value can adversely affect other areas of the system that may be low on available memory.

Parameters

<*memory-limit*>	The amount of memory (300-4000) in kilobytes to allow the system spooler to use. By default, this value is 800.

Restrictions

None.

Example

To set the spooler memory limit to 1,000 KB use the following:

```
xp(config)# system set spooler-memory-limit 1000
```

system set stp-pkts-threshold

Purpose

Set maximum number of spanning tree packets processed before entering receive-control-only mode.

Format

system set stp-pkts-threshold <*number*>

Mode

Configure (diagnostic mode)

Description

The **system set stp-pkts-threshold** command is a system tuning command used only under the direction of Enterasys support personnel.

Parameters

<*number*>	The maximum number of spanning tree packets.

Restrictions

None.

system set syslog

Purpose

The **system set syslog** command identifies the Syslog server to which the X-Pedition should send system messages. You can control the type of messages sent based on message severity (controlled by the option **level**) and the facility selected. On the Syslog server, you can decide what to do with these messages based on the level as well as the facility. For example, you might choose to discard the messages, write them to a file or send them out to the console. You can further identify the source of the system messages sent to the Syslog server by specifying a source IP address for the Syslog on the X-Pedition.

The X-Pedition keeps the last <*n*> messages in a local circular buffer. By default, this buffer keeps the last 50 Syslog messages. You can change the buffer size to hold anywhere from 10–200 messages. To view the current buffer size, enter the **system show syslog buffer** command.

Format

system set syslog [**server** <*hostname-or-IPaddr*>] [**local**] [**level** <*level-type*>] [**facility** <*syslog-facility-type*>] [**source** <*source-IPaddr*>] [**buffer-size** <*size*>]

Mode

Configure

Parameters

server <*hostname-or-IP-addr*>
Hostname or IP address of the Syslog server.

If the syslog server is not receiving messages, enter the **system show syslog** command to view the number of sent and unsent messages. If there are multiple sent messages (e.g., 50) and no unsent messages (i.e., 0), a misconfiguration may exist on the server side. You may also use the **system show syslog buffer** command to display the last 20 sent messages, or **system show syslog buffer number** <*num*> to display the last *n* sent messages followed by any unsent messages.

Note: When using the **system show syslog buffer** command, if the total number of sent messages is fewer than 20, the X-Pedition will also display any unsent messages—as long as the total number of messages displayed does not exceed 20 (or the number specified).

[local]
This parameter logs Syslog messages to a local log file, **int-flash/cfg/syslog**—even if you have not configured a remote Syslog server.

Note: The local flash is *NOT the flash card*. It is the X-Pedition's internal buffer.

Each time the router reboots and the Syslog facility initializes, the local Syslog file moves to **int-flash/cfg/syslog.bak** and a new log is created. Local logging is subject to the Syslog filtering mechanism. To display the contents of the local log files use either of the following:

```
xp# file type syslog
```

```
xp# file type syslog.bak
```

Note: You may still use the **system show syslog buffer** command to display the buffered messages.

level <*level-type*>
Level of messages you want the X-Pedition to log. Specify one of the following:

fatal Logs only fatal messages.

error Logs fatal messages and error messages.

warning Logs fatal messages, error messages, and warning messages. This is the default.

audit Logs fatal messages, error messages, warning messages, and audit messages.

info Logs all messages, including informational messages.

facility <*syslog-facility-type*>
Type of facility under which you want messages to be sent. By default, unless specified otherwise, messages are sent under facility *local7*. The facility-type can be one of the following:

kern kernel messages

user user messages

daemon daemon messages

local0 Reserved for local use

local1 Reserved for local use

local2 Reserved for local use

local3 Reserved for local use

local4 Reserved for local use

local5 Reserved for local use

local6 Reserved for local use

local7 Reserved for local use

source <*source-IPaddr*>
Source IP address of the messages sent to the Syslog server. You must specify a Unicast IP address in the form a.b.c.d.

buffer-size <*size*>
The Syslog message buffer size. The size specifies how many messages the Syslog buffer can hold. You can specify a number from 10 – 200, giving the buffer a capacity to hold from 10–200 Syslog messages. The default is 50.

Restrictions

None.

Example

To enable the syslog client on the X-Pedition, enter the **system set syslog server** command into the configuration. After you make the configuration active, the router will begin logging syslog messages (if you enter this command in the startup configuration, the X-Pedition will also log boot messages). To send messages to a directly connected server, enter the following:

```
xp(config)# vlan create vlan_server port-based
xp(config)# vlan add port et.1.1 to vlan_server
xp(config)# interface create ip servernet address-netmask 10.1.1.1/24 vlan vlan_server
xp(config)# system set syslog server 10.1.1.50
```

For a server located at 192.168.1.230:

```
xp(config)# system set syslog server 192.168.1.230
```

For a server located at 192.168.1.230 that logs to facility local6 with a minimum syslog level of info:

```
xp(config)# system set syslog server 192.168.1.230 facility local6 level info
```

For a server located at admin.mycompany.com:

```
xp(config)# system set syslog server admin.mycompany.com
```

Note: If you specify a server name, you will need to configure a DNS entry for that server in your domain. If DNS is not running, the X-Pedition will not be able to resolve the name and the router will eventually drop syslog messages.

To set up a simple syslog server with the syslog server denoted as an IP address:

```
xp(config)# vlan create servernet port-based
xp(config)# vlan add ports et.3.3 to servernet
xp(config)# vlan add ports et.3.7 to servernet
xp(config)# vlan add ports et.3.8 to servernet
xp(config)# interface create ip servers address-netmask 10.10.10.1/24 vlan servernet
xp(config)# system set syslog server 10.10.10.76
```

To set up a simple syslog server with the syslog server denoted as an DNS name:

```
xp(config)# vlan create servernet port-based
xp(config)# vlan add ports et.3.3 to servernet
xp(config)# vlan add ports et.3.7 to servernet
xp(config)# vlan add ports et.3.8 to servernet
xp(config)# interface create ip servers address-netmask 10.10.10.1/24 vlan servernet
xp(config)# system set dns server 10.10.10.55
xp(config)# system set dns domain mycompany.com
xp(config)# system set syslog server logmach
```

The following example demonstrates how to set up a complex server were the facility specified is LOG_LOCAL6, the system logs only error messages and above, and the sent buffer-size is 10. All syslog messages from this router will be sourced with the IP address of 10.10.10.1:

```
xp(config)# vlan create servernet port-based
xp(config)# vlan add ports et.3.3 to servernet
xp(config)# vlan add ports et.3.7 to servernet
xp(config)# vlan add ports et.3.8 to servernet
xp(config)# interface create ip servers address-netmask 10.10.10.1/24 vlan servernet
xp(config)# system set dns server 10.10.10.55
xp(config)# system set syslog server logmach.mycompany.com facility local6 source 10.10.10.1 level
error buffer-size 10
```

To log only fatal and error level messages to the Syslog server on 10.1.43.77:

```
xp(config)# system set syslog server 10.1.43.77 level error
```

system set syslog-levels

Purpose

This command allows users to override the default Syslog message severity level for a given error message facility. See *system show syslog levels* on page 1297 for a list of error message facilities. The *system set syslog* on page 1275 command is used to establish the default message severity level.

Format

system set syslog-levels <*facility*> **level** <*level*>

Mode

Configure.

Parameters

<*facility*> This parameter identifies which error message facility to set to the selected level.

<*level*> The minimum Syslog message level.

fatal	Log fatal messages only.
error	Log fatal and error messages only.
warning	Log fatal, error, and warning messages only.
audit	Log fatal, error, audit, and warning messages only.
info	Log all messages.

Restrictions

None.

system set terminal

Format

system set terminal baud <*baud-rate*>**| columns** <*number*> **rows** <*number*>

Mode

Configure

Description

The **system set terminal** command globally sets parameters for a serial console's baud rate, output columns, and output rows.

Parameters

baud <*baud-rate*>
Sets the baud rate. You can specify one of the following:

- **300**
- **600**
- **1200**
- **2400**
- **4800**
- **9600**
- **19200**
- **38400**

columns <*number*>
Default number of columns (20—32767 inclusive).

rows <*number*>
Default number of rows (0—32767 inclusive). Enter 0 to disable pagination.

Restrictions

None.

Example

The command in the following example sets the baud rate for the management terminal connected to the System Control module.

```
xp(config)# system set terminal baud 38400
```

system set tftpsource

Purpose

Set the IP address to use when uploading file to a TFTP server.

Format

system set tftpsource <*Ip address*>

Mode

Configure.

Description

Use this command to specify the source address to use when uploading files to a TFTP server. The IP Address must be a directly connected or management IP address. By default, the X-Pedition will use the default source address (usually the IP address of the outgoing interface). To change the source address, please refer to the **copy active to tftpserver** command.

Parameter

<*Ip address*> The IP address to use when uploading files to the TFTP server.

Example

```
xp# system set tftpsource 10.50.88.2
```

system set timezone

Purpose

Sets time zone information or time offset.

Format

system set timezone <*timezone*>|<*minutes*>

Mode

Configure

Description

The **system set timezone** command sets the local time zone for the X-Pedition. You can use one of the time zone keywords to specify the local time zone or specify the time offset in minutes. You must configure the time zone in order to use NTP (Network Time Protocol) to synchronize the X-Pedition's real time clock.

Parameters

<*timezone*>

Sets the time zone using one of the following keywrods:

est	Eastern Standard Time (UCT -05:00)
cst	Central Standard Time (UCT -06:00)
mst	Mountain Standard Time (UCT -07:00)
pst	Pacific Standard Time (UCT -08:00)
uct-12	Eniwetok, Kawajalein (UCT -12:00)
uct-11	Midway Island, Samoa (UCT -11:00)
uct-10	Hawaii (UCT -10:00)
uct-9	Alasaka (UCT -09:00)
uct-8	Pacific Standard Time (UCT -08:00)
uct-7	Mountain Standard Time (UCT -07:00)
uct-6	Central Standard Time (UCT -06:00)
uct-5	Eastern Standard Time (UCT -05:00)

uct-4	Caracas, La Paz (UCT -04:00)
uct-3	Buenos Aires, Georgetown (UCT -03:00)
uct-2	Mid-Atlantic (UCT -02:00)
uct-1	Azores, Cape Verde Island (UCT -01:00)
uct	Greenwich, London, Dublin (UCT)
uct+1	Berlin, Madrid, Middle European Time, Paris (UCT +01:00)
uct+2	Athens, Helsinki, Istanbul, Cairo (UCT +02:00)
uct+3	Moscow, Nairobi, Riyadh (UCT +03:00)
uct+4	Abu Dhabi, Kabul(UCT +05:00)
uct+5	Pakistan (UCT +05:00)
uct+5:30	India (UCT +05:30)
uct+6	Bangladesh (UCT +06:00)
uct+7	Bangkok, Jakarta (UCT +07:00)
uct+8	Beijing, Hong Kong, Singapore(UCT +08:00)
uct+9	Japan, Korea (UCT +09:00)
uct+10	Sydney, Guam (UCT +10:00)
uct+11	Solomon Is. (UCT +11:00)
uct+12	Fiji, Marshall Is. Auckland (UCT +12:00)

<*minutes*>
Specify the time zone offset in minutes. Valid values are between -720 minutes to + 720 minutes.

Restrictions

None.

Example

To set the local time zone to Pacific Standard Time (UCT -8:00).

```
xp(config)# system set timezone pst
```

system set user

Purpose

The audit trail monitors what administrative changes are performed on the system and who performs them. The **system set user** command allows network administrators to create an account for each user and specify a user ID, password, and access privileges.

See also *system set password-policy* on page 1269.

Format

system set user <*username*> **[new-password] [password-option {never-expires| normal}] [privilege-level {login | enable| config}] [status {always-enabled | disabled| enabled}]**

Mode

Configure.

Parameters

<*username*> The name selected for this user.

new-password
The router will ask you for the password only after you execute this command—do not enter it as part of this parameter. You must always enter a password when creating an account for a new user or when changing the access privileges for an existing user.

password-option [never-expires | normal]
When you select the **normal** option (the default), the router assigns a lifetime limit to the current user's password. To exempt this user from the password lifetime limit, specify **never-expires**

privilege-level [login | enable]
The mode or privilege where the current user can gain access to the system. You must specify login (non-enable user mode) or enable mode.

status [always-enabled | disabled | enabled]
To prevent this user account from being disabled after too many failed login attempts, specify **always-enabled**. To disable this user account, specify **disable**. To disable the account when the router detects too many failed login attempts, select **enabled** (the default).

Note: Once users are configured, the login prompt changes to request the username and password.

Restrictions

None.

system show

Purpose

Show system information.

Format

system show <*system-parm*>

Mode

Enable

Description

The **system show command** shows the active settings for the following system parameters:

- Active configuration (CLI configuration of the running system)
- Size of the Syslog message buffer
- Contact information for the X-Pedition administrator (if you set one using the **system set contact** command)
- Current system time and date (if you set them using **system set date** command)
- Time that has elapsed since the X-Pedition was rebooted and the system time and date when the last reboot occurred
- IP address(es) and domain name of DNS servers the X-Pedition can use (if you set them using **system set dns** command)
- Hardware information
- Location of the X-Pedition (if you set one using the **system set location** command)
- System name of the X-Pedition (if you set one using the **system set name** command)
- IP address or hostname of Syslog server and the message level (if you set these parameters using the **system set syslog** command)
- Configuration changes in the scratchpad that are waiting for activation
- Software version running on the Control Module
- Last five Telnet connections to the X-Pedition
- Current Telnet sessions on the X-Pedition
- CPU and other resource usage

Parameters

<system-parm>
System parameter you want to display. Specify one of the following:

6000-backplane-status (Advanced Router Module only)
Shows backplane status of the Advanced Router Module for the SmartSwitch 6000 (6SSRM-02 5SSRM-02).

active-config
Shows the active configuration of the system.

bootlog
Shows the contents of the boot log file, which contains all the system messages generated during bootup.

bootprom
Shows boot PROM parameters for TFTP downloading of the system image. This information is useful only if you have configured the system to download the system image via TFTP.

buffer-size
Determines the size (10-200) of the *sent* buffer. The default value is **50**.

capacity all| chassis| task| cpu| memory
Shows usage information about various resources on the **X-Pedition**. See *system show capacity* on page 1293.

contact
Shows the contact information (administrator name, phone number, and so on).

cpu-utilization
Shows the percentage of the CPU that is currently being used.

date
Shows the system time and date.

dns
Shows the IP addresses and domain names for the DNS servers the X-Pedition can use.

environmental-info
Shows environmental information, such as temperature and power supply status.

hardware
Shows detailed hardware information about installed CPUs, switching fabrics, and line cards.

idle-timeout serial|telnet
Shows the timeout value (in minutes). If the communication interface remains idle past **idle-timeout** value, the communication session will be closed by the system. You can specify a timeout value for a serial connection or a telnet connection. A value of 0 means that the **idle-timeout** feature is disabled.

location
Shows the X-Pedition's location.

login-banner
Shows the X-Pedition's login banner. The login banner can be configured using the **system set login-banner** command.

name
Shows the X-Pedition's name.

poweron-selftest-mode
Shows the type of Power-On Self Test (POST) that should be performed, if any.

scratchpad
Shows the configuration changes in the scratchpad. These changes have not yet been activated.

security-log
Displays information on up to five previous users who logged in to the X-Pedition using TACACS+ or RADIUS.

ssh-access
Shows a summary of the last five SSH clients to access the router.

startup-config
Shows the contents of the Startup configuration file.

switching-fabric
Shows the status of switching fabric cards.

syslog
The X-Pedition can store up to 2000 boot messages to send to the Syslog server—5000 during boot—in the *unsent* message queue. The X-Pedition can also store the last 50 *sent* messages in memory. The Syslog parameter allows you to display the IP address of the Syslog server and the level of messages the X-Pedition sends to the server (e.g., Minimum Syslog level: INFO, Buffer Size: 50 sent messages 15 unsent messages).

syslog buffer
Displays up to 20 of the most recently sent messages. If the total number of sent messages is fewer than 20, the X-Pedition will also display any unsent messages—as long as the total number of messages does not exceed 20. The X-Pedition uses "(*)" to denote an unsent Syslog message. See *Examples* on page 1291.

number <*num*>	The total number of messages to display.
levels	Displays the minimum Syslog levels configured for each facility. The following example depicts a sample configuration (please refer to *Facility Support* on page 51 and *Logging Methods* on page 55 for further information).

```
xp# system show syslog levels

Syslog levels for error message facilities

------- INFO ------->
RMON SSH

------- AUDIT ------->
SNMP TELNETD VLAN

------- WARNING ------->
BGP OSPF PIM RIP

------- ERROR ------->
ACL             ACL_LOG     AGGRGEN     ATM_DIAG    ATM         ARP
AUTH            CDP         CLI         CONFIG      CONS        CTRONCHASSIS
DDT             DVMRP        ERR        ETH         FDDI        GARP
GATED           GVRP        HBT         INTERFACE   IGMP        IGMP_PIM
IP              IPC         IPHELPER    IPRED       STRNK       LOADBAL
L2TM            L3AGE       MIRRORING   MULTICAST   MT          NETSTAT
NI              NTP         PHY_POLL    PING        POLICY      PPP
PROFILE         QOS         RCP         RDISC       RES         RL
SIO             SONET       SR          STATIC      STATS       STP
SYSLOG          SYS         TFTP        TR          T1T3CLI     UNICAST
WAN             WC          ARE         ATALK       COMMON      NAT
NETFLOW         PBR         DHCPD       IPX         RARPD       RELAY

------- FATAL ------->
PTY
```

telnet-access
Lists the last five Telnet connections to the X-Pedition.

terminal
Shows the default terminal settings (number of rows, number of columns, and baud rate).

timezone
Shows the time zone offset from UCT in minutes.

uptime
Shows how much time has elapsed since the most recent reboot.

users
Shows the current Telnet connections to the X-Pedition.

version
Shows the software version running on the X-Pedition.

Restrictions

None.

Examples

Use the **show** command showing configured syslog parameters:

```
xp# system show syslog

Syslog host: 10.10.10.76, Facility: LOG_LOCAL6
Minimum syslog level: INFO, Buffer Size: 50 sent messages 89 unsent messages
Source IP address: 10.10.10.1
```

Display the actual buffer showing the last 10 messages sent:

```
xp# system show syslog buffer number 10

2001-10-26 11:31:37 %SYS-W-NOPASSWD, no password for enable, use 'system set password' in Config mode
2001-10-26 11:31:35 %SYS-W-NOPASSWD, no password for login, use 'system set password' in Config mode
2001-10-26 11:31:35 %SYS-I-NETSTART, network interfaces are now enabled
2001-10-26 11:31:35 %SNMP-I-ENABLED, SNMP Agent enabled
2001-10-26 11:31:35 %GATED-I-RECONFIGDONE, Routing configuration changes completed (pid 0x80e763a8).
2001-10-26 11:31:35 %OSPF-I-ROUTERIDFND, OSPF Router Id found: 172.1.1.1
2001-10-26 11:31:34 %STP-I-PORT_STATUS, Port status change detected: et.5.8 - Port Down
2001-10-26 11:31:34 %STP-I-PORT_STATUS, Port status change detected: et.5.7 - Port Down
2001-10-26 11:31:34 %STP-I-PORT_STATUS, Port status change detected: et.5.6 - Port Down
2001-10-26 11:31:34 %STP-I-PORT_STATUS, Port status change detected: et.5.5 - Port Down
(*) denotes an unsent syslog message
```

To display last 10 messages sent to the Syslog server and any unsent messages, enter the following:

```
xp-181-11# system show syslog buffer number 10

2001-10-26 11:31:35 %SYS-I-NETSTART, network interfaces are now enabled
2001-10-26 11:31:35 %SNMP-I-ENABLED, SNMP Agent enabled
2001-10-26 11:31:35 %GATED-I-RECONFIGDONE, Routing configuration changes completed (pid 0x80e763a8).
2001-10-26 11:31:35 %OSPF-I-ROUTERIDFND, OSPF Router Id found: 172.1.1.1
2001-10-26 11:31:34 %STP-I-PORT_STATUS, Port status change detected: et.5.8 - Port Down
2001-10-26 11:31:34 %STP-I-PORT_STATUS, Port status change detected: et.5.7 - Port Down
2001-10-26 11:31:34 %STP-I-PORT_STATUS, Port status change detected: et.5.6 - Port Down
2001-10-26 11:31:34 %STP-I-PORT_STATUS, Port status change detected: et.5.5 - Port Down
2001-10-26 11:31:37 (*) %SYS-W-NOPASSWD, no password for enable, use 'system set password' in Config mode
2001-10-26 11:31:35 (*) %SYS-W-NOPASSWD, no password for login, use 'system set password' in Config mode

(*) denotes an unsent syslog message
```

To display information on the last users to log in to the X-Pedition using TACACS+ or RADIUS, enter the following:

```
xp-181-11# system show security-log

 User ID                : johnny
 tty                    : tty1
 Security Type          : Radius
 Last AAA server used   : 10.136.15.101
 Number of Sessions     : 1
 Start Session Time     : 2002-06-27 09:10:16
 Connection Status      : Currently Connected
 Last Session ended at  : 2002-06-27 09:09:54
 Time Last Accessed     : 2002-06-27 09:26:34
 Last Command           : system show security-log
 Current Mode           : Enable
 Config command Cntr    : 0
 Enable command Cntr    : 1
 Login command Cntr     : 3
 Total command Cntr     : 4
```

Field Descriptions

User ID	The user ID used to log in
tty	Connected through
Security Type	The security methodology used
Last AAA server used	Authenticated to what authentication server
Start Session Time	The timestamp of when the user connected to the router. If not currently connected, this indicates the previous connection time.
Connection Status	Indicates whether or not the user's connection is active.
Last Session ended at	The timestamp of the first time the user connected since boot. If this is the first time, the "First time connected since boot" message.
Time Last Accessed	The timestamp of the most recent command entered by the user.
Last Command	The most recent command entered.
Current Mode	The active CLI mode (i.e., enable, config, login).
Config command Cntr	The number of commands executed from configuration mode.
Enable command Cntr	The number of commands executed from enable mode.
Login command Cntr	The number of commands executed from login mode.
Total command Cntr	The total number of commands executed.

system show capacity

Format

system show capacity all | chassis | task | cpu | memory

Mode

Enable

Description

The **system show capacity** command displays information about the X-Pedition's resources.

Restrictions

None.

Example

To display usage information for all X-Pedition resources:

```
xp# system show capacity all

Capacity MIB Chassis Information:
Total       Used        Free        CPU          Power Supply  Switch Fabric
Slots       Slots       Slots       Redundancy   Redundancy    Redundancy
----------  ----------  ----------  ----------   ------------  ------------
8           7           1           Present      Present       No Support

Capacity MIB Task Information:

Index       Name        Count       Task Status      Memory Used
-----       ----        -----       -----------      -----------
1           CONS_T      2           Suspended (event) 8096
6           IPX_T       16996       Suspended (event) 8096
11          STATS_T     169939      Suspended (event) 8096
16          PHY_POLL    339892      Suspended (event) 16384
21          L3_ACL_T    1           Suspended (event) 16192
26          IPC         2           Suspended (event) 8096
31          PINGER_T    7623        Suspended (event) 8096
36          BOUNCE      2           Suspended (event) 8096
41          CONS2T      2           Suspended (event) 8096

Capacity MIB Storage Information:
Type        Description     Size      Free       Used      Block  Remov  Fail
-------     ---------------  ----------  ----------  ----------  -----  -----
CPU         Internal CPU    4194304   3837935   356369    16     True   0
FLASH       Internal Flash  756       745       11        64     True   0
L2HW        port et.2.1     5888      5887      1         64     True   0
L2HW        port et.2.2     5888      5887      1         64     True   0
L2HW        port et.2.3     5888      5887      1         64     True   0
L2HW        port et.2.4     5888      5887      1         64     True   0
L2HW        port et.2.5     5888      5887      1         64     True   0
L2HW        port et.2.6     5888      5887      1         64     True   0
L2HW        port et.2.7     5888      5887      1         64     True   0
L2HW        port et.2.8     5888      5887      1         64     True   0
L2HW        port et.3.1     5888      5887      1         64     True   0
L2HW        port et.3.2     5888      5887      1         64     True   0
L2HW        port et.3.3     5888      5887      1         64     True   0
L2HW        port et.3.4     5888      5887      1         64     True   0
.
Capacity MIB CPU Information:
Slot    Util    L3 Learned/Aged      L2 Learned/Aged      NIA Received/Xmt
-----   -----   ---------------------  ---------------------  ------------------------
0       1         0         /0         0         /0         0       /75684
xp#
```

Field Definitions

Field	Description
Capacity MIB Chassis Information displays information about the chassis:	
Total Slots	The total number of slots in the chassis, including the slot for the CPU.
Used Slots	The number of used slots, including the slot used by the CPU.
Free Slots	The number of available slots.
CPU Redundancy	A redundant control module is present.
Power Supply Redundancy	A redundant power supply is present.
Switch Fabric Redundancy	A redundant switch fabric is installed (XP-8600 only).
Capacity MIB Task Information displays information about the tasks scheduled for the CPU:	
Index	The unique index assigned to the task.
Name	The encrypted name assigned to the task. This is unique for each type of task.
Count	The number of times the task was scheduled to run. This represents a cumulative count from the time the router was started.
Task Status	The current status of the task. The task status can be *Ready* (task is scheduled and ready), *Suspended* (task is waiting for something, such as a queue or memory), *Finished*, or *Terminated*.
Memory Used	The amount of memory consumed by the task. This can be used to monitor the excess memory used by a particular task and is expressed in bytes.
Capacity Storage Information provides information about the non-volatile memory devices in the router:	
Type	The type of storage device.
Description	Describes the storage device.
Size	The total memory capacity of the device, expressed in blocks.

Field	Description
Free	The amount of free memory in the device, expressed in blocks./
Used	The size of the used memory on the device, expressed in blocks. This includes blocks of memory that are used only partially.
Block	The size of the memory blocks in the memory device. This is the minimum block size of memory returned when requesting memory. This value is expressed in bytes.
Remove	Indicates whether or not the memory can be removed.
Fail	The number of times a memory allocation in the memory device has failed. For Layer-2 and Layer-3 hardware, this refers to the number of times a full hash bucket condition has been met.

Capacity MIB CPU Information displays information about the various hardware tables:

Slot	The slot number of the CPU.
Utilization	The CPU utilization expressed as an integer percentage. This is calculated over the last 5 seconds at a 0.1 second interval as a simple average.
L3 Learned	The total number of new Layer-3 flows the CPU has processed and programmed into the Layer-3 hardware flow tables. Layer-3 flows are IP or IPX packets that are routed from one subnet to another.
L3 Aged	The total number of Layer-3 flows that were removed from the Layer-3 hardware flow table across all modules.
L2 Learned	The total number of Layer-2 flows or addresses learned.
L2 Aged	The total number of Layer-2 flows that were removed from the Layer-2 lookup tables.
NIA Received	The total number of packets received by the NIA chip. This is useful in gauging how many packets are forwarded to the CPU for processing.
NIA XMT	The total number of packets transmitted by the NIA chip. This is useful in determining how much the CPU communicates directly with management stations and other routes.

system show syslog levels

Purpose

This command allows users to view the Syslog message facility levels. The output for each facility indicates the minimum Syslog message level configured.

Format

system show syslog levels

Mode

Enable.

Parameters

None.

Restrictions

None.

Example

The following example uses the Syslog configuration below:

```
system set syslog server 10.136.15.101 local level error
system set syslog-levels RMON level info
system set syslog-levels SNMP level audit
system set syslog-levels VLAN level audit
system set syslog-levels SSH level info
system set syslog-levels OSPF level warning
system set syslog-levels RIP level warning
system set syslog-levels BGP level warning
system set syslog-levels PIM level warning
system set syslog-levels TELNETD level audit
system set syslog-levels PTY level fatal
```

```
xp# system show syslog levels
Syslog levels for error message facilities

------- INFO ------->
RMON SSH

------- AUDIT ------->
SNMP TELNETD VLAN

------- WARNING ------->
BGP OSPF PIM RIP

------- ERROR ------->
ACL             ACL_LOG         AGGRGEN         ATM_DIAG        ATM             ARP
AUTH            CDP             CLI             CONFIG          CONS            CTRONCHASSIS
DDT             DVMRP           ERR             ETH             FDDI            GARP
GATED           GVRP            HBT             INTERFACE       IGMP            IGMP_PIM
IP              IPC             IPHELPER        IPRED           STRNK           LOADBAL
L2TM            L3AGE           MIRRORING       MULTICAST       MT              NETSTAT
NI              NTP             PHY_POLL        PING            POLICY          PPP
PROFILE         QOS             RCP             RDISC           RES             RL
SIO             SONET           SR              STATIC          STATS           STP
SYSLOG          SYS             TFTP            TR              T1T3CLI         UNICAST
WAN             WC              ARE             ATALK           COMMON          NAT
NETFLOW         PBR             DHCPD           IPX             RARPD           RELAY

------- FATAL ------->
PTY
```

Field Descriptions

ACL	Access Control List
ACL_LOG	Access Control List Log
AGGRGEN	Aggregated/Generated Root
ARE	Advanced Routing Engine
ARP	Address Resolution Protocol
ATALK	Apple-Talk
ATM	Asynchronous Transfer Mode
ATM_DIAG	Asynchronous Transfer Mode Diagnostics
AUTH	Authentication
BGP	Border Gateway Protocol
CDP	Cabletron/Cisco Discovery Protocol
CLI	Command Line Interface

COMMON	Common Command Line Interface
CONFIG	Configuration
CONS	Console
CTRONCHASSIS	Chassis-Related
DDT	Dynamic Disassembly Tool
DHCPD	Dynamic Host Configuration Protocol
DVMRP	Distance Vector Multicast Routing Protocol
ERR	Error
ETH	10Base-T Ethernet Driver
FDDI	Fiber Distributed Data Interface
GARP	Generic Attribute Registration Protocol
GATED	Gate Daemon
GVRP	GARP VLAN Registration Protocol
HBT	Control Module Heartbeat
IGMP	Internet Group Membership Protocol
IGMP_PIM	Internet Group Membership Protocol Protocol Independent Multicast
INTERFACE	Interface
IP	IP Stack
IPC	The IPC facility used by WAN
IPHELPER	The IP Helper and BOOTP/DHCP Relay Agent
IPRED	IP Redundancy (VRRP)
IPX	Internet Packet Exchange
L2TM	Layer-2 Table Manager
L3AGE	Layer-3 Aging facility
LOADBAL	Load Balance
MIRRORING	Mirroring
MT	Multicast Traceroute
MULTICAST	Multicast
NAT	Network Address Translation
NETFLOW	NetFlow
NETSTAT	Netstat
NI	Network Interface Driver
NTP	Network Time Protocol
OSPF	Open Shortest Path First

PBR	IP Policy
PHY_POLL	Phy_Poll
PIM	Protocol Independent Multicast
PING	Ping
POLICY	Policy
PPP	Point-to-Point Protocol
PROFILE	Profile
PTY	Pseudo TTY
QOS	Quality of Service
RARPD	Reverse Address Resolution Protocol
RCP	Remote Copy Protocol
RDISC	Router Discovery
RELAY	Relay
RES	Resolver
RIP	Routing Information Protocol
RL	Rate Limit
RMON	Remote Network Monitoring
SIO	Serial Input/Output
SNMP	Simple Network Management Protocol
SONET	Packet-Over-Sonet
SR	Temperature-Related Messages
SSH	Secure Shell
STATIC	Static Address
STATS	Statistics
STP	Spanning Tree Protocol
STRNK	SmartTRUNK
SYS	System
SYSLOG	Syslog
T1T3CLI	T1/T3 Configuration Commands
TELNETD	Telnet
TFTP	Trivial File Transfer Protocol
TR	Traceroute
UNICAST	Unicast

VLAN	Virtual Local Area Network
WAN	Wide Area Network
WC	Web Cache

Chapter 68

tacacs-plus Commands

The **tacacs-plus** commands let you secure access to the X-Pedition using the TACACS Plus protocol. When users log in to the X-Pedition or try to access Enable mode, they are prompted for a password. If TACACS Plus authentication is enabled on the X-Pedition, it will contact a TACACS Plus server to verify the user. If the user is verified, he or she is granted access to the X-Pedition.

Notes:

- The X-Pedition currently supports the Password Authentication Protocol (PAP) method of authentication but not the Challenge Handshake Authentication Protocol (CHAP) method.
- The X-Pedition no longer supports TACACS and will ignore any commands used for it in the configuration—without generating an error.

Command Summary

Table 54 lists the **tacacs-plus** commands. The sections following the table describe the command syntax.

Table 54. tacacs-plus commands

tacacs-plus accounting command level <*level*>
tacacs-plus accounting shell start|stop|all
tacacs-plus accounting snmp active|startup
tacacs-plus accounting system fatal|error|warning|info
tacacs-plus authentication login|enable|system

Table 54. tacacs-plus commands (Continued)

tacacs-plus enable
tacacs-plus set server <*IPaddr*> **[port <***number***>] [timeout** <*number*>**] [retries** <*number*>**] [deadtime** <*number*>**] [key** <*string*>**] [source** <*IFname_IPaddr*>**]** **tacacs-plus set [timeout** <*number*>**] [retries** <*number*>**] [deadtime** <*number*>**] [key** <*string*>**] [source** <*IFname_IPaddr*>**] [last-resort password|succeed|deny]**
tacacs-plus show stats|all

tacacs-plus accounting command level

Purpose

Causes the specified types of commands to be logged to the TACACS Plus server.

Format

tacacs-plus accounting command level <*level*>

Mode

Configure

Description

The **tacacs-plus accounting command level** command allows you specify the types of commands that are logged to the TACACS Plus server. The user ID and timestamp are also logged.

Parameters

<level> Specifies the type(s) of commands that are logged to the TACACS Plus server. Enter one of the following values:

- 5 Log Configure commands.
- 10 Log all Configure and Enable commands.
- 15 Log all Configure, Enable, and User commands.

Restrictions

None.

Example

To cause Configure, Enable, and User mode commands to be logged on the TACACS Plus server:

```
xp(config)# tacacs-plus accounting command level 15
```

tacacs-plus accounting shell

Purpose

Causes an entry to be logged on the TACACS Plus server when a shell is stopped or started on the X-Pedition.

Format

tacacs-plus accounting shell start|stop|all

Mode

Configure

Description

The **tacacs-plus accounting shell** command allows you to track shell usage on the X-Pedition. It causes an entry to be logged on the TACACS Plus server when a shell is started or stopped. You can specify that an entry be logged when a shell is started, when a shell is stopped, or when a shell is either started or stopped.

Parameters

start Logs an entry when a shell is started.

stop Logs an entry when a shell is stopped

all Logs an entry when a shell is either started or stopped

Restrictions

None.

Example

To cause an entry to be logged on the TACACS Plus server when a shell is either started or stopped on the X-Pedition:

```
xp(config)# tacacs-plus accounting shell all
```

tacacs-plus accounting snmp

Purpose

Logs to the TACACS Plus server any changes made to the startup or active configuration via SNMP.

Format

tacacs-plus accounting snmp active|startup

Mode

Configure

Description

The **tacacs-plus accounting snmp** command allows you to track changes made to the active or startup configuration through SNMP. It causes an entry to be logged on the TACACS Plus server whenever a change is made to the ACL configuration. You can specify that an entry be logged to the active or startup configuration.

Parameters

active Logs an entry when a change is made to the active configuration.

startup Logs an entry when a change is made to the startup configuration.

Restrictions

None.

Example

To cause an entry to be logged on the TACACS Plus server whenever an ACL configuration change is made via SNMP to the active configuration:

```
xp(config)# tacacs-plus accounting snmp active
```

tacacs-plus accounting system

Purpose

Specifies the type(s) of messages to be logged on the TACACS Plus server.

Format

tacacs-plus accounting system fatal|error|warning|info

Mode

Configure

Description

The **tacacs-plus accounting system** command allows you to specify the types of messages that are logged on the TACACS Plus server.

Parameters

fatal
Logs only fatal messages.

error
Logs fatal messages and error messages.

warning
Logs fatal messages, error messages, and warning messages.

info
Logs all messages, including informational messages.

Restrictions

None.

Example

To log only fatal and error messages on the TACACS Plus server:

```
xp(config)# tacacs-plus accounting system error
```

tacacs-plus authentication

Purpose

Causes TACACS Plus authentication to be performed at either the X-Pedition login prompt or when the user tries to access Enable mode.

Format

tacacs-plus authentication login|enable|system

Mode

Configure

Description

The **tacacs-plus authentication** command allows you to specify when TACACS Plus authentication is performed: either when a user logs in to the X-Pedition, or tries to access Enable mode.

Parameters

login	Authenticates users at the X-Pedition login prompt.
enable	Authenticates users when they try to access Enable mode.
system	Authenticates $enab<n>$user when they try to access Enable or Login mode.

Restrictions

None.

Example

To perform TACACS Plus authentication at the X-Pedition login prompt:

```
xp(config)# tacacs-plus authentication login
```

tacacs-plus enable

Purpose

Enables TACACS Plus authentication on the X-Pedition. TACACS Plus authentication is disabled by default on the X-Pedition.

Format

tacacs-plus enable

Mode

Configure

Description

The **tacacs-plus enable** command causes TACACS Plus authentication to be activated on the X-Pedition. You set TACACS Plus-related parameters with the **tacacs-plus set**, **tacacs-plus accounting shell**, and **tacacs-plus authorization** commands, then use the **tacacs-plus enable** command to activate TACACS Plus authentication.

Parameters

None.

Restrictions

None.

Example

The following commands set TACACS Plus-related parameters on the X-Pedition. The commands are then activated with the **tacacs-plus enable** command:

```
xp(config)# tacacs-plus set server 207.135.89.15
xp(config)# tacacs-plus set timeout 30
xp(config)# tacacs-plus authentication login
xp(config)# tacacs-plus accounting shell all
xp(config)# tacacs-plus enable
```

tacacs-plus set

Purpose

Sets default parameters for authenticating the X-Pedition through a TACACS-Plus server.

Format

tacacs-plus set [timeout <*number*>**] [retries** <*number*>**] [deadtime** <*number*>**] [key** <*string*>**] [source** <*IFname_IPaddr*>**] [last-resort password|succeed|deny]**

Mode

Configure

Description

The **tacacs-plus set** command allows you to set TACACS-Plus-related parameters on the X-Pedition, how long to wait for the TACACS-Plus server to authenticate the user, an encryption key, and what to do if the TACACS-Plus server does not reply by a given time.

Parameters

timeout <*number*>	Is the maximum time (1-30) in seconds to wait for a TACACS-Plus server to reply. The default is 3 seconds.
retries <*number*>	The default number of times (1-10) to attempt to contact a server.
deadtime <*number*>	The length of time for transaction requests to skip over a TACACS-Plus server—up to a maximum of 1440 minutes (24 hours). This command causes the X-Pedition to mark as “dead” any TACACS-Plus server that fails to respond to authentication requests, thus avoiding the wait for the request to timeout before trying the next configured server. Additional requests for a TACACS-Plus server marked as “dead” will skip the server for the duration of minutes specified (unless all servers are marked “dead”).
key <*string*>	An encryption key shared with the TACACS-Plus server. The maximum length of this string is 128 bytes. If you do not specify an encryption key, the TACACS-Plus packet will not be encrypted (the default). If you defined multiple TACACS-Plus servers and need to encrypt only *some* of them, use the **key** parameter of the **tacacs-plus set server** command to encrypt the servers. You cannot remove the key on an individual server after creating the default key.

source <*IFname_IPaddr*> Sets the source interface name or IP address for TACACS-Plus messages.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

last-resort Is the action to take if a TACACS-Plus server does not reply within the time specified by the **timeout** parameter. Specify one of the following:

- **password** The user is prompted for the password set with **system set password** command (if one has been set).
- **succeed** Access to the X-Pedition is granted.
- **deny** Unable to connect to TACACS server, access to the X-Pedition is denied.

Restrictions

None.

Example

The following commands specify that hosts 137.72.5.9 and 137.72.5.41 are TACACS Plus servers, and the X-Pedition should wait no more than 30 seconds for a response from one of these servers. If a response from a TACACS-Plus server doesn't arrive in 30 seconds, the user is prompted for the password that was set with the X-Pedition **system set password** command.

```
xp(config)# tacacs-plus set server 137.72.5.9
xp(config)# tacacs-plus set server 137.72.5.41
xp(config)# tacacs-plus set timeout 30
xp(config)# tacacs-plus set last-resort password
```

tacacs-plus set server

Purpose

Sets parameters for authenticating the X-Pedition through a specific TACACS-Plus server.

Format

tacacs-plus set server <*IPaddr*> **[port <***number***>] [timeout** <*number*>**] [retries** <*number*>**] [deadtime** <*number*>**] [key** <*string*>**] [source** <*IFname_IPaddr*>**]**

Mode

Configure

Description

The **tacacs-plus set server** command allows you to set TACACS-Plus-related parameters on the X-Pedition, including the IP address of a specific TACACS-Plus server, how long to wait for the TACACS-Plus server to authenticate the user, an encryption key, and what to do if the TACACS-Plus server does not reply by a given time.

Parameters

server <*IPaddr*>	Is the IP address of a TACACS-Plus server. You can enter up to five TACACS Plus servers. Enter one server per **tacacs-plus set server** command.
port <*number*>	The TACACS-Plus TCP port you will use (1-65535). The default port is 49.
timeout <*number*>	Is the maximum time (1-30) in seconds to wait for a TACACS Plus server to reply. The default is 3 seconds.
retries <*number*>	The default number of times (1-10) to attempt to contact this TACACS server.
deadtime <*number*>	The length of time for transaction requests to skip over a TACACS server—up to a maximum of 1440 minutes (24 hours). This command causes the X-Pedition to mark as “dead” any TACACS server that fails to respond to authentication requests, thus avoiding the wait for the request to timeout before trying the next configured server. Additional requests for a TACACS server marked as “dead” will skip the server for the duration of minutes specified (unless all servers are marked “dead”).
key <*string*>	Is an encryption key to be shared with the TACACS-Plus server. The maximum length of this string is 128 bytes.

source <*IFname_IPaddr*> Sets the source interface name or IP address for TACACS-Plus messages.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

None.

Example

The following commands specify that hosts 137.72.5.9 and 137.72.5.41 are TACACS-Plus servers, and the X-Pedition should wait no more than 30 seconds for a response from one of these servers. If a response from a TACACS Plus server doesn't arrive in 30 seconds, the user is prompted for the password that was set with the X-Pedition **system set password** command.

```
xp(config)# tacacs-plus set server 137.72.5.9
xp(config)# tacacs-plus set server 137.72.5.41
xp(config)# tacacs-plus set timeout 30
xp(config)# tacacs-plus set last-resort password
```

tacacs-plus show

Purpose

Displays information about TACACS Plus configuration on the X-Pedition.

Format

tacacs-plus show stats|all

Mode

Enable

Description

The **tacacs-plus show** command displays statistics and configuration parameters related to TACACS Plus configuration on the X-Pedition. The statistics displayed include:

accepts Number of times each server responded and validated the user successfully.

rejects Number of times each server responded and denied the user access, either because the user wasn't known, or the wrong password was supplied.

timeouts Number of times each server did not respond.

Parameters

stats Displays the accepts, rejects, and timeouts for each TACACS Plus server.

all Displays the configuration parameters set with the **tacacs-plus set** command, in addition to the accepts, rejects, and timeouts for each TACACS Plus server.

Restrictions

None.

Example

To display configuration parameters and TACACS Plus server statistics:

```
xp# tacacs-plus show all
TACACS+ status:                 ACTIVE
TACACS+ last resort:             Succeed when server fails
Command Level Logging:           15 - Log Configure, Enable and User Commands
Default TACACS+ timeout (seconds): 3
Default TACACS+ retries:         3
Default TACACS+ deadtime (minutes):0
Default TACACS+ key:             net
Default TACACS+ source IP address: Let system decide

TACACS+ servers listed in order of priority:

Server:              10.136.16.102
Port:               49
Timeout (seconds):   <Default>
Retries:             <Default>
Deadtime (minutes):  3
Key:                net
Source IP:           <Default>
Server is dead. Will be made tested again in 2 minutes

Server:              10.136.15.100
Port:               49
Timeout (seconds):   <Default>
Retries:             <Default>
Deadtime (minutes):  <Default>
Key:
Source IP:           <Default>

Server:              10.136.15.101
Port:               49
Timeout (seconds):   <Default>
Retries:             <Default>
Deadtime (minutes):  <Default>
Key:                net
Source IP:           <Default>

TACACS+ server host statistics:

Host              Accepts   Rejects   Timeouts
10.136.16.102      0        0         3
10.136.15.100      1        0         0         * Sever being used
10.136.15.101      0        0         0
```

Chapter 69

telnet Command

Format

telnet <*hostname-or-IPaddr*> **[socket** <*socket-number*>**]**

Mode

User or Enable

Description

The **telnet** command allows you to open a Telnet session to the specified host.

Parameters

<*hostname-or-IPaddr*>
The host name or IP address of the remote computer that you want to access.

socket <*socket-number*>
The TCP port through which the Telnet session will be opened. If this parameter is not specified, the Telnet port (socket number 23) is assumed. This parameter can be used to test other ports; for example, socket number 21 is the port for FTP.

Restrictions

Secure Shell (**ssh**) is a "secure" replacement for Telnet. SSH provides the same remote access to the XP that Telnet provides, but does so securely by encrypting all session data—including passwords.

Note: When you enable the SSH server, the XP automatically disables Telnet access.

Example

To open a Telnet session on the host “xp4”:

```
xp# telnet xp4
```

Chapter 70

traceroute Command

The **traceroute** command traces the path a packet takes to reach a remote host.

Format

traceroute <*host*> **[max-ttl** <*num*>**] [probes** <*num*>**] [size** <*num*>**] [source** <*host*>**] [tos** <*num*>**] [wait-time** <*secs*>**] [verbose] [noroute]**

Mode

User

Description

The **traceroute** command traces the route taken by a packet to reach a remote IP host. The **traceroute** command examines the route taken by a packet traveling from a source to a destination. By default, the source of the packet is the X-Pedition. However, one can specify a different source and track the route between it and a destination. The route is calculated by initially sending a probe (packet) from the source to the destination with a TTL of 1. Each intermediate router that is not able to reach the final destination directly will send back an ICMP Time Exceeded message. Subsequent probes from the source will increase the TTL value by 1. As each Time Exceeded message is received, the program keeps track of the address of each intermediate gateway. The probing stops when the packet reaches the destination or the TTL exceeds the **max-ttl** value.

Parameters

<*host*>
Hostname or IP address of the destination

max-ttl <*num*>
Maximum number of gateways (“hops”) to trace

probes <*num*>
Number of probes to send

size <*num*>
Packet size of each probe

source <*host*>
Hostname or IP address of the source

tos <*num*>
Type of Service value in the probe packet

wait-time <*secs*>
Maximum time to wait for a response

verbose
Displays results in verbose mode

noroute
Ignores the routing table and sends a probe to a host on a directly attached network. If the destination is not on the local network, an error is returned.

Restrictions

None.

Example

To display the route from the X-Pedition to the host *othello* in verbose mode:

```
xp# traceroute othello verbose
```

Chapter 71

vlan Commands

The **vlan** commands allow the user to perform the following tasks:

- Create VLANs
- List VLANs
- Add ports to VLANs
- Deny the addition of new ports to VLANs
- Change the port membership of VLANs
- Make a VLAN port either a trunk port or an access port

Command Summary

Table 55 lists the **vlan** commands. The sections following the table describe the command syntax.

Table 55. vlan commands

vlan add ports <*port-list*> **to** <*vlan-name*>
vlan create <*vlan-name*> <*type*> **id** <*num*>
vlan enable l4-bridging on <*vlan-name*>
vlan forbid ports <*port-list*> **from** <*string*>
vlan make <*port-type*> <*port-list*>
vlan multi-add ports <*port-list*> **to** <*basename*> **id** <*num*> **through** <*num*>
vlan multi-create <*basename*> <*type*> **id** <*num*> **through** <*num*>
vlan show

vlan add ports

Purpose

Adds ports to a VLAN.

Format

vlan add ports <*port-list*> **to** <*vlan-name*>

Mode

Configure

Description

The **vlan add ports** command adds ports and trunk ports to an existing VLAN. You do not need to specify the VLAN type when you add ports (you specify the VLAN type when you create the VLAN). For information about creating VLANs, see *vlan create* on page 1323.

Parameters

<*port-list*>
The ports you are adding to the VLAN. You can specify a single port or a comma-separated list of ports. Example: et.1.3,et.(1-3).(4,6-8).

<*vlan-name*>
Name of the VLAN to which you are adding ports.

Note: The X-Pedition will display VLAN names up to 32 characters in length.

Restrictions

The VLAN to which you add ports must already exist. To create a VLAN, use the **vlan create** command. An access port can be added to only one IP VLAN, one IPX VLAN, and one bridged-protocols VLAN.

vlan create

Purpose

Creates a VLAN based on ports or protocol.

Format

vlan create <*vlan-name*> <*type*> **id <*num*>**

Mode

Configure

Description

The **vlan create** command creates a VLAN definition. You can create a port-based VLAN or a protocol-based VLAN. For information about adding ports and trunk ports to a VLAN, see *vlan add ports* on page 1322.

Parameters

<*vlan-name*> Name of the VLAN. The VLAN name is a string up to 32 characters long.

Note: The VLAN name cannot begin with an underscore (_) or the word “SYS_”. The names “control,” “default,” “blackhole,” “reserved,” and “learning” cannot be used. The X-Pedition will display VLAN names up to 32 characters in length.

<*type*> The type of VLAN you are adding. The VLAN type determines the types of traffic the X-Pedition will forward on the VLAN. Specify any combination of the first seven types that follow ***or*** specify **port-based**:

ip
Create this VLAN for IP traffic

ipx
Create this VLAN for IPX traffic

appletalk
Create this VLAN for AppleTalk traffic

dec
Create this VLAN for DECnet traffic

sna
Create this VLAN for SNA traffic

ipv6
Create this VLAN for IPv6 traffic

bridged-protocols
Create this VLAN for extended VLAN types (DEC, SNA, Appletalk, IPv6), and non-IP and non-IPX protocols

Note: You can specify a combination of **ip**, **ipx**, **appletalk**, **dec**, **sna**, **ipv6**, and **bridged-protocols**. If you specify *any* of the extended VLAN types (**sna**, **dec**, **appletalk**, **ipv6**) with the **bridged-protocols** option, then all the other extended VLAN types are removed from the VLAN. See the following table:

Configuration Command	Protocols Included in VLAN	Protocols Excluded from VLAN
vlan create <*vlan-name*> ip	**IP**	**IPX, SNA, IPv6, DECnet, Appletalk, Other**
vlan create <*vlan-name*> ip bridged-protocols	**IP, SNA, DECnet, IPv6, Appletalk, Other**	**IPX**
vlan create <*vlan-name*> ip bridged-protocols sna	**IP, SNA, Other**	**IPX, IPv6, DECnet, Appletalk**
vlan create <*vlan-name*> ip bridged-protocols sna ipv6	**IP, SNA, IPv6, Other**	**IPX, DECnet, Appletalk**

port-based
Create this VLAN for all traffic types listed above (port-based VLAN)

Note: You can specify a combination of **ip**, **ipx**, **appletalk**, **dec**, **sna**, **ipv6**, and **bridged-protocols** *or* you can specify **port-based**; you cannot specify **port-based** with any of the other options.

id <*num*> ID of this VLAN. The ID must be unique. You can specify a number from 2 – 4094. If more than one X-Pedition will be configured with the same VLAN, you must specify the same VLAN ID on each X-Pedition.

Restrictions

The following *cannot* be used for VLAN names:

- control
- default
- blackhole
- reserved
- learning

- names starting with an underscore (_) or “sys_”

Note: The X-Pedition will display VLAN names up to 32 characters in length.

Note: Specify both SNA and bridged-protocol to successfully create an SNA based VLAN. The SNA-protocol-based VLAN (implemented in version 3.0 and later) needs to be configured with the following command:
vlan create sna bridged-protocols id <id#>
in order to forward all SNA protocol types. Refer to the following Technical Bulletin for more detail: TB0973-1

Examples

The following command creates a VLAN ‘blue’ for IP, SNA, non-IPX, non-DECnet, non-Appletalk, non-IPv6 protocols:

```
xp(config)# vlan create blue ip bridged-protocols sna
```

The following command creates a VLAN ‘red’ for IP, non-IPX, and extended VLAN types SNA, DECnet, Appletalk, and IPv6:

```
xp(config)# vlan create red ip bridged-protocols
```

vlan enable

Purpose

Enable VLAN specific features.

Format

vlan enable l4-bridging on <*vlan-name*>

Mode

Configure

Description

The **vlan enable** command allows you to enable VLAN features.

Parameters

<*vlan-name*> The name of the VLAN.

Note: The X-Pedition will display VLAN names up to 32 characters in length.

l4-bridging on This optional parameter enables Layer-4 bridging.

Restrictions

None.

vlan forbid ports

Purpose

Forbids ports from being added to an existing VLAN.

Format

vlan forbid ports <*port-list*> **from** <*string*>

Mode

Configure

Description

The **vlan forbid ports** command prevents the addition of new ports to a VLAN.

Parameters

<*port-list*> Specifies forbidden ports. You can specify a single port or use commas to specify a list of ports. For example: et.1.3, et.(1-3), (4,6-8).

<*string*> Specifies name of a valid VLAN.

Note: The X-Pedition will display VLAN names up to 32 characters in length.

Restrictions

None.

Example

The following command forbids ports et.1.1 and et.1.2 from VLAN red:

```
xp(config)# vlan forbid ports et.1.(1-2) from red
```

vlan make

Purpose

Configures the specified ports into either trunk or access ports.

Format

vlan make <*port-type*> <*port-list*>

Mode

Configure

Description

The **vlan make** command turns a port into a VLAN trunk or VLAN access port. A VLAN trunk port can forward traffic for multiple VLANs. Use trunk ports when you want to connect X-Pedition switches together and send traffic for multiple VLANs on a single network segment connecting the switches. When you create a trunk port, you must use the **vlan add ports** command to add the trunk port to a VLAN.

Note: When you create a VLAN trunk, make sure the port you use is not already assigned to an existing VLAN (other than the default). After you define the trunk, you must convert it to 802.1Q in order to add it to an existing non-default VLAN.

By default, the default VLAN (VLAN ID 1) is always assigned to all VLAN trunks. Enter the following command from Configure mode to filter default VLAN traffic from a trunk:

Filter default VLAN traffic from a list of trunk ports.	filters add address-filter name NODEFAULTVLAN source-mac ffffff:ffffff source-mac-mask 000000:000000 vlan 1 in-port-list <*port-list*>

Parameters

<*port-type*>
The port type. You can specify one of the following types:

trunk-port
The port will forward traffic for multiple VLANs. The X-Pedition will encapsulate all traffic in IEEE 802.1Q tag headers.

access-port
The port will forward traffic only for the VLANs to which you have added the ports and the traffic will be untagged. This is the default.

<port-list>
The ports you are configuring. You can specify a single port or a comma-separated list of ports. Example: et.1.3,et.(1-3).(4,6-8).

Restrictions

The X-Pedition does not support VLAN trunks that are not assigned to the default VLAN. To filter default VLAN traffic from VLAN trunks, enter the following command:

```
xp# filters add address-filter name NODEFAULTVLAN source-mac ffffff:ffffff source-mac-mask
000000:000000 vlan 1 in-port-list <port-list>
```

vlan multi-add

Purpose

Adds ports to multiple VLANS. Used with the vlan multi-create command.

Format

vlan multi-add ports <*port-list*> **to** <*basename*> **id** <*num*> **through** <*num*>

Mode

Configure

Description

The vlan multi-add ports command adds ports to existing VLANS created with the vlan multi-create command.

Parameters

<*port-list*>	The ports you are adding to the VLANS. You can specify a single port or a comma-separated list of ports. Example: et.1.3,et.(1-3).(4,6-8).
<*basename*>	The basename of the VLANS to add ports to.
id <*num*>	The starting ID of the VLANS to add ports to.
through <*num*>	The ending ID of the VLANS to add ports to.

Restrictions

The VLANS to which you add ports must already exist. To create multiple VLANS, use the vlan multi-create command. Ports must be trunk ports to be used with this command.

Example

The following command adds ports to VLANS RED2, RED3, RED4, and RED5.

```
vlan multi-add ports gi.1.1-2 to RED id 2 through 5
```

vlan multi-create

Purpose

Creates multiple VLANS based on ports or protocol.

Format

vlan multi-create <*basename*> <*type*> **id** <*num*> **through** <*num*>

Mode

Configure

Description

The vlan multi-create command creates multiple VLANS. The name of the VLAN will be the basename appended with the VLAN ID.

Parameters

<*basename*> The basename of the VLAN. VLAN IDS will be appended to the basename for the VLAN name.

Note: The X-Pedition will display VLAN names up to 32 characters in length.

<*type*> The type of VLAN you are adding. The VLAN type determines the types of traffic the X-Pedition will forward on the VLAN. Specify any combination of the first seven types that follow ***or*** specify **port-based**:

ip
Create this VLAN for IP traffic

ipx
Create this VLAN for IPX traffic

appletalk
Create this VLAN for AppleTalk traffic

dec
Create this VLAN for DECnet traffic

sna
Create this VLAN for SNA traffic

ipv6
Create this VLAN for IPv6 traffic

bridged-protocols
Create this VLAN for extended VLAN types (DEC, SNA, Appletalk, IPv6), and non-IP and non-IPX protocols

Note: You can specify a combination of **ip**, **ipx**, **appletalk**, **dec**, **sna**, **ipv6**, and **bridged-protocols**. If you specify *any* of the extended VLAN types (**sna**, **dec**, **appletalk**, **ipv6**) with the **bridged-protocols** option, then all the other extended VLAN types are removed from the VLAN. See the following table:

Configuration Command	Protocols Included in VLAN	Protocols Excluded from VLAN
vlan create <*vlan-name*> ip	**IP**	**IPX, SNA, IPv6, DECnet, Appletalk, Other**
vlan create <*vlan-name*> ip bridged-protocols	**IP, SNA, DECnet, IPv6, Appletalk, Other**	**IPX**
vlan create <*vlan-name*> ip bridged-protocols sna	**IP, SNA, Other**	**IPX, IPv6, DECnet, Appletalk**
vlan create <*vlan-name*> ip bridged-protocols sna ipv6	**IP, SNA, IPv6, Other**	**IPX, DECnet, Appletalk**

port-based
Create this VLAN for all traffic types listed above (port-based VLAN)

Note: You can specify a combination of **ip**, **ipx**, **appletalk**, **dec**, **sna**, **ipv6**, and **bridged-protocols** *or* you can specify **port-based**; you cannot specify **port-based** with any of the other options.

id <*num*> The starting ID of the VLANS to be created. This ID must be smaller than the end ID.

through <*num*> The ending ID of the VLANS to be created.

Restrictions

The following *cannot* be used for VLAN names:

- control
- default
- blackhole
- reserved

- learning
- names starting with an underscore (_) or “sys_”

Note: The X-Pedition will display VLAN names up to 32 characters in length.

Note: Specify both SNA and bridged-protocol to successfully create an SNA based VLAN. The SNA-protocol-based VLAN (implemented in version 3.0 and later) needs to be configured with the following command:

vlan multi-create <*basename*> **sna bridged-protocols id** <*num*> **through** <*num*>

in order to forward all SNA protocol types. Refer to the following Technical Bulletin for more detail: TB0973-1

Example:

The following command creates VLANS RED2, RED3, RED4, and RED5.

```
vlan multi-create RED id 2 through 5
```

vlan show

Purpose

Displays a list of all active VLANs on the X-Pedition.

Format

vlan show

Mode

User or Enable

Description

The **vlan show** command lists all the VLANs that have been configured on the X-Pedition.

Parameters

None.

Restrictions

None.

Chapter 72

web-cache Commands

The **web-cache** commands allow you to transparently redirect HTTP requests to a group of local cache servers. This feature can provide faster user responses and reduce demands for WAN bandwidth.

Command Summary

Table 56 lists the **web-cache** commands. The sections following the table describe the command syntax.

Table 56. web-cache commands

web-cache <*cache-name*> **apply interface** <*interface-name*>
web-cache clear all|cache-name <*cache-name*>
web-cache <*cache-name*> **create bypass-list range** <*ipaddr-range*>**|list** <*ipaddr-list*>**|acl** <*acl-name*>
web-cache <*cache-name*> **create server-list** <*server-list-name*> **range** <*ipaddr-range*>**|list** <*ipaddr-list*>
web-cache <*cache-name*> **permit|deny hosts range** <*ipaddr-range*>**|list** <*ipaddr-list*>**|acl** <*acl-name*>
web-cache <*cache-name*> **set [maximum-connections** <*number*>**]| [http-port** <*port number*>**]| [round-robin {range** <*ipaddr-range*>**| list** <*ipaddr-list*>**}]| [server-options {ping-interval** <*number*>**| ping-attempts** <*number*>**| app-interval** <*number*>**| app-attempts** <*number*> **| no-application-check}]**
web-cache show [all] [cache-name <*cache-name*>**|all] [servers cache** <*cache-name*>**|all]**

web-cache apply interface

Purpose

Applies a caching policy to an interface.

Format

web-cache <*cache-name*> **apply interface** <*interface-name*>

Mode

Configure

Description

The **web-cache apply** command lets you apply a configured cache policy to an outbound interface to start the redirection. The interface to which the cache policy is applied is typically the interface that connects to the Internet. This command redirects outbound HTTP traffic to the cache servers.

Parameters

<*cache-name*>
The name of a cache policy configured with the **web-cache create server-list** command.

<*interface-name*>
The name of the outbound interface that connects to the actual Web server. Typically, this is the interface that connects to the Internet.

Note: Enterasys recommends that you use alphabetic characters when defining interface names—purely numeric interfaces will be interpreted as IP addresses. The X-Pedition will display interface names up to 32 characters in length.

Restrictions

None.

Example

To apply the caching policy ‘websrv1’ to the interface ‘inet2’:

```
xp(config)# web-cache websrv1 apply interface inet2
```

web-cache clear

Purpose

Clears statistics for the specified caching policy.

Format

web-cache clear all|cache-name <*cache-name*>

Mode

Enable

Description

The **web-cache clear** command lets you clear statistics for all caching policies or for specified policies.

Parameters

all
Clears statistics for all caching policies.

cache-name <*cache-name*>

Clears statistics for the specified caching policy.

Restrictions

None.

Examples

To clear statistics for the caching policy ‘websrv1’:

```
xp# web-cache clear cache-name websrv1
```

web-cache create bypass-list

Purpose

Defines the destination sites for which HTTP requests are not redirected to the cache servers, but sent direct.

Format

web-cache <*cache-name*> **create bypass-list range** <*ipaddr-range*>|**list** <*ipaddr-list*>|**acl** <*acl-name*>

Mode

Configure

Description

Certain web sites require authentication of source IP addresses for user access. Requests to these sites cannot be sent to the cache servers. The **web-cache create bypass-list** command allows you to define the destinations to which HTTP requests must be sent directly without redirection to a cache server. You can specify a range of IP addresses, a list of up to four IP addresses, or an ACL that qualifies these hosts.

Parameters

<*cache-name*>
The name of the caching policy for which the specified hosts will not apply.

range <*ipaddr-range*>
A range of host IP addresses in the form “176.89.10.10 176.89.10.50”. This adds the hosts 176.89.10.10, 176.89.10.11, etc., through 176.89.10.50 to the bypass list.

list <*ipaddr-list*>
A list of up to four destination IP addresses in the form “176.89.10.10 176.89.10.11 176.89.10.12”.

acl <*acl-name*>
Name of the ACL profile that defines the packet profile to bypass. The ACL may contain either **permit** or **deny** keywords. The **web-cache create bypass-list** command only looks at the following ACL rule parameter values: protocol, source IP address, destination IP address, source port, destination port, and TOS.

Restrictions

None.

Examples

To specify the hosts 176.89.10.10 and 176.89.10.11 for the bypass list for the caching policy ‘websrv1’:

```
xp(config)# web-cache websrv1 create bypass-list list “176.89.10.10 176.89.10.11”
```

To specify the hosts defined in the ACL ‘nocache’ for the bypass list for the caching policy ‘websrv1’:

```
xp(config)# web-cache websrv1 create bypass-list acl nocache
```

web-cache create server-list

Purpose

Defines the list of servers to be used for caching.

Format

web-cache <*cache-name*> **create server-list** <*server-list-name*> **range** <*ipaddr-range*>|**list** <*ipaddr-list*>

Mode

Configure

Description

The **web-cache create server-list** command allows you to create a group of servers that are used for the specified caching policy. If there are multiple cache servers, load balancing is done based on the destination IP address. If any cache server fails, traffic is redirected to other active servers. You can specify either a range of IP addresses or a list of up to four IP addresses. Note that traffic that is sent from a server in the server list is not redirected.

Parameters

<*cache-name*>
The name of the caching policy.

<*server-list-name*>
The name of this list of servers.

range <*ipaddr-range*>
A range of host IP addresses in the form “176.89.10.10 176.89.10.50”. This adds the hosts 176.89.10.10, 176.89.10.11, etc., through 176.89.10.50 to the server list.

list <*ipaddr-list*>
A list of up to four host IP addresses in the form “176.89.10.10 176.89.10.11 176.89.10.12”.

Restrictions

None.

Examples

To specify the server list ‘servers1’ for the caching policy ‘websrv1’:

```
xp(config)# web-cache websrv1 create server-list servers1 range “10.10.10.10 10.10.10.50”
```

web-cache permit|deny hosts

Purpose

Specifies the hosts whose HTTP requests are redirected to the cache servers.

Format

web-cache <*cache-name*> **permit|deny hosts range** <*ipaddr-range*>|**list** <*ipaddr-list*>|**acl** <*acl-name*>

Mode

Configure

Description

The **web-cache permit** command lets you specify the hosts (users) whose HTTP requests are redirected to the cache servers, while the **web-cache deny** command lets you specify the hosts whose HTTP requests are not redirected to the cache servers. If no **permit** command is specified, all HTTP requests are redirected to the cache servers. You can specify a range of IP addresses, a list of up to four IP addresses, or an ACL that qualifies these hosts.

Parameters

<*cache-name*>
The name of the cache.

range <*ipaddr-range*>
A range of host IP addresses in the form “176.89.10.10 176.89.10.50”.

list <*ipaddr-list*>
A list of up to four host IP addresses in the form “176.89.10.10 176.89.10.11 176.89.10.12”.

acl <*acl-name*>
Name of the ACL profile to be used. This defines the profile of the packets to be permitted or denied. The **web-cache permit/deny** command only looks at the following ACL rule parameter values: protocol, source IP address, destination IP address, source port, destination port, and TOS.

Restrictions

None.

Examples

To allow the HTTP requests of certain hosts to be redirected to the cache servers:

```
xp(config)# web-cache websrv1 permit hosts range “10.10.20.10 10.10.20.50”
```

To specify that the HTTP requests of certain hosts not be redirected to the cache servers:

```
xp(config)# web-cache websrv1 deny hosts list “10.10.20.61 10.10.20.75”
```

web-cache set

Format

web-cache <*cache-name*> **set [maximum-connections** <*number*>**]| [http-port** <*port number*>**]| [round-robin {range** <*ipaddr-range*>**| list** <*ipaddr-list*>**}]| [server-options {ping-interval** <*number*>**| ping-attempts** <*number*>**| app-interval** <*number*>**| app-attempts** <*number*> **| no-application-check}]**

Mode

Configure

Description

Web caching provides a way to store frequently accessed Web objects on a cache of local servers. Each HTTP request is transparently redirected by the X-Pedition to a configured cache server. The first time a user accesses a Web object, the object is stored on a cache server—each subsequent request for the object uses this cached object. Web caching allows multiple users to access Web objects stored on local servers with a much faster response time than accessing the same objects over a WAN connection. This can also result in substantial cost savings by reducing the WAN bandwidth usage.

Note: The X-Pedition itself does not act as cache for web objects. It redirects HTTP requests to local servers on which the web objects are cached. One or more local servers are needed to work as cache servers with the RS.s web caching function.

The **web-cache set** command allows users to specify the behavior of a web caching server group.

Parameters

maximum-connections <*number*>
The limit of connections to support for a web caching server group. This number is the maximum number of connections allowed for each server in a list of web caching servers. (This list must already have been created with the **web-cache create server-list** command.)

http-port <*port number*>
Some networks use proxy servers that listen for HTTP requests on a non-standard port number. The **http-port** parameter lets you specify the port number used by the proxy server for HTTP requests. Specify a value between 1 and 65535 (the default is 80).

round-robin If a certain web site is accessed very frequently, the cache server that services HTTP requests to this web site can become overloaded with user requests. The **round-robin** parameter allows users to distribute destination IP addresses for HTTP requests across cache servers in a round-robin manner. If a cache server fails, the address range associated with that server is redistributed among the

remaining servers.

range <*ipaddr-range*>
A range of host IP addresses in the form “176.89.10.10 176.89.10.50”.

list <*ipaddr-list*>
A list of up to four destination IP addresses in the form “176.89.10.10 176.89.10.11 176.89.10.12”.

server-options Use the server-options command to set various parameters for a group of web cache servers.

ping-interval <*number*>
Set interval (in seconds, from 1 to 3600) for ping checks to this server-list.

ping-attempts <*number*>
Set number (from 1 to 255) of failed ping attempts before server is considered down.

app-interval <*number*>
Set interval (in seconds, from 1 to 3600) for application checks to this server-list.

app-attempts <*number*>
Set the number (from 1 to 255) of failed application attempts before application is considered down.

no-application-check
Disable TCP application health checking for this server-list.

Restrictions

None.

Examples

To limit the number of connections for servers in the server list *servers1* to 1000 connections:

```
xp(config)# web-cache set maximum-connections servers1 1000
```

To set the port number for HTTP requests:

```
xp(config)# web-cache websvr1 set http-port 100
```

To specify destination IP addresses to be distributed across the caching policy ‘websvr1’ servers:

```
xp(config)# web-cache set round-robin list “176.20.20.10 176.20.50.60”
```

To ping the servers in the list *service2* in the cache group *websvr1* every 10 seconds:

```
xp(config)# web-cache websvr1 set server-options service2 ping-int 10
```

web-cache show

Purpose

Displays information about caching policies.

Format

web-cache show [all] [cache-name <*cache-name*>**|all] [servers cache** <*cache-name*>**|all]**

Mode

Enable

Description

The **web-cache show** command allows you to display web caching information for specific caching policies or server lists.

Parameters

all
Displays all web cache information for all caching policies and all server lists.

cache-name <*cache-name*>|**all**
Displays web cache information for the specified caching policy. **all** displays all caching policies.

servers cache <*cache-name*>|**all**
Displays information for the servers configured for the specified caching policy. **all** displays all configured cache servers.

Restrictions

None.

Examples

To display web cache information for all caching policies and server lists:

```
xp# web-cache show all

web-cache show all" sample output:

--------------------------------------------------------------------------------
Cache Name          : cachename
Applied Interfaces  : none
HTTP Port           : 80
Bypass list         : none

ACL         Source IP/Mask     Dest. IP/Mask      SrcPort   DstPort   TOS   TOS-MASK    Prot
---          --------------          -------------          -------       -------      ---      --------            ----

PI: Ping Check Interval
PA: Ping Check Attempts before server is considered down
AC: Application Checking Admin Status
AI: Application Check Interval
AA: Application Check Attempts before application is considered down

Server List  PI   PA   AC  AI   AA   Max con IP address(es)
-----------  ---- ---- --- ---- ---- ------- --------------
group1       5    4    On  15   4    2000    10.10.10.1, 10.10.10.2
group2       10   4    Off n/a  n/a  2000    10.10.20.1

Access Users
------ -----
Permit All Users
```

To display web cache information for a specific caching policy:

```
xp# web-cache show cache-name cache1
Cache Name : cache1 ❶
Applied Interfaces : ip1 ❷
Bypass list : none ❸
HTTP Port : 80 ❹

 ❺          ❻              ❼          ❽        ❾   ❿ ⓫
ACL     Source IP/Mask   Dest. IP/Mask   SrcPort   DstPort   TOS Port
---     --------------   --------------  --------- --------- --- ----
deny207  172.89.1.1/32    207.135.0.0/16  any       http      0   IP

  ⓬      ⓭        ⓮
Server   Max con IP address
---- --  ------- ----------
s1       2000    176.89.10.50 - 176.89.10.60

Access Users ⓯
------ -----
Permit All Users
Deny   profile deny207
```

Legend:

1. The name of the cache policy.
2. The outbound interface where the cache policy was applied, typically an interface that connects to the Internet.
3. Destination sites for which HTTP requests are *not* redirected to cache servers and are sent direct.
4. The HTTP port used by a proxy server. A port number other than 80 can be specified with the **web-cache set http-port** command.
5. The names of the profiles (created with an **acl** statement) associated with this cache policy.
6. The source address and filtering mask.
7. The destination address and filtering mask.
8. The source port.
9. The destination port.
10. The TOS value in the packet.
11. The protocol.
12. The server list name.
13. The maximum number of connections that can be handled by each server in the server list.
14. The list or range of IP addresses of the servers in the server list.

15. The hosts (users) whose HTTP requests *are* redirected to the cache servers and the hosts whose HTTP requests are *not* redirected to the cache servers. If no **permit** command is specified, all HTTP requests are redirected to the cache servers.

To display information for a specific web cache servers:

```
xp# web-cache show servers cache cache1
Cache name : cache1 ❶

 ❷     ❸          ❹       ❺       ❻
Block IP address    Max Conn  Used Cnt  Status
----- ----------    --------  --------  ------
s1    176.89.10.50  2000      0         Down
s1    176.89.10.51  2000      0         Down
s1    176.89.10.52  2000      0         Down
s1    176.89.10.53  2000      0         Down
s1    176.89.10.54  2000      0         Down
s1    176.89.10.55  2000      0         Down
s1    176.89.10.56  2000      0         Down
s1    176.89.10.57  2000      0         Down
s1    176.89.10.58  2000      0         Down
s1    176.89.10.59  2000      0         Down
s1    176.89.10.60  2000      0         Down
```

Legend:

1. The name of the cache policy.
2. The server list name.
3. The IP address of a server in the server list.
4. The maximum number of connections that can be handled by the server.
5. The number of connections currently being handled by the server.
6. The current status of the server.

To display information for all configured web cache servers:

```
xp# web-cache show servers cache all

Cache name       : cachename

Block      IP address       Max Conn    Used Cnt    Server      Application
---------- ----------       --------    --------     ------      -----------
group1     10.10.10.1       2000          0            Up        Up
group1     10.10.10.2       2000          0            Up        Down
group2     10.10.20.1       2000          0            Up        n/a
```

Appendix A

RMON 2 Protocol Directory

This appendix lists the protocol encapsulations that can be managed with the RMON 2 Protocol Directory group on the X-Pedition. You can specify protocol encapsulations with the **rmon set protocol-directory** or **rmon show protocol-directory** commands. For example, ether2.ipx specifies IPX over Ethernet II, while *ether2.ipx specifies IPX over any link layer protocol. The protocol object IDs are defined in RFC 2074.

The protocols are listed in the following order:

Protocol Encapsulation	Protocol Identifier (Object ID)
Ethernet Applications	
ether2.idp	8.0.0.0.1.0.0.6.0.2.0.0
ether2.ip-v4	8.0.0.0.1.0.0.8.0.2.0.0
ether2.chaosnet	8.0.0.0.1.0.0.8.4.2.0.0
ether2.arp	8.0.0.0.1.0.0.8.6.2.0.0
ether2.vip	8.0.0.0.1.0.0.11.173.2.0.0
ether2.vloop	8.0.0.0.1.0.0.11.174.2.0.0
ether2.vecho	8.0.0.0.1.0.0.11.175.2.0.0
ether2.netbios-3com	8.0.0.0.1.0.0.60.0.2.0.0
ether2.dec	8.0.0.0.1.0.0.96.0.2.0.0
ether2.mop	8.0.0.0.1.0.0.96.1.2.0.0
ether2.mop2	8.0.0.0.1.0.0.96.2.2.0.0
ether2.drp	8.0.0.0.1.0.0.96.3.2.0.0
ether2.lat	8.0.0.0.1.0.0.96.4.2.0.0
ether2.dec-diag	8.0.0.0.1.0.0.96.5.2.0.0
ether2.lavc	8.0.0.0.1.0.0.96.7.2.0.0
ether2.rarp	8.0.0.0.1.0.0.128.53.2.0.0
ether2.atalk	8.0.0.0.1.0.0.128.155.2.0.0
ether2.vloop2	8.0.0.0.1.0.0.128.196.2.0.0
ether2.vecho2	8.0.0.0.1.0.0.128.197.2.0.0
ether2.sna-th	8.0.0.0.1.0.0.128.213.2.0.0
ether2.aarp	8.0.0.0.1.0.0.128.243.2.0.0
ether2.ipx	8.0.0.0.1.0.0.129.55.2.0.0
ether2.snmp	8.0.0.0.1.0.0.129.76.2.0.0
ether2.ip-v6	8.0.0.0.1.0.0.134.221.2.0.0
ether2.loopback	8.0.0.0.1.0.0.144.0.2.0.0
*ether2.ip-v4	8.1.0.0.1.0.0.8.0.2.0.1
*ether2.ipx	8.1.0.0.1.0.0.129.55.2.0.0

Protocol Encapsulation	Protocol Identifier (Object ID)
IP (version 4) Applications	
*ether2.ip-v4.icmp	12.1.0.0.1.0.0.8.0.0.0.0.0.1.3.0.1.0
*ether2.ip-v4.igmp	12.1.0.0.1.0.0.8.0.0.0.0.0.2.3.0.1.0
*ether2.ip-v4.ggp	12.1.0.0.1.0.0.8.0.0.0.0.0.3.3.0.1.0
*ether2.ip-v4.ipip4	12.1.0.0.1.0.0.8.0.0.0.0.0.4.3.0.1.0
*ether2.ip-v4.st	12.1.0.0.1.0.0.8.0.0.0.0.0.5.3.0.1.0
*ether2.ip-v4.tcp	12.1.0.0.1.0.0.8.0.0.0.0.0.6.3.0.1.0
*ether2.ip-v4.ucl	12.1.0.0.1.0.0.8.0.0.0.0.0.7.3.0.1.0
*ether2.ip-v4.egp	12.1.0.0.1.0.0.8.0.0.0.0.0.8.3.0.1.0
*ether2.ip-v4.igp	12.1.0.0.1.0.0.8.0.0.0.0.0.9.3.0.1.0
*ether2.ip-v4.bbn-rcc-mon	12.1.0.0.1.0.0.8.0.0.0.0.0.10.3.0.1.0
*ether2.ip-v4.nvp2	12.1.0.0.1.0.0.8.0.0.0.0.0.11.3.0.1.0
*ether2.ip-v4.pup	12.1.0.0.1.0.0.8.0.0.0.0.0.12.3.0.1.0
*ether2.ip-v4.argus	12.1.0.0.1.0.0.8.0.0.0.0.0.13.3.0.1.0
*ether2.ip-v4.emcon	12.1.0.0.1.0.0.8.0.0.0.0.0.14.3.0.1.0
*ether2.ip-v4.xnet	12.1.0.0.1.0.0.8.0.0.0.0.0.15.3.0.1.0
*ether2.ip-v4.chaos	12.1.0.0.1.0.0.8.0.0.0.0.0.16.3.0.1.0
*ether2.ip-v4.udp	12.1.0.0.1.0.0.8.0.0.0.0.0.17.3.0.1.0
*ether2.ip-v4.mux	12.1.0.0.1.0.0.8.0.0.0.0.0.18.3.0.1.0
*ether2.ip-v4.dcn-meas	12.1.0.0.1.0.0.8.0.0.0.0.0.19.3.0.1.0
*ether2.ip-v4.hmp	12.1.0.0.1.0.0.8.0.0.0.0.0.20.3.0.1.0
*ether2.ip-v4.prm	12.1.0.0.1.0.0.8.0.0.0.0.0.21.3.0.1.0
*ether2.ip-v4.xns-idp	12.1.0.0.1.0.0.8.0.0.0.0.0.22.3.0.1.0
*ether2.ip-v4.trunk-1	12.1.0.0.1.0.0.8.0.0.0.0.0.23.3.0.1.0
*ether2.ip-v4.trunk-2	12.1.0.0.1.0.0.8.0.0.0.0.0.24.3.0.1.0
*ether2.ip-v4.leaf-1	12.1.0.0.1.0.0.8.0.0.0.0.0.25.3.0.1.0
*ether2.ip-v4.leaf-2	12.1.0.0.1.0.0.8.0.0.0.0.0.26.3.0.1.0
*ether2.ip-v4.rdp	12.1.0.0.1.0.0.8.0.0.0.0.0.27.3.0.1.0
*ether2.ip-v4.irtp	12.1.0.0.1.0.0.8.0.0.0.0.0.28.3.0.1.0
*ether2.ip-v4.iso-tp4	12.1.0.0.1.0.0.8.0.0.0.0.0.29.3.0.1.0

Protocol Encapsulation	Protocol Identifier (Object ID)
*ether2.ip-v4.netbit	12.1.0.0.1.0.0.8.0.0.0.0.30.3.0.1.0
*ether2.ip-v4.mfe-nsp	12.1.0.0.1.0.0.8.0.0.0.0.31.3.0.1.0
*ether2.ip-v4.merit-inp	12.1.0.0.1.0.0.8.0.0.0.0.32.3.0.1.0
*ether2.ip-v4.sep	12.1.0.0.1.0.0.8.0.0.0.0.33.3.0.1.0
*ether2.ip-v4.third-pc	12.1.0.0.1.0.0.8.0.0.0.0.34.3.0.1.0
*ether2.ip-v4.idpr	12.1.0.0.1.0.0.8.0.0.0.0.35.3.0.1.0
*ether2.ip-v4.xtp	12.1.0.0.1.0.0.8.0.0.0.0.36.3.0.1.0
*ether2.ip-v4.ddp	12.1.0.0.1.0.0.8.0.0.0.0.37.3.0.1.0
*ether2.ip-v4.idpr-cmtp	12.1.0.0.1.0.0.8.0.0.0.0.38.3.0.1.0
*ether2.ip-v4.tp-plus-plus	12.1.0.0.1.0.0.8.0.0.0.0.39.3.0.1.0
*ether2.ip-v4.il	12.1.0.0.1.0.0.8.0.0.0.0.40.3.0.1.0
*ether2.ip-v4.sip	12.1.0.0.1.0.0.8.0.0.0.0.41.3.0.1.0
*ether2.ip-v4.sdrp	12.1.0.0.1.0.0.8.0.0.0.0.42.3.0.1.0
*ether2.ip-v4.sip-sr	12.1.0.0.1.0.0.8.0.0.0.0.43.3.0.1.0
*ether2.ip-v4.sip-frag	12.1.0.0.1.0.0.8.0.0.0.0.44.3.0.1.0
*ether2.ip-v4.idrp	12.1.0.0.1.0.0.8.0.0.0.0.45.3.0.1.0
*ether2.ip-v4.rsvp	12.1.0.0.1.0.0.8.0.0.0.0.46.3.0.1.0
*ether2.ip-v4.gre	12.1.0.0.1.0.0.8.0.0.0.0.47.3.0.1.0
*ether2.ip-v4.mhrp	12.1.0.0.1.0.0.8.0.0.0.0.48.3.0.1.0
*ether2.ip-v4.bna	12.1.0.0.1.0.0.8.0.0.0.0.49.3.0.1.0
*ether2.ip-v4.sipp-esp	12.1.0.0.1.0.0.8.0.0.0.0.50.3.0.1.0
*ether2.ip-v4.sipp-ah	12.1.0.0.1.0.0.8.0.0.0.0.51.3.0.1.0
*ether2.ip-v4.i-nlsp	12.1.0.0.1.0.0.8.0.0.0.0.52.3.0.1.0
*ether2.ip-v4.swipe	12.1.0.0.1.0.0.8.0.0.0.0.53.3.0.1.0
*ether2.ip-v4.nhrp	12.1.0.0.1.0.0.8.0.0.0.0.54.3.0.1.0
*ether2.ip-v4.priv-host	12.1.0.0.1.0.0.8.0.0.0.0.61.3.0.1.0
*ether2.ip-v4.cftp	12.1.0.0.1.0.0.8.0.0.0.0.62.3.0.1.0
*ether2.ip-v4.priv-net	12.1.0.0.1.0.0.8.0.0.0.0.63.3.0.1.0
*ether2.ip-v4.sat-expak	12.1.0.0.1.0.0.8.0.0.0.0.64.3.0.1.0
*ether2.ip-v4.kryptolan	12.1.0.0.1.0.0.8.0.0.0.0.65.3.0.1.0

Protocol Encapsulation	Protocol Identifier (Object ID)
*ether2.ip-v4.rvd	12.1.0.0.1.0.0.8.0.0.0.0.66.3.0.1.0
*ether2.ip-v4.ippc	12.1.0.0.1.0.0.8.0.0.0.0.67.3.0.1.0
*ether2.ip-v4.priv-distfile	12.1.0.0.1.0.0.8.0.0.0.0.68.3.0.1.0
*ether2.ip-v4.sat-mon	12.1.0.0.1.0.0.8.0.0.0.0.69.3.0.1.0
*ether2.ip-v4.visa	12.1.0.0.1.0.0.8.0.0.0.0.70.3.0.1.0
*ether2.ip-v4.ipcv	12.1.0.0.1.0.0.8.0.0.0.0.71.3.0.1.0
*ether2.ip-v4.cpnx	12.1.0.0.1.0.0.8.0.0.0.0.72.3.0.1.0
*ether2.ip-v4.cphb	12.1.0.0.1.0.0.8.0.0.0.0.73.3.0.1.0
*ether2.ip-v4.wsn	12.1.0.0.1.0.0.8.0.0.0.0.74.3.0.1.0
*ether2.ip-v4.pvp	12.1.0.0.1.0.0.8.0.0.0.0.75.3.0.1.0
*ether2.ip-v4.br-sat-mon	12.1.0.0.1.0.0.8.0.0.0.0.76.3.0.1.0
*ether2.ip-v4.sun-nd	12.1.0.0.1.0.0.8.0.0.0.0.77.3.0.1.0
*ether2.ip-v4.wb-mon	12.1.0.0.1.0.0.8.0.0.0.0.78.3.0.1.0
*ether2.ip-v4.wb-expak	12.1.0.0.1.0.0.8.0.0.0.0.79.3.0.1.0
*ether2.ip-v4.iso-ip	12.1.0.0.1.0.0.8.0.0.0.0.80.3.0.1.0
*ether2.ip-v4.vmtp	12.1.0.0.1.0.0.8.0.0.0.0.81.3.0.1.0
*ether2.ip-v4.secure-mvtp	12.1.0.0.1.0.0.8.0.0.0.0.82.3.0.1.0
*ether2.ip-v4.vines	12.1.0.0.1.0.0.8.0.0.0.0.83.3.0.1.0
*ether2.ip-v4.ttp	12.1.0.0.1.0.0.8.0.0.0.0.84.3.0.1.0
*ether2.ip-v4.nfsnet-igp	12.1.0.0.1.0.0.8.0.0.0.0.85.3.0.1.0
*ether2.ip-v4.dgp	12.1.0.0.1.0.0.8.0.0.0.0.86.3.0.1.0
*ether2.ip-v4.tcf	12.1.0.0.1.0.0.8.0.0.0.0.87.3.0.1.0
*ether2.ip-v4.igrp	12.1.0.0.1.0.0.8.0.0.0.0.88.3.0.1.0
*ether2.ip-v4.ospf	12.1.0.0.1.0.0.8.0.0.0.0.89.3.0.1.0
*ether2.ip-v4.sprite-rpc	12.1.0.0.1.0.0.8.0.0.0.0.90.3.0.1.0
*ether2.ip-v4.larp	12.1.0.0.1.0.0.8.0.0.0.0.91.3.0.1.0
*ether2.ip-v4.mtp	12.1.0.0.1.0.0.8.0.0.0.0.92.3.0.1.0
*ether2.ip-v4.ax-25	12.1.0.0.1.0.0.8.0.0.0.0.93.3.0.1.0
*ether2.ip-v4.ipip	12.1.0.0.1.0.0.8.0.0.0.0.94.3.0.1.0
*ether2.ip-v4.micp	12.1.0.0.1.0.0.8.0.0.0.0.95.3.0.1.0

Protocol Encapsulation	Protocol Identifier (Object ID)
*ether2.ip-v4.scc-sp	12.1.0.0.1.0.0.8.0.0.0.0.96.3.0.1.0
*ether2.ip-v4.etherip	12.1.0.0.1.0.0.8.0.0.0.0.97.3.0.1.0
*ether2.ip-v4.encap	12.1.0.0.1.0.0.8.0.0.0.0.98.3.0.1.0
*ether2.ip-v4.priv-encript	12.1.0.0.1.0.0.8.0.0.0.0.99.3.0.1.0
*ether2.ip-v4.gmtp	12.1.0.0.1.0.0.8.0.0.0.0.100.3.0.1.0
IPX Applications	
*ether2.ipx.nov-pep	12.1.0.0.1.0.0.129.55.0.0.0.0.3.0.0.0
*ether2.ipx.nov-pep.ncp	16.1.0.0.1.0.0.129.55.0.0.0.0.0.0.4.81.4.0.0.0.0
*ether2.ipx.nov-pep.nov-sap	16.1.0.0.1.0.0.129.55.0.0.0.0.0.0.4.82.4.0.0.0.0
*ether2.ipx.nov-pep.nov-rip	16.1.0.0.1.0.0.129.55.0.0.0.0.0.0.4.83.4.0.0.0.0
*ether2.ipx.nov-pep.nov-netbios	16.1.0.0.1.0.0.129.55.0.0.0.0.0.0.4.85.4.0.0.0.0
*ether2.ipx.nov-pep.nov-diag	16.1.0.0.1.0.0.129.55.0.0.0.0.0.0.4.86.4.0.0.0.0
*ether2.ipx.nov-pep.nov-sec	16.1.0.0.1.0.0.129.55.0.0.0.0.0.0.4.87.4.0.0.0.0
*ether2.ipx.nov-pep.smb	16.1.0.0.1.0.0.129.55.0.0.0.0.0.0.5.80.4.0.0.0.0
*ether2.ipx.nov-pep.smb2	16.1.0.0.1.0.0.129.55.0.0.0.0.0.0.5.82.4.0.0.0.0
*ether2.ipx.nov-pep.burst	16.1.0.0.1.0.0.129.55.0.0.0.0.0.0.13.5.4.0.0.0.0
*ether2.ipx.nov-pep.nov-watchdog	16.1.0.0.1.0.0.129.55.0.0.0.0.0.0.64.4.4.0.0.0.0
*ether2.ipx.nov-pep.nov-bcast	16.1.0.0.1.0.0.129.55.0.0.0.0.0.0.64.5.4.0.0.0.0
*ether2.ipx.nov-pep.nlsp	16.1.0.0.1.0.0.129.55.0.0.0.0.0.0.144.1.4.0.0.0.0
*ether2.ipx.nov-pep.snmp	16.1.0.0.1.0.0.129.55.0.0.0.0.0.0.144.15.4.0.0.0.0
*ether2.ipx.nov-pep.snmptrap	16.1.0.0.1.0.0.129.55.0.0.0.0.0.0.144.16.4.0.0.0.0
*ether2.ipx.nov-rip	12.1.0.0.1.0.0.129.55.0.0.0.1.3.0.0.0
*ether2.ipx.nov-echo	12.1.0.0.1.0.0.129.55.0.0.0.2.3.0.0.0
*ether2.ipx.nov-error	12.1.0.0.1.0.0.129.55.0.0.0.3.3.0.0.0
*ether2.ipx.nov-pep2	12.1.0.0.1.0.0.129.55.0.0.0.4.3.0.0.0
*ether2.ipx.nov-spx	12.1.0.0.1.0.0.129.55.0.0.0.5.3.0.0.0
*ether2.ipx.nov-pep3	12.1.0.0.1.0.0.129.55.0.0.0.17.3.0.0.0
*ether2.ipx.nov-netbios	12.1.0.0.1.0.0.129.55.0.0.0.20.3.0.0.0

Protocol Encapsulation	Protocol Identifier (Object ID)
TCP Applications	
*ether2.ip-v4.tcp.tcpmux	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.1.4.0.1.0.0
*ether2.ip-v4.tcp.compressnet-mgmt	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.2.4.0.1.0.0
*ether2.ip-v4.tcp.compressnet	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.3.4.0.1.0.0
*ether2.ip-v4.tcp.rje	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.5.4.0.1.0.0
*ether2.ip-v4.tcp.echo	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.7.4.0.1.0.0
*ether2.ip-v4.tcp.discard	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.9.4.0.1.0.0
*ether2.ip-v4.tcp.systat	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.11.4.0.1.0.0
*ether2.ip-v4.tcp.daytime	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.13.4.0.1.0.0
*ether2.ip-v4.tcp.qotd	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.17.4.0.1.0.0
*ether2.ip-v4.tcp.msp	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.18.4.0.1.0.0
*ether2.ip-v4.tcp.chargen	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.19.4.0.1.0.0
*ether2.ip-v4.tcp.ftp-data	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.20.4.0.1.0.0
*ether2.ip-v4.tcp.ftp	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.21.4.0.1.0.0
*ether2.ip-v4.tcp.telnet	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.23.4.0.1.0.0
*ether2.ip-v4.tcp.priv-mail	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.24.4.0.1.0.0
*ether2.ip-v4.tcp.smtp	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.25.4.0.1.0.0
*ether2.ip-v4.tcp.nsw-fe	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.27.4.0.1.0.0
*ether2.ip-v4.tcp.msg-icp	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.29.4.0.1.0.0
*ether2.ip-v4.tcp.msg-auth	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.31.4.0.1.0.0
*ether2.ip-v4.tcp.dsp	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.33.4.0.1.0.0
*ether2.ip-v4.tcp.priv-print	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.35.4.0.1.0.0
*ether2.ip-v4.tcp.time	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.37.4.0.1.0.0
*ether2.ip-v4.tcp.rap	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.38.4.0.1.0.0
*ether2.ip-v4.tcp.graphics	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.41.4.0.1.0.0
*ether2.ip-v4.tcp.nicname	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.43.4.0.1.0.0
*ether2.ip-v4.tcp.mpm-flags	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.44.4.0.1.0.0
*ether2.ip-v4.tcp.mpm	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.45.4.0.1.0.0
*ether2.ip-v4.tcp.mpm-send	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.46.4.0.1.0.0
*ether2.ip-v4.tcp.ni-ftp	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.47.4.0.1.0.0

Protocol Encapsulation	Protocol Identifier (Object ID)
*ether2.ip-v4.tcp.auditd	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.48.4.0.1.0.0
*ether2.ip-v4.tcp.tacacs	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.49.4.0.1.0.0
*ether2.ip-v4.tcp.xns-time	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.52.4.0.1.0.0
*ether2.ip-v4.tcp.domain	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.53.4.0.1.0.0
*ether2.ip-v4.tcp.xns-ch	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.54.4.0.1.0.0
*ether2.ip-v4.tcp.isi-gl	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.55.4.0.1.0.0
*ether2.ip-v4.tcp.xns-auth	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.56.4.0.1.0.0
*ether2.ip-v4.tcp.priv-term	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.57.4.0.1.0.0
*ether2.ip-v4.tcp.xns-mail	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.58.4.0.1.0.0
*ether2.ip-v4.tcp.priv-file	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.59.4.0.1.0.0
*ether2.ip-v4.tcp.ni-mail	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.61.4.0.1.0.0
*ether2.ip-v4.tcp.acas	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.62.4.0.1.0.0
*ether2.ip-v4.tcp.covia	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.64.4.0.1.0.0
*ether2.ip-v4.tcp.tacacs-ds	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.65.4.0.1.0.0
*ether2.ip-v4.tcp.sql*net	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.66.4.0.1.0.0
*ether2.ip-v4.tcp.gopher	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.70.4.0.1.0.0
*ether2.ip-v4.tcp.netrjs-1	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.71.4.0.1.0.0
*ether2.ip-v4.tcp.netrjs-2	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.72.4.0.1.0.0
*ether2.ip-v4.tcp.netrjs-3	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.73.4.0.1.0.0
*ether2.ip-v4.tcp.netrjs-4	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.74.4.0.1.0.0
*ether2.ip-v4.tcp.priv-dialout	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.75.4.0.1.0.0
*ether2.ip-v4.tcp.deos	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.76.4.0.1.0.0
*ether2.ip-v4.tcp.priv-rje	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.77.4.0.1.0.0
*ether2.ip-v4.tcp.vettcp	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.78.4.0.1.0.0
*ether2.ip-v4.tcp.finger	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.79.4.0.1.0.0
*ether2.ip-v4.tcp.www-http	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.80.4.0.1.0.0
*ether2.ip-v4.tcp.hosts2-ns	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.81.4.0.1.0.0
*ether2.ip-v4.tcp.xfer	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.82.4.0.1.0.0
*ether2.ip-v4.tcp.mit-ml-dev	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.83.4.0.1.0.0
*ether2.ip-v4.tcp.ctf	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.84.4.0.1.0.0

Protocol Encapsulation	Protocol Identifier (Object ID)
*ether2.ip-v4.tcp.mit-ml-dev	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.85.4.0.1.0.0
*ether2.ip-v4.tcp.mfcobol	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.86.4.0.1.0.0
*ether2.ip-v4.tcp.priv-termlink	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.87.4.0.1.0.0
*ether2.ip-v4.tcp.kerberos	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.88.4.0.1.0.0
*ether2.ip-v4.tcp.su-mit-tg	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.89.4.0.1.0.0
*ether2.ip-v4.tcp.dnsix	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.90.4.0.1.0.0
*ether2.ip-v4.tcp.mit-dov	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.91.4.0.1.0.0
*ether2.ip-v4.tcp.npp	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.92.4.0.1.0.0
*ether2.ip-v4.tcp.dcp	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.93.4.0.1.0.0
*ether2.ip-v4.tcp.objcall	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.94.4.0.1.0.0
*ether2.ip-v4.tcp.supdup	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.95.4.0.1.0.0
*ether2.ip-v4.tcp.dixie	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.96.4.0.1.0.0
*ether2.ip-v4.tcp.swift-rvf	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.97.4.0.1.0.0
*ether2.ip-v4.tcp.tacnews	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.98.4.0.1.0.0
*ether2.ip-v4.tcp.metagram	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.99.4.0.1.0.0
*ether2.ip-v4.tcp.newacct	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.100.4.0.1.0.0
*ether2.ip-v4.tcp.hostname	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.101.4.0.1.0.0
*ether2.ip-v4.tcp.iso-tsap	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.102.4.0.1.0.0
*ether2.ip-v4.tcp.gppitnp	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.103.4.0.1.0.0
*ether2.ip-v4.tcp.acr-nema	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.104.4.0.1.0.0
*ether2.ip-v4.tcp.csnet-ns	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.105.4.0.1.0.0
*ether2.ip-v4.tcp.3com-tsmux	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.106.4.0.1.0.0
*ether2.ip-v4.tcp.rtelnet	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.107.4.0.1.0.0
*ether2.ip-v4.tcp.snagas	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.108.4.0.1.0.0
*ether2.ip-v4.tcp.pop2	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.109.4.0.1.0.0
*ether2.ip-v4.tcp.pop3	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.110.4.0.1.0.0
*ether2.ip-v4.tcp.sunrpc	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.111.4.0.1.0.0
*ether2.ip-v4.tcp.mcidas	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.112.4.0.1.0.0
*ether2.ip-v4.tcp.auth	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.113.4.0.1.0.0
*ether2.ip-v4.tcp.audionews	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.114.4.0.1.0.0

Protocol Encapsulation	Protocol Identifier (Object ID)
*ether2.ip-v4.tcp.sftp	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.115.4.0.1.0.0
*ether2.ip-v4.tcp.ansanotify	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.116.4.0.1.0.0
*ether2.ip-v4.tcp.uucp-path	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.117.4.0.1.0.0
*ether2.ip-v4.tcp.sqlserv	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.118.4.0.1.0.0
*ether2.ip-v4.tcp.nntp	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.119.4.0.1.0.0
*ether2.ip-v4.tcp.erpc	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.121.4.0.1.0.0
*ether2.ip-v4.tcp.smakynet	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.122.4.0.1.0.0
*ether2.ip-v4.tcp.ansatrader	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.124.4.0.1.0.0
*ether2.ip-v4.tcp.locus-map	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.125.4.0.1.0.0
*ether2.ip-v4.tcp.unitary	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.126.4.0.1.0.0
*ether2.ip-v4.tcp.locus-con	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.127.4.0.1.0.0
*ether2.ip-v4.tcp.gss-xlicen	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.128.4.0.1.0.0
*ether2.ip-v4.tcp.pwdgen	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.129.4.0.1.0.0
*ether2.ip-v4.tcp.cisco-fna	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.130.4.0.1.0.0
*ether2.ip-v4.tcp.cisco-tna	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.131.4.0.1.0.0
*ether2.ip-v4.tcp.cisco-sys	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.132.4.0.1.0.0
*ether2.ip-v4.tcp.statsrv	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.133.4.0.1.0.0
*ether2.ip-v4.tcp.ingres-net	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.134.4.0.1.0.0
*ether2.ip-v4.tcp.loc-srv	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.135.4.0.1.0.0
*ether2.ip-v4.tcp.profile	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.136.4.0.1.0.0
*ether2.ip-v4.tcp.netbios-ns	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.137.4.0.1.0.0
*ether2.ip-v4.tcp.netbios-dgm	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.138.4.0.1.0.0
*ether2.ip-v4.tcp.netbios-ssn	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.139.4.0.1.0.0
*ether2.ip-v4.tcp.emfis-data	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.140.4.0.1.0.0
*ether2.ip-v4.tcp.emfis-cntl	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.141.4.0.1.0.0
*ether2.ip-v4.tcp.bl-idm	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.142.4.0.1.0.0
*ether2.ip-v4.tcp.imap2	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.143.4.0.1.0.0
*ether2.ip-v4.tcp.news	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.144.4.0.1.0.0
*ether2.ip-v4.tcp.uaac	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.145.4.0.1.0.0
*ether2.ip-v4.tcp.iso-tp0	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.146.4.0.1.0.0

Protocol Encapsulation	Protocol Identifier (Object ID)
*ether2.ip-v4.tcp.iso-ip	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.147.4.0.1.0.0
*ether2.ip-v4.tcp.cronus	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.148.4.0.1.0.0
*ether2.ip-v4.tcp.aed-512	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.149.4.0.1.0.0
*ether2.ip-v4.tcp.sql-net	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.150.4.0.1.0.0
*ether2.ip-v4.tcp.hems	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.151.4.0.1.0.0
*ether2.ip-v4.tcp.bftp	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.152.4.0.1.0.0
*ether2.ip-v4.tcp.netsc-prod	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.154.4.0.1.0.0
*ether2.ip-v4.tcp.netsc-dev	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.155.4.0.1.0.0
*ether2.ip-v4.tcp.sqlsrv	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.156.4.0.1.0.0
*ether2.ip-v4.tcp.knet-cmp	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.157.4.0.1.0.0
*ether2.ip-v4.tcp.pcmail-srv	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.158.4.0.1.0.0
*ether2.ip-v4.tcp.nss-routing	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.159.4.0.1.0.0
*ether2.ip-v4.tcp.snmp	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.161.4.0.1.0.0
*ether2.ip-v4.tcp.snmptrap	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.162.4.0.1.0.0
*ether2.ip-v4.tcp.cmip-man	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.163.4.0.1.0.0
*ether2.ip-v4.tcp.cmip-agent	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.164.4.0.1.0.0
*ether2.ip-v4.tcp.xns-courier	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.165.4.0.1.0.0
*ether2.ip-v4.tcp.s-net	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.166.4.0.1.0.0
*ether2.ip-v4.tcp.namp	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.167.4.0.1.0.0
*ether2.ip-v4.tcp.rsvd	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.168.4.0.1.0.0
*ether2.ip-v4.tcp.send	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.169.4.0.1.0.0
*ether2.ip-v4.tcp.print-srv	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.170.4.0.1.0.0
*ether2.ip-v4.tcp.multiplex	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.171.4.0.1.0.0
*ether2.ip-v4.tcp.cl-1	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.172.4.0.1.0.0
*ether2.ip-v4.tcp.xyplex-mux	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.173.4.0.1.0.0
*ether2.ip-v4.tcp.mailq	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.174.4.0.1.0.0
*ether2.ip-v4.tcp.vmnet	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.175.4.0.1.0.0
*ether2.ip-v4.tcp.genrad-mux	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.176.4.0.1.0.0
*ether2.ip-v4.tcp.nextstep	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.178.4.0.1.0.0
*ether2.ip-v4.tcp.bgp	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.179.4.0.1.0.0

Protocol Encapsulation	Protocol Identifier (Object ID)
*ether2.ip-v4.tcp.ris	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.180.4.0.1.0.0
*ether2.ip-v4.tcp.unify	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.181.4.0.1.0.0
*ether2.ip-v4.tcp.audit	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.182.4.0.1.0.0
*ether2.ip-v4.tcp.ocbinder	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.183.4.0.1.0.0
*ether2.ip-v4.tcp.ocserver	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.184.4.0.1.0.0
*ether2.ip-v4.tcp.remote-kis	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.185.4.0.1.0.0
*ether2.ip-v4.tcp.kis	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.186.4.0.1.0.0
*ether2.ip-v4.tcp.aci	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.187.4.0.1.0.0
*ether2.ip-v4.tcp.mumps	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.188.4.0.1.0.0
*ether2.ip-v4.tcp.qft	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.189.4.0.1.0.0
*ether2.ip-v4.tcp.gacp	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.190.4.0.1.0.0
*ether2.ip-v4.tcp.prospero	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.191.4.0.1.0.0
*ether2.ip-v4.tcp.osu-nms	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.192.4.0.1.0.0
*ether2.ip-v4.tcp.srmp	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.193.4.0.1.0.0
*ether2.ip-v4.tcp.irc	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.194.4.0.1.0.0
*ether2.ip-v4.tcp.dn6-nlm-aud	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.195.4.0.1.0.0
*ether2.ip-v4.tcp.dn6-smm-red	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.196.4.0.1.0.0
*ether2.ip-v4.tcp.dls	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.197.4.0.1.0.0
*ether2.ip-v4.tcp.dls-mon	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.198.4.0.1.0.0
*ether2.ip-v4.tcp.smux	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.199.4.0.1.0.0
*ether2.ip-v4.tcp.src	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.200.4.0.1.0.0
*ether2.ip-v4.tcp.at-rtmp	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.201.4.0.1.0.0
*ether2.ip-v4.tcp.at-nbp	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.202.4.0.1.0.0
*ether2.ip-v4.tcp.at-3	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.203.4.0.1.0.0
*ether2.ip-v4.tcp.at-echo	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.204.4.0.1.0.0
*ether2.ip-v4.tcp.at-5	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.205.4.0.1.0.0
*ether2.ip-v4.tcp.at-zis	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.206.4.0.1.0.0
*ether2.ip-v4.tcp.at-7	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.207.4.0.1.0.0
*ether2.ip-v4.tcp.at-8	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.208.4.0.1.0.0
*ether2.ip-v4.tcp.tam	16.1.0.0.1.0.0.8.0.0.0.0.6.0.0.0.209.4.0.1.0.0

Protocol Encapsulation	Protocol Identifier (Object ID)
*ether2.ip-v4.tcp.z39-50	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.210.4.0.1.0.0
*ether2.ip-v4.tcp.914c-g	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.211.4.0.1.0.0
*ether2.ip-v4.tcp.anet	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.212.4.0.1.0.0
*ether2.ip-v4.tcp.vmpwscs	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.214.4.0.1.0.0
*ether2.ip-v4.tcp.softpc	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.215.4.0.1.0.0
*ether2.ip-v4.tcp.atls	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.216.4.0.1.0.0
*ether2.ip-v4.tcp.dbase	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.217.4.0.1.0.0
*ether2.ip-v4.tcp.mpp	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.218.4.0.1.0.0
*ether2.ip-v4.tcp.uarps	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.219.4.0.1.0.0
*ether2.ip-v4.tcp.imap3	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.220.4.0.1.0.0
*ether2.ip-v4.tcp.fln-spx	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.221.4.0.1.0.0
*ether2.ip-v4.tcp.rsh-spx	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.222.4.0.1.0.0
*ether2.ip-v4.tcp.cdc	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.223.4.0.1.0.0
*ether2.ip-v4.tcp.sur-meas	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.243.4.0.1.0.0
*ether2.ip-v4.tcp.link	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.245.4.0.1.0.0
*ether2.ip-v4.tcp.dsp3270	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.0.246.4.0.1.0.0
*ether2.ip-v4.tcp.ldap	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.1.133.4.0.1.0.0
*ether2.ip-v4.tcp.https	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.1.187.4.0.1.0.0
*ether2.ip-v4.tcp.exec	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.2.0.4.0.1.0.0
*ether2.ip-v4.tcp.login	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.2.1.4.0.1.0.0
*ether2.ip-v4.tcp.cmd	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.2.2.4.0.1.0.0
*ether2.ip-v4.tcp.printer	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.2.3.4.0.1.0.0
*ether2.ip-v4.tcp.uucp	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.2.28.4.0.1.0.0
*ether2.ip-v4.tcp.banyan-vip	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.2.61.4.0.1.0.0
*ether2.ip-v4.tcp.doom	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.2.154.4.0.1.0.0
*ether2.ip-v4.tcp.notes	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.5.72.4.0.1.0.0
*ether2.ip-v4.tcp.oracl-srv	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.5.245.4.0.1.0.0
*ether2.ip-v4.tcp.oracl-tns	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.5.246.4.0.1.0.0
*ether2.ip-v4.tcp.oracl-tns-srv	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.5.247.4.0.1.0.0
*ether2.ip-v4.tcp.oracl-coauthor	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.5.249.4.0.1.0.0

Protocol Encapsulation	Protocol Identifier (Object ID)
*ether2.ip-v4.tcp.oracl-remdb	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.6.35.4.0.1.0.0
*ether2.ip-v4.tcp.oracl-names	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.6.39.4.0.1.0.0
*ether2.ip-v4.tcp.oracl-em1	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.6.212.4.0.1.0.0
*ether2.ip-v4.tcp.oracl-em2	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.6.218.4.0.1.0.0
*ether2.ip-v4.tcp.ms-streaming	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.6.219.4.0.1.0.0
*ether2.ip-v4.tcp.oracl-vp2	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.7.16.4.0.1.0.0
*ether2.ip-v4.tcp.oracl-vp1	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.7.17.4.0.1.0.0
*ether2.ip-v4.tcp.ccmail	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.12.192.4.0.1.0.0
*ether2.ip-v4.tcp.xwin	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.23.112.4.0.1.0.0
*ether2.ip-v4.tcp.quake	16.1.0.0.1.0.0.8.0.0.0.0.0.6.0.0.101.144.4.0.1.0.0
UDP Applications	
*ether2.ip-v4.udp.echo	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.7.4.0.1.0.0
*ether2.ip-v4.udp.discard	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.9.4.0.1.0.0
*ether2.ip-v4.udp.systat	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.11.4.0.1.0.0
*ether2.ip-v4.udp.daytime	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.13.4.0.1.0.0
*ether2.ip-v4.udp.qotd	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.17.4.0.1.0.0
*ether2.ip-v4.udp.msp	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.18.4.0.1.0.0
*ether2.ip-v4.udp.chargen	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.19.4.0.1.0.0
*ether2.ip-v4.udp.priv-mail	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.24.4.0.1.0.0
*ether2.ip-v4.udp.nsw-fe	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.27.4.0.1.0.0
*ether2.ip-v4.udp.msg-icp	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.29.4.0.1.0.0
*ether2.ip-v4.udp.msg-auth	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.31.4.0.1.0.0
*ether2.ip-v4.udp.dsp	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.33.4.0.1.0.0
*ether2.ip-v4.udp.priv-print	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.35.4.0.1.0.0
*ether2.ip-v4.udp.time	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.37.4.0.1.0.0
*ether2.ip-v4.udp.rlp	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.39.4.0.1.0.0
*ether2.ip-v4.udp.graphics	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.41.4.0.1.0.0
*ether2.ip-v4.udp.nameserver	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.42.4.0.1.0.0
*ether2.ip-v4.udp.auditd	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.48.4.0.1.0.0
*ether2.ip-v4.udp.re-mail-ck	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.50.4.0.1.0.0

Protocol Encapsulation	Protocol Identifier (Object ID)
*ether2.ip-v4.udp.la-maint	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.51.4.0.1.0.0
*ether2.ip-v4.udp.xns-time	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.52.4.0.1.0.0
*ether2.ip-v4.udp.domain	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.53.4.0.1.0.0
*ether2.ip-v4.udp.xns-ch	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.54.4.0.1.0.0
*ether2.ip-v4.udp.isi-gl	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.55.4.0.1.0.0
*ether2.ip-v4.udp.xns-auth	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.56.4.0.1.0.0
*ether2.ip-v4.udp.priv-term	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.57.4.0.1.0.0
*ether2.ip-v4.udp.xns-mail	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.58.4.0.1.0.0
*ether2.ip-v4.udp.priv-file	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.59.4.0.1.0.0
*ether2.ip-v4.udp.ni-mail	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.61.4.0.1.0.0
*ether2.ip-v4.udp.bootps	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.67.4.0.1.0.0
*ether2.ip-v4.udp.bootpc	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.68.4.0.1.0.0
*ether2.ip-v4.udp.tftp	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.69.4.0.1.0.0
*ether2.ip-v4.udp.priv-dialout	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.75.4.0.1.0.0
*ether2.ip-v4.udp.deos	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.76.4.0.1.0.0
*ether2.ip-v4.udp.priv-rje	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.77.4.0.1.0.0
*ether2.ip-v4.udp.vettcp	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.78.4.0.1.0.0
*ether2.ip-v4.udp.hosts2-ns	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.81.4.0.1.0.0
*ether2.ip-v4.udp.xfer	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.82.4.0.1.0.0
*ether2.ip-v4.udp.mit-ml-dev	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.83.4.0.1.0.0
*ether2.ip-v4.udp.ctf	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.84.4.0.1.0.0
*ether2.ip-v4.udp.mit-ml-dev	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.85.4.0.1.0.0
*ether2.ip-v4.udp.kerberos	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.88.4.0.1.0.0
*ether2.ip-v4.udp.npp	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.92.4.0.1.0.0
*ether2.ip-v4.udp.dcp	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.93.4.0.1.0.0
*ether2.ip-v4.udp.dixie	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.96.4.0.1.0.0
*ether2.ip-v4.udp.swift-rvf	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.97.4.0.1.0.0
*ether2.ip-v4.udp.tacnews	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.98.4.0.1.0.0
*ether2.ip-v4.udp.metagram	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.99.4.0.1.0.0
*ether2.ip-v4.udp.iso-tsap	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.102.4.0.1.0.0

Protocol Encapsulation	Protocol Identifier (Object ID)
*ether2.ip-v4.udp.gppitnp	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.103.4.0.1.0.0
*ether2.ip-v4.udp.csnet-ns	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.105.4.0.1.0.0
*ether2.ip-v4.udp.3com-tsmux	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.106.4.0.1.0.0
*ether2.ip-v4.udp.pop3	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.110.4.0.1.0.0
*ether2.ip-v4.udp.sunrpc	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.111.4.0.1.0.0
*ether2.ip-v4.udp.audionews	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.114.4.0.1.0.0
*ether2.ip-v4.udp.ansanotify	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.116.4.0.1.0.0
*ether2.ip-v4.udp.sqlserv	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.118.4.0.1.0.0
*ether2.ip-v4.udp.cfdptkt	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.120.4.0.1.0.0
*ether2.ip-v4.udp.erpc	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.121.4.0.1.0.0
*ether2.ip-v4.udp.smakynet	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.122.4.0.1.0.0
*ether2.ip-v4.udp.ntp	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.123.4.0.1.0.0
*ether2.ip-v4.udp.ansatrader	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.124.4.0.1.0.0
*ether2.ip-v4.udp.unitary	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.126.4.0.1.0.0
*ether2.ip-v4.udp.gss-xlicen	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.128.4.0.1.0.0
*ether2.ip-v4.udp.pwdgen	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.129.4.0.1.0.0
*ether2.ip-v4.udp.cisco-fna	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.130.4.0.1.0.0
*ether2.ip-v4.udp.cisco-tna	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.131.4.0.1.0.0
*ether2.ip-v4.udp.cisco-sys	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.132.4.0.1.0.0
*ether2.ip-v4.udp.statsrv	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.133.4.0.1.0.0
*ether2.ip-v4.udp.loc-srv	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.135.4.0.1.0.0
*ether2.ip-v4.udp.netbios-ns	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.137.4.0.1.0.0
*ether2.ip-v4.udp.netbios-dgm	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.138.4.0.1.0.0
*ether2.ip-v4.udp.netbios-ssn	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.139.4.0.1.0.0
*ether2.ip-v4.udp.emfis-data	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.140.4.0.1.0.0
*ether2.ip-v4.udp.emfis-cntl	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.141.4.0.1.0.0
*ether2.ip-v4.udp.bl-idm	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.142.4.0.1.0.0
*ether2.ip-v4.udp.news	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.144.4.0.1.0.0
*ether2.ip-v4.udp.uaac	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.145.4.0.1.0.0
*ether2.ip-v4.udp.iso-tp0	16.1.0.0.1.0.0.8.0.0.0.0.17.0.0.0.146.4.0.1.0.0

Protocol Encapsulation	Protocol Identifier (Object ID)
*ether2.ip-v4.udp.iso-ip	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.147.4.0.1.0.0
*ether2.ip-v4.udp.cronus	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.148.4.0.1.0.0
*ether2.ip-v4.udp.aed-512	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.149.4.0.1.0.0
*ether2.ip-v4.udp.sql-net	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.150.4.0.1.0.0
*ether2.ip-v4.udp.sgmp	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.153.4.0.1.0.0
*ether2.ip-v4.udp.netsc-prod	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.154.4.0.1.0.0
*ether2.ip-v4.udp.netsc-dev	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.155.4.0.1.0.0
*ether2.ip-v4.udp.nss-routing	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.159.4.0.1.0.0
*ether2.ip-v4.udp.sgmp-traps	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.160.4.0.1.0.0
*ether2.ip-v4.udp.snmp	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.161.4.0.1.0.0
*ether2.ip-v4.udp.snmptrap	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.162.4.0.1.0.0
*ether2.ip-v4.udp.cmip-man	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.163.4.0.1.0.0
*ether2.ip-v4.udp.cmip-agent	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.164.4.0.1.0.0
*ether2.ip-v4.udp.xns-courier	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.165.4.0.1.0.0
*ether2.ip-v4.udp.s-net	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.166.4.0.1.0.0
*ether2.ip-v4.udp.namp	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.167.4.0.1.0.0
*ether2.ip-v4.udp.rsvd	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.168.4.0.1.0.0
*ether2.ip-v4.udp.send	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.169.4.0.1.0.0
*ether2.ip-v4.udp.print-srv	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.170.4.0.1.0.0
*ether2.ip-v4.udp.multiplex	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.171.4.0.1.0.0
*ether2.ip-v4.udp.cl-1	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.172.4.0.1.0.0
*ether2.ip-v4.udp.xyplex-mux	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.173.4.0.1.0.0
*ether2.ip-v4.udp.mailq	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.174.4.0.1.0.0
*ether2.ip-v4.udp.vmnet	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.175.4.0.1.0.0
*ether2.ip-v4.udp.genrad-mux	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.176.4.0.1.0.0
*ether2.ip-v4.udp.xdmcp	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.177.4.0.1.0.0
*ether2.ip-v4.udp.nextstep	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.178.4.0.1.0.0
*ether2.ip-v4.udp.ris	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.180.4.0.1.0.0
*ether2.ip-v4.udp.unify	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.181.4.0.1.0.0
*ether2.ip-v4.udp.audit	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.182.4.0.1.0.0

Protocol Encapsulation	Protocol Identifier (Object ID)
*ether2.ip-v4.udp.ocbinder	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.183.4.0.1.0.0
*ether2.ip-v4.udp.ocserver	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.184.4.0.1.0.0
*ether2.ip-v4.udp.remote-kis	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.185.4.0.1.0.0
*ether2.ip-v4.udp.kis	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.186.4.0.1.0.0
*ether2.ip-v4.udp.aci	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.187.4.0.1.0.0
*ether2.ip-v4.udp.mumps	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.188.4.0.1.0.0
*ether2.ip-v4.udp.osu-nms	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.192.4.0.1.0.0
*ether2.ip-v4.udp.srmp	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.193.4.0.1.0.0
*ether2.ip-v4.udp.irc	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.194.4.0.1.0.0
*ether2.ip-v4.udp.dls	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.197.4.0.1.0.0
*ether2.ip-v4.udp.dls-mon	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.198.4.0.1.0.0
*ether2.ip-v4.udp.src	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.200.4.0.1.0.0
*ether2.ip-v4.udp.at-rtmp	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.201.4.0.1.0.0
*ether2.ip-v4.udp.at-nbp	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.202.4.0.1.0.0
*ether2.ip-v4.udp.at-3	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.203.4.0.1.0.0
*ether2.ip-v4.udp.at-echo	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.204.4.0.1.0.0
*ether2.ip-v4.udp.at-5	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.205.4.0.1.0.0
*ether2.ip-v4.udp.at-zis	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.206.4.0.1.0.0
*ether2.ip-v4.udp.at-7	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.207.4.0.1.0.0
*ether2.ip-v4.udp.at-8	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.208.4.0.1.0.0
*ether2.ip-v4.udp.tam	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.209.4.0.1.0.0
*ether2.ip-v4.udp.914c-g	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.211.4.0.1.0.0
*ether2.ip-v4.udp.anet	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.212.4.0.1.0.0
*ether2.ip-v4.udp.ipx-tunnel	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.213.4.0.1.0.0
*ether2.ip-v4.udp.vmpwscs	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.214.4.0.1.0.0
*ether2.ip-v4.udp.softpc	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.215.4.0.1.0.0
*ether2.ip-v4.udp.atls	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.216.4.0.1.0.0
*ether2.ip-v4.udp.dbase	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.217.4.0.1.0.0
*ether2.ip-v4.udp.uarps	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.219.4.0.1.0.0
*ether2.ip-v4.udp.fln-spx	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.221.4.0.1.0.0

Protocol Encapsulation	Protocol Identifier (Object ID)
*ether2.ip-v4.udp.rsh-spx	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.222.4.0.1.0.0
*ether2.ip-v4.udp.cdc	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.223.4.0.1.0.0
*ether2.ip-v4.udp.sur-meas	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.243.4.0.1.0.0
*ether2.ip-v4.udp.link	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.245.4.0.1.0.0
*ether2.ip-v4.udp.dsp3270	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.0.246.4.0.1.0.0
*ether2.ip-v4.udp.ldap	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.1.133.4.0.1.0.0
*ether2.ip-v4.udp.biff	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.2.0.4.0.1.0.0
*ether2.ip-v4.udp.who	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.2.1.4.0.1.0.0
*ether2.ip-v4.udp.syslog	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.2.2.4.0.1.0.0
*ether2.ip-v4.udp.ip-xns-rip	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.2.8.4.0.1.0.0
*ether2.ip-v4.udp.banyan-vip	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.2.61.4.0.1.0.0
*ether2.ip-v4.udp.notes	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.5.72.4.0.1.0.0
*ether2.ip-v4.udp.ccmail	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.12.192.4.0.1.0.0
*ether2.ip-v4.udp.quake	16.1.0.0.1.0.0.8.0.0.0.0.0.17.0.0.101.144.4.0.1.0.0