Lucent Technologies
Bell Labs Innovations

Stinger®

IP2000 Configuration Guide

Part Number: 7820-0976-003
For software version 9.5-206
August 2003

Notice

Every effort was made to ensure that the information in this document was complete and accurate at the time of printing, but information is subject to change.

European Community (EC) RTTE compliance

CE Hereby, Lucent Technologies, declares that the equipment documented in this publication is in compliance with the essential requirements and other relevant provisions of the Radio and Telecommunications Technical Equipment (RTTE) Directive 1999/5/EC.

To view the official *Declaration of Conformity* certificate for this equipment, according to EN 45014, access the Lucent INS online documentation library at http://www.lucentdocs.com/ins.

Safety, compliance, and warranty Information

Before handling any Lucent Access Networks hardware product, read the *Edge Access and Broadband Access Safety and Compliance Guide* included in your product package. See that guide also to determine how products comply with the electromagnetic interference (EMI) and network compatibility requirements of your country. See the warranty card included in your product package for the limited warranty that Lucent Technologies provides for its products.

Security statement

In rare instances, unauthorized individuals make connections to the telecommunications network through the use of access features.

Trademarks

Lucent, the Lucent logo, and all Lucent brand and product names are trademarks or registered trademarks of Lucent Technologies Inc. Other brand and product names are trademarks of their respective holders.

Ordering Information

You can order the most up-to-date product information and computer-based training online at http://www.lucentdocs.com/bookstore.

Feedback

Lucent Technologies appreciates customer comments about this manual. Please send them to techcomm@lucent.com.

Lucent Technologies

Customer Service

Product and service information, and software upgrades, are available 24 hours a day. Technical assistance options accommodate varying levels of urgency.

Finding information and software

To obtain software upgrades, release notes, and addenda for this product, log in to Lucent OnLine Customer Support at `http://www.lucent.com/support`.

Lucent OnLine Customer Support also provides technical information, product information, and descriptions of available services. The center is open 24 hours a day, seven days a week. Log in and select a service.

Obtaining technical assistance

Lucent OnLine Customer Support at `http://www.lucent.com/support` provides access to technical support. You can obtain technical assistance through email or the Internet, or by telephone. If you need assistance, make sure that you have the following information available:

- Active service or maintenance contract number, entitlement ID, or site ID
- Product name, model, and serial number
- Software version
- Software and hardware options If supplied by your carrier, service profile identifiers (SPIDs) associated with your line
- Your local telephone company's switch type and operating mode, such as AT&T, 5ESS Custom, or Northern Telecom National ISDN-1
- Whether you are routing or bridging with your Lucent product
- Type of computer you are using
- Description of the problem

Obtaining assistance through email or the Internet

If your services agreement allows, you can communicate directly with a technical engineer through Email Technical Support or a Live Chat. Select one of these sites when you log in to `http://www.lucent.com/support`.

Calling the technical assistance center (TAC)

If you cannot find an answer through the tools and information of Lucent OnLine Customer Support or if you have a very urgent need, contact TAC. Access Lucent OnLine Customer Support at `http://www.lucent.com/support` and click `Contact Us` for a list of telephone numbers inside and outside the United States.

Alternatively, call 1-866-LUCENT8 (1-866-582-3688) from any location in North America for a menu of Lucent services. Or call +1 510-747-2000 for an operator. You must have an active services agreement or contract.

Contents

Figures

Tables

About This Guide

A Stinger unit with the IP2000 control module (a Stinger IP2000) supports identical Asynchronous Transfer Mode (ATM) capabilities to those in Stinger units with the standard control module. In addition, a Stinger IP2000 can terminate IP traffic and forward it across a built-in Gigabit Ethernet interface.

Note Instructions for installing and configuring the management functions of the IP2000 are found in the *Getting Started Guide* for your Stinger platform.

What is in this guide

This guide focuses on the aspects of Stinger configuration that are specific to IP2000 control module capabilities. To fully configure the system for both ATM and IP capabilities, use this guide with the *Stinger ATM Configuration Guide*.

Note You can configure the amount of bandwidth allocated to LIM interfaces and control modules for carrying upstream traffic. For details about that aspect of using the IP2000 control module, as well as for ATM quality of service (QoS) and other traffic management capabilities, see the *Stinger ATM Configuration Guide*.

This guide describes how to configure IP routing and related functions in the Stinger Stinger. It includes information about local and global network IP issues, as well as how to configure both IP-routed switch-through ATM permanent virtual circuits (PVCs) and RFC 1483 PVCs.

This guide also describes how to set up IEEE 802.1Q virtual local area network (VLAN) support on the Gigabit Ethernet interface, and how to configure the system to support multicast video over DSL with Internet Group Management Protocol (IGMP) version-1 or version-2 messaging.

Warning Before installing your Stinger unit, be sure to read the safety instructions in the *Edge Access and Broadband Access Safety and Compliance Guide*. For information specific to your unit, see the "Safety-Related Physical, Environmental, and Electrical Information" appendix in the *Getting Started Guide* for your Stinger unit.

Documentation conventions

Following are all the special characters and typographical conventions used in this manual:

Convention	Meaning
`Monospace text`	Represents text that appears on your computer's screen, or that could appear on your computer's screen.
`Boldface monospace text`	Represents characters that you enter exactly as shown (unless the characters are also in ***`italics`***—see *Italics*, below). If you could enter the characters but are not specifically instructed to, they do not appear in boldface.
Italics	Represent variable information. Do not enter the words themselves in the command. Enter the information they represent. In ordinary text, italics are used for titles of publications, for some terms that would otherwise be in quotation marks, and to show emphasis.
[ ]	Square brackets indicate an optional argument you might add to a command. To include such an argument, type only the information inside the brackets. Do not type the brackets unless they appear in boldface.
|	Separates command choices that are mutually exclusive.
>	Points to the next level in the path to a parameter or menu item. The item that follows the angle bracket is one of the options that appear when you select the item that precedes the angle bracket.
Key1+Key2	Represents a combination keystroke. To enter a combination keystroke, press the first key and hold it down while you press one or more other keys. Release all the keys at the same time. (For example, Ctrl+H means hold down the Ctrl key and press the H key.)
Press Enter	Means press the Enter or Return key or its equivalent on your computer.
 **Note**	Introduces important additional information.
 **Caution**	Warns that a failure to follow the recommended procedure could result in loss of data or damage to equipment.
 **Warning**	Warns that a failure to take appropriate safety precautions could result in physical injury.
 **Warning**	Warns of danger of electric shock.

Stinger documentation set

The Stinger documentation set consists of the following manuals, which can be found at `http://www.lucent.com/support` and `http://www.lucentdocs.com/ins`.

- **Read me first:**
  - *Edge Access and Broadband Access Safety and Compliance Guide*. Contains important safety instructions and country-specific information that you must read before installing a Stinger unit.
  - *TAOS Command-Line Interface Guide*. Introduces the TAOS command-line environment and shows you how to use the command-line interface effectively. This guide describes keyboard shortcuts and introduces commands, security levels, profile structure, and parameter types.
- **Installation and basic configuration:**
  - *Getting Started Guide* for your Stinger platform. Shows how to install your Stinger chassis and hardware. This guide also shows you how to use the command-line interface to configure and verify IP access and basic access security on the unit, and how to configure Stinger control module redundancy on units that support it.
  - Module guides. For each Stinger line interface module (LIM), trunk module, or other type of module, an individual guide describes the module's features and provides instructions for configuring the module and verifying its status.
- **Configuration:**
  - *Stinger ATM Configuration Guide*. Describes how to integrate the Stinger into the ATM and Digital Subscriber Line (DSL) access infrastructure. The guide explains how to configure PVCs, and shows how to use standard ATM features such as quality of service (QoS), connection admission control (CAC), and subtending.
  - *Stinger IP2000 Configuration Guide*. For Stinger IP2000 systems, this guide describes how to integrate the system into the IP infrastructure. Topics include IP-routed switch-through ATM PVCs and RFC 1483 PVCs, IEEE 802.1Q VLAN, and forwarding multicast video transmissions on DSL interfaces.
  - *Stinger Private Network-to-Network Interface (PNNI) Supplement*. For the optional PNNI software, this guide provides quick-start instructions for configuring PNNI and soft PVCs (SPVCs), and describes the related profiles and commands.
  - *Stinger SNMP Management of the ATM Stack Supplement.* Describes SNMP management of ATM ports, interfaces, and connections on a Stinger unit to provide guidelines for configuring and managing ATM circuits through any SNMP management utility.
  - *Stinger T1000 Module Routing and Tunneling Supplement.* For the optional T1000 module, this guide describes how to configure the Layer 3 routing and virtual private network (VPN) capabilities.
- **RADIUS:** *TAOS RADIUS Guide and Reference*. Describes how to set up a unit to use the Remote Authentication Dial-In User Service (RADIUS) server and contains a complete reference to RADIUS attributes.

- **Administration and troubleshooting:** *Stinger Administration Guide*. Describes how to administer the Stinger unit and manage its operations. Each chapter focuses on a particular aspect of Stinger administration and operations. The chapters describe tools for system management, network management, and Simple Network Management Protocol (SNMP) management.
- **Reference**:
  - *Stinger Reference*. An alphabetic reference to Stinger profiles, parameters, and commands.
  - *TAOS Glossary*. Defines terms used in documentation for Stinger units.

Related documents

The following industry documents provide background information about features described in this guide:

- RFC 951, *Bootstrap Protocol*
- RFC 1112, *Host Extensions for IP Multicasting*
- RFC 1334, *PPP Authentication Protocols*
- RFC 1483, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*
- RFC 1587, *The OSPF NSSA Option.*
- RFC 1700, *Assigned Numbers*
- RFC 1723, *RIP Version 2: Carrying Additional Information*
- RFC 1812, *Requirements for IP Version 4 Routers*
- RFC 1994, *PPP Challenge Handshake Authentication Protocol (CHAP)*
- RFC 2131, *Dynamic Host Configuration Protocol*
- RFC 2132, *DHCP Options and BOOTP Vendor Extensions*
- RFC 2236, *Internet Group Management Protocol Version 2*
- RFC 2328, *OSPF Version 2*
- RFC 2362, *Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification,* `draft-ietf-pim-sm-v2-new-07.txt`, March 2003, `draft-ietf-pim-sm-bsr-03.txt`, February 2003
- RFC 2364, *PPP over AAL5*
- RFC 2516, *A Method for Transmitting PPP Over Ethernet (PPPoE)*
- RFC 3046, *DHCP Relay Agent Information Option*
- IEEE 802.1Q-1998, *IEEE Standard for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks*

Welcome to the IP2000

1

Stinger platforms and model numbers

The IP2000 control module supports a fiber-based Gigabit Ethernet (GigE) interface, with a modular Small Form Factor Pluggable (SFP) transceiver. The IP2000 is supported on he Stinger FS, the Stinger FS+, Stinger LS, and Stinger RT platforms. Table 1-1 shows IP2000 model numbers and platform support:

Table 1-1. IP2000 model numbers and platform support

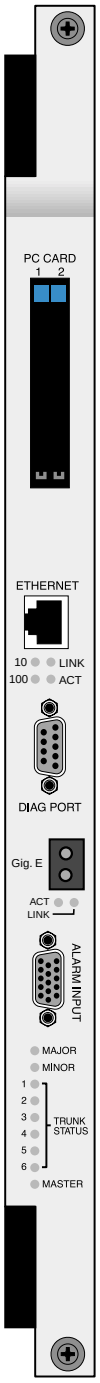
IP2000 model number	Description	Supporting platforms
STGR-CM-IP2000-F	IP2000 with fiber gigabit Ethernet	Stinger FS Stinger FS+ Stinger LS
STGRRT-CM-IP2000-F	IP2000 with fiber gigabit Ethernet, environmentally hardened	Stinger RT
STGR-SFP-SX	Short-haul gigabit Ethernet SFP module	Stinger FS Stinger FS+ Stinger LS
STGR-SFP-LX	Long-haul gigabit Ethernet SFP module	Stinger FS Stinger FS+ Stinger LS
STGRRT-SFP-LX	Long-haul gigabit Ethernet SFP module, environmentally hardened	Stinger RT

IP2000 software specifications

The IP2000 control module supports the following connection features:

Software capability	Specifications
ATM protocols	ATM Forum UNI (v3.0 and v3.1)
	ATM Forum Interim Interswitch Signaling Protocol (IISP)
	ATM Forum Traffic Management v4.0
	ATM Forum PNNI 1.0 (optional)
IP routing protocols	RIPv1, RIPv2, OSPF
IP multicast	Internet Group Management Protocol (IGMP) v1, v2, Protocol Independent Multicast Sparse Mode (PIM-SM v2)
IP/ATM	Multiprotocol Encapsulation over ATM Adaptation Layer 5 (RFC 1483), bridged IP routing (BIR)
Broadband RAS	Broadband remote access server (BRAS) for PPP sessions over DSL interfaces
IP/Ethernet	IP support for Gigabit Ethernet interface
VLAN	IEEE 802.1Q tagged VLANs
ATM QoS and IP CoS	ATM traffic is assigned the highest priority and passed through. IP traffic is assigned a strict priority based on service classification and placed in a CoS queue: ■ Level 1: Protocol control messages ■ Level 2: IP multicast traffic ■ Level 3: IP unicast traffic
	IP traffic shaping in downstream direction (toward CPE) on per-VC basis
	Traffic management for CoS and ATM queues
Security	RADIUS, Extended RADIUS
	Password Authentication Protocol (PAP)
	Challenge Authentication Protocol (CHAP)
	Profile-based access

IP2000 hardware specifications



Category	Specifications	
Physical dimensions	Height:	15 inches (38.1cm)
	Width:	1.06 inches (2.69cm)
	Depth:	9 inches (22.8cm)
Weight	3.4 pounds (1.5kg)	
Operating Requirements	Power:	70 Watts Maximum
	Temperature:	FS/LS version: 32°F–131°F (0°C–55°C)
		RT version: 40°F–149°F (-40°C–65°C)
	Relative humidity:	10% through 95% (noncondensing)
	Operating altitude:	Up to 13,123 feet (4,000m)
LAN interface (fiber)	Modular Small Form Factor Pluggable (SFP) transceiver with duplex LC connector	
	IEEE 802.3z 1000BASE-SX (short haul) over multi-mode fiber, distance support to 550m	
	IEEE 802.3z 1000BASE-LX (long haul) over single mode fiber, distance support to 10km	
Mgmt interfaces	10/100 BASE-T Ethernet, RS-232 serial port	
Status indicators	10/100 BaseT	10 (Green): 10 Mbps speed
		100 (Green): 100 Mbps speed
		LINK (Green): Operational link
		ACT (Green): Traffic activity
	Gigabit Ethernet	LINK (Green): Operational link
		ACT (Green): Traffic activity
	Others	MAJOR (Red): Major alarm detected
		MINOR (Red): Minor alarm detected
		TRUNK STATUS 1-6 (Amber): Trunk port status
		MASTER (Green): Module is master controller
Electromagnetic compliance	FCC Part 15 Class A, EN55022 Class A, AS/NZS 3548 Class A, VCCI Class A, CISPR 22 Class A, EN 300386-2	
Certification	Bellcore GR-63-CORE (NEBS Level 1-3), Bellcore-GR-1089-CORE, EN / IEC 60950	
Expansion slot	One PC card slot for configuration or upgrade storage	
Switching fabric	64x64 nonblocking ATM crosspoint switch	
	1.6Gbps ATM switching capacity	
	2.4Gpbs IP switching/routing capacity	

Network architecture overview

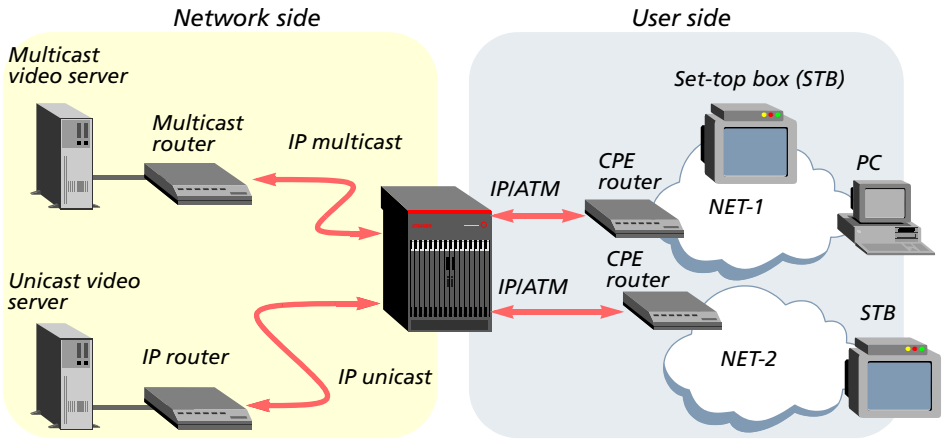
A Stinger IP2000 supports all standard Stinger ATM features, such as data and voice services over DSL. In addition, it supports IP services such as multicast video, unicast video-on-demand, and other video and IPTV applications. The services supported by the IP2000 are provided downstream to DSL subscribers. In the upstream direction, DSL subscribers accessing the Stinger IP2000 via IP over ATM can be directly routed via the IP infrastructure to Internet services.

A Stinger IP2000 supports IP routing, ATM quality of service and traffic shaping, virtual LAN, and multicasting capabilities to provide fast, efficient access to ATM and IP services.

Multicast video

A Stinger IP2000 uses the Internet Group Management Protocol (IGMP) to manage group memberships of downstream video to a PC application or set-top box, as shown in Figure 1-1. Administrators can configure levels of service that control subscribers' access to specific multicast groups. Connection to originating router can be across the Gigabit Ethernet interface or through a high-speed IP over ATM connection.

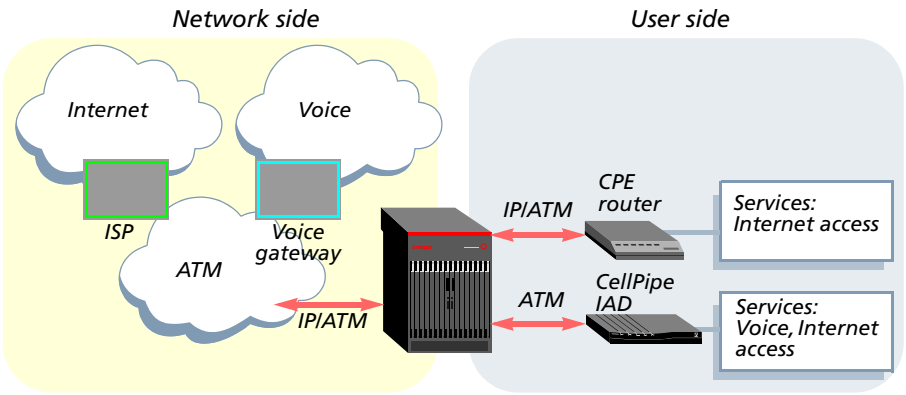
Figure 1-1. Sample setup showing multicast and unicast video services



Internet and voice access

When a subscriber has DSL Integrated Access Device (IAD) equipment (such as a CellPipe®), the Stinger can deliver integrated voice and data services over the local copper loop, providing a efficient, low-cost solution for enterprise, small business, home office, and residential subscribers.

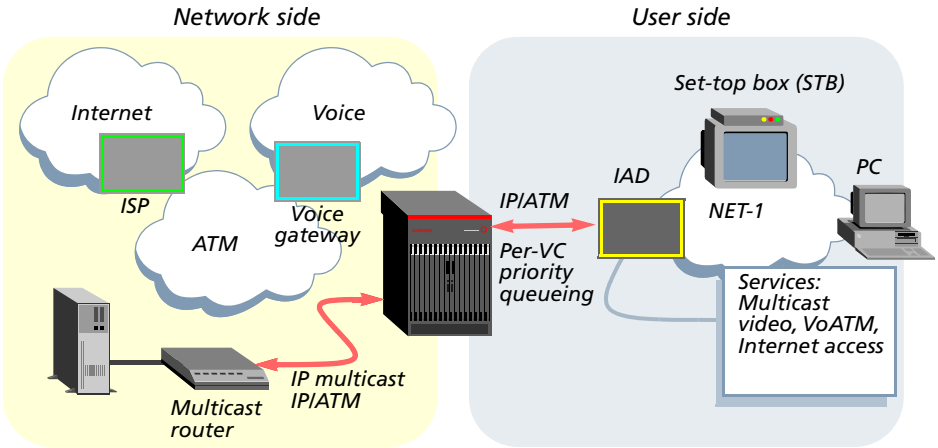
Figure 1-2. Sample setup showing Internet access and voice over ATM



Multiplexing multiple IP flows on a single ATM VCC

A Stinger IP2000 supports an implementation of Class of Service (CoS) that co-exists with the Stinger ATM QoS implementation. This feature allows transferring multiple IP video streams (multicast and unicast) over single user-side ATM virtual circuit.

Figure 1-3. Sample setup showing multiple IP flows to a CPE router



The proprietary CoS implementation enables the delivery of differentiated services over an IP infrastructure. All traffic handled by the IP2000, whether encapsulated IP or native ATM, passes through the network processor function.

Non-IP terminated ATM traffic, including operations, administration, and maintenance (OAM) F5 traffic, is treated as highest priority and handled in an *ATM pass-through mode*. This traffic passes through the network processor with no further processing.

RFC 1483 IP traffic that terminates on the IP2000 is reassembled from ATM cells into IP packets. It is then classified and assigned to priority output queues. Per-VC strict-priority queuing is supported with three priority levels as described in Table 1-2.

Table 1-2. CoS and per-VC queueing for prioritizing IP packet processing

Priority queue	Priority level	Packet classification assigned to queue
1	High	IP Control Protocol Classification
		■ ARP/RARP protocol messages
		■ ICMP protocol messages
		■ RIP protocol messages
		■ IGMP protocol messages
2	Medium	Multicast Classification
		■ IP multicast data
3	Low	Unicast Classification
		■ IP unicast data

Per-VC queuing operates in conjunction with the associated ATM shaping rate. The aggregate rate of the combination of three priority queues (Class of Service Queuing with Strict Priority) associated with a particular ATM virtual circuit is controlled by the SCR (sustained cell rate) configured for the VC. In this case, SCR is configured equal to PCR (peak cell rate). Rate information is configurable in the `atm-qos` profile for each virtual circuit. For details about configuring ATM QoS, see the *Stinger ATM Configuration Guide*.

Gigabit Ethernet Configuration 2

The IP2000 controller has two Ethernet interfaces, one 10/100 BASE-T interface for management access to the unit via Telnet or SNMP, and one Gigabit Ethernet interface for high-speed access to a local IP subnet. For information about configuring the management interface, see the *Getting Started Guide* for your Stinger platform.

The Gigabit Ethernet MAC (GMAC) physical interface operates only in full-duplex mode only for a full 1Gbps throughput. It supports auto-negotiation for advertising its rate and duplex mode, but not for renegotiating it on the IEEE 802 LAN.

Stinger units with redundant IP2000 controllers can be configured to enable Gigabit Ethernet port redundancy. With proper configuration, RFC 1483 (MPoA) connections, VLAN connections, and MBONE interface functions can be maintained across Gigabit Ethernet following a controller switchover.

Configuring the physical and logical interface

The system creates configuration profiles for both IP2000 Ethernet interfaces. For each controller, interface 1 is always the 10/100 BASE-T management interface, and interface 2 is the Gigabit Ethernet interface. For example:

```
admin> dir ethernet
    18  07/11/2003 13:55:31  { shelf-1 first-control-module 1 }
    24  07/11/2003 19:34:41  { shelf-1 first-control-module 2 }
    18  07/11/2003 13:55:31  { shelf-1 second-control-module 1 }
    24  07/11/2003 13:57:32  { shelf-1 second-control-module 2 }
admin> dir ip-interface
    21  07/11/2003 13:55:31  { { any-shelf any-slot 0 } 0 }
    31  07/11/2003 22:46:34  { { shelf-1 first-control-module 1 } 0 }
    21  07/11/2003 13:57:01  { { shelf-1 first-control-module 2 } 0 }
    21  07/11/2003 13:55:31  { { shelf-1 second-control-module 1 } 0 }
    21  07/11/2003 13:57:01  { { shelf-1 second-control-module 2 } 0 }
```

Viewing ethernet profile settings

To configure the data-link functions of the Gigabit Ethernet interface, open the `ethernet` profile. For example:

```
admin> read ethernet { 1 8 2 }
ETHERNET/{ shelf-1 first-control-module 2 } read

admin> list
[in ETHERNET/{ shelf-1 first-control-module 2 }]
interface-address* = { shelf-1 first-control-module 2 }
link-state-enabled = no
enabled = yes
ether-if-type = fiber
bridging-enabled = no
filter-name = ""
duplex-mode = full-duplex
pppoe-options = { no no }
bridging-options = { 0 no no }
media-speed-mbit = 1000mb
auto-negotiate = no
```

For details about each of the parameters shown above, see the *Stinger Reference*. Following are some Gigabit Ethernet-specific notes about the profile contents:

Parameter	Notes about Gigabit Ethernet settings
`interface-address*`	The profile index and `interface-address` value of the profile for a Gigabit Ethernet interface always specifies an interface number of 2. For example: `shelf-1 first-control-module 2`
`link-state-enabled`	With the default value, the system discards packets and does not choose an alternate route if the interface is down. If you set this to `yes`, the system deletes routes to the interface when the interface is unavailable, and then restores the routes when the interface becomes available again.
`enabled`	If you set this to `no` and write the profile, the interface is unavailable.
`ether-if-type`	This setting is read-only and specifies either `fiber` or `utp` (CAT5 unshielded twisted pair).
`bridging-enabled`	Enable/disable LAN packet bridging on the interface. Set this parameter to `yes` to enable bridging on the Gigabit Ethernet port. This is required for VLAN operations. See Chapter 3, "VLAN Configuration."
`filter-name`	Applies a data filter to the interface. See Chapter 10, "Using IP Filters."
`duplex-mode`	This setting is read-only and specifies full-duplex mode.
`pppoe-options`	*Not used by the IP2000.*

Parameter	Notes about Gigabit Ethernet settings
`bridging-options`	*Not used by the IP2000.* If you are configuring VLANs, you set bridging options in the `vlan-ethernet` profile (not in the `ethernet` profile). For details, see Chapter 3, "VLAN Configuration."
`media-speed-mbit`	This setting is read-only and specifies 1Gbps.
`auto-negotiate`	Setting this parameter to `yes` does not cause the IP2000 to negotiate its duplex mode or speed, but it does cause the system to advertise a full-duplex 1Gbps port , which helps to ensure compatibility with remote Gigabit Ethernet interfaces that support autonegotiation.

Modifying default `ethernet` settings

With the default settings, the Gigabit Ethernet interface is fully operational. The following commands enable autonegotiation, to help ensure compatibility with other Gigabit Ethernet interfaces that can negotiate between full-duplex and half-duplex operations. (The IP2000 Gigabit Ethernet always operates in full-duplex mode.)

```
admin> read ethernet { 1 8 2 }
ETHERNET/{ shelf-1 first-control-module 2 } read

admin> set auto-negotiate = yes

admin> write -f
ETHERNET/{ shelf-1 first-control-module 2 } written
```

Assigning an IP address in the `ip-interface` profile

For details about the `ip-interface` profile, and about enabling dynamic routing or configuring static routes to enable the system to communicate beyond its own subnet, see "Configuring ip-interface profiles for Ethernet ports" on page 4-6. The following commands provide the minimal configuration of an IP address for the Gigabit Ethernet interface:

```
admin> read ip-interface { { 1 8 2 } 0 }
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } read

admin> set ip-address = 10.99.99.101/24

admin> write
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } written
```

Verifying the Gigabit Ethernet interface setup

After you assign an IP address, you can verify that the Gigabit Ethernet interface is able to handle IP traffic by checking some command output. For details about the `netstat` and `gmac` commands, see the *Stinger Reference*.

You can also use the debug-level `ifmgr -d` command to verify that the Gigabit Ethernet interface is active. This is described in Appendix A, "IP2000 Diagnostics."

Checking the routing table

The following command output verifies that the routing table has an entry for the Gigabit Ethernet interface (IP address 100.1.1.3/32):

```
admin> netstat -rn
Destination       Gateway        IF         Flg   Pref Met     Use        Age
0.0.0.0/0         1.1.2.1       ie0        SGP     60   1    3817        828
20.1.2.0/24       -              ie1-1      C        0   0       0        828
20.1.2.3/32       -              local      CP       0   0       0        828
100.1.1.0/8       -              ie1        C        0   0    4683        828
100.1.1.3/32      -              local      CP       0   0    1580        828
127.0.0.0/8       -              bh0        CP       0   0       0        828
127.0.0.1/32      -              local      CP       0   0       0        828
127.0.0.2/32      -              rj0        CP       0   0       0        828
1.1.2.0/24        -              ie0        C        0   0    1428        828
1.1.2.65/32       -              local      CP       0   0    2937        828
224.0.0.0/4       -              mcast      CP       0   0       0        828
224.0.0.1/32      -              local      CP       0   0       0        828
224.0.0.2/32      -              local      CP       0   0       0        828
224.0.0.9/32      -              local      CP       0   0       0        828
255.255.255.255/32 -             ie0        CP       0   0       0        828
```

Verifying the network processor setup for the interface

The network processor on the IP2000 creates a connection entry for the Gigabit Ethernet interface when the interface becomes operational. You can force the network processor to create a connection entry for the Gigabit Ethernet interface by using the following command:

```
admin> gmac -n
NP setup for gmac done.
```

Verifying the SAR setup for the interface

The Stinger Segmentation and Reassembly (SAR) creates an ATM connection entry for the Gigabit Ethernet interface. You can force the SAR setup by using the following command:

```
admin> gmac -s
GMAC: SAR conn. open with vpi = 0, vci = 200
```

Verifying IP packet transfer on the interface

The following command clears statistics gathered on the Gigabit Ethernet interface:

```
admin> gmac -d -c
```

The next command pings a host on the same subnet as the Gigabit Ethernet interface:

```
admin> ping 100.1.1.10
PING 100.1.1.10 (100.1.1.10): 56 data bytes
64 bytes from 100.1.1.10: icmp_seq=0 ttl=255 time=0 ms
64 bytes from 100.1.1.10: icmp_seq=1 ttl=255 time=0 ms
64 bytes from 100.1.1.10: icmp_seq=2 ttl=255 time=0 ms
64 bytes from 100.1.1.10: icmp_seq=3 ttl=255 time=0 ms
64 bytes from 100.1.1.10: icmp_seq=4 ttl=255 time=0 ms
```

```
64 bytes from 100.1.1.10: icmp_seq=5 ttl=255 time=0 ms
64 bytes from 100.1.1.10: icmp_seq=6 ttl=255 time=0 ms
64 bytes from 100.1.1.10: icmp_seq=7 ttl=255 time=0 ms
--- 100.1.1.10 ping statistics ---
8 packets transmitted, 8 packets received, 0% packet loss
round-trip min/avg/max = 0/0/0 ms
```

The following command displays GMAC statistics that show packet transfer. The txGoodPackets and rxGoodPackets fields in the command output show 8 packets transmitted and received in the ICMP sequence shown immediately above. For more details on the command output fields, see "Total transmit statistics" on page A-24 and "Total receive statistics" on page A-26.

```
admin> gmac -d
Gigabit Ethernet port statistics :

txOctetsLow     = 816
txOctetsHigh    = 0
txGoodPackets   = 8
txPkt64         = 0
txPkt65127      = 8
txPkt128255     = 0
txPkt256511     = 0
txPkt5121023    = 0
txPkt1024Max    = 0
txPktDefer      = 0
txPktUndSz      = 0
txUnderFlow     = 0
txPfcf          = 0
txPfcc          = 0
txRfcf          = 0
txRfcc          = 0
txOverFlow      = 0
txAlmostFull    = 0

rxOctetsLow     = 816
rxOctetsHigh    = 0
rxGoodPackets   = 8
rxPkt64         = 0
rxPkt65127      = 8
rx128255        = 0
rx256511        = 0
rx5121023       = 0
rx1024Max       = 0
rxMacType       = 0
rxCrcErrors     = 0
rxUnderSize     = 0
rxOverSize      = 0
rxAlmostFull    = 0
rxOverRun       = 0
rxMulticastPackets      = 0
rxBroadcastPackets      = 0
rxJabber        = 0
```

```
rxPfc           = 0
rxRfc           = 0
```

Gigabit Ethernet port redundancy

With the proper configuration, systems with redundant controllers support Gigabit Ethernet redundancy, which enables the system to maintain RFC 1483 (MPoA) connections, VLAN connections, and LAN MBONE interface functions across a controller switchover.

Note A soft IP interface configuration is required for Gigabit Ethernet redundancy of RFC 1483 connections and a LAN MBONE interface.

Configuring a soft IP interface for Gigabit Ethernet redundancy

The soft IP interface is an internal interface that is not associated with a specific physical port, but that can be accessed through the Ethernet interface of whichever controller is primary. For background information, see "Defining a soft interface for increased accessibility" on page 4-8.

The system creates one soft interface profile by default. For example:

```
admin> dir ip-interface
       35  07/10/2003 11:26:10  { { any-shelf any-slot 0 } 0 }
       35  07/10/2003 11:26:10  { { shelf-1 first-control-module 1 } 0 }
       38  07/10/2003 11:26:11  { { shelf-1 first-control-module 2 } 0 }
       35  07/10/2003 11:26:10  { { shelf-1 second-control-module 1 } 0 }
       38  07/10/2003 11:26:11  { { shelf-1 second-control-module 2 } 0 }
```

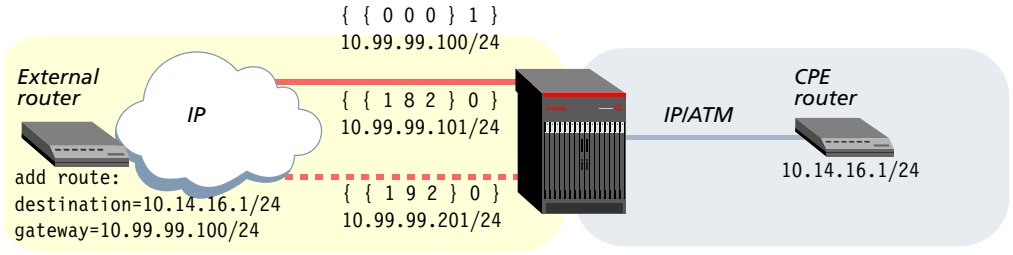
You can use the default soft IP interface { {0 0 0 } 0 } for Gigabit Ethernet redundancy. However, if you have already used the default profile for the soft IP address of the 10/100M base Ethernet management ports, you can create another soft IP interface using a profile index of { { 0 0 0 } *x* }, as long as the IP address in that profile is on the same subnet as the Gigabit Ethernet ports.

Note The system associates its Ethernet interfaces with a particular soft address based on the subnet assignment. The IP interface address of the Gigabit Ethernet ports on the primary and secondary controllers and the soft IP interface address must be on the same subnet.

Configuring Gigabit Ethernet redundancy for RFC 1483 connections

Figure 2-1 shows a Stinger with redundant IP2000 controllers. The Gigabit Ethernet port in slot 8 ({ { 1 8 2 } 0 }), the Gigabit Ethernet port in slot 9 ({ { 1 9 2 } 0 }), and the soft IP interface ({ { 0 0 0 } 1 }), all have IP address assignments on the same subnet. In addition, the external router has a routing table entry that specifies the soft IP interface address as the gateway to the CPE router destination.

Figure 2-1. Gigabit Ethernet redundancy for RFC 1483 connectivity



The following commands configure the Gigabit Ethernet port in slot 8:

```
admin> read ip-interface { { 1 8 2 } 0 }
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } read

admin> set ip-address = 10.99.99.101/24

admin> write
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } written
```

The next commands configure the Gigabit Ethernet port on slot 9:

```
admin> read ip-interface { { 1 9 2 } 0 }
IP-INTERFACE/{ { shelf-1 second-control-module 2 } 0 } read

admin> set ip-address = 10.99.99.201/24

admin> write -f
IP-INTERFACE/{ { shelf-1 second-control-module 2 } 0 } written
```

The following commands configure a soft IP interface on the same subnet:

```
admin> new ip-interface { { 0 0 0 } 1 }
IP-INTERFACE/{ { any-shelf any-slot 0 } 1 } read

admin> set ip-address = 10.99.99.100/24

admin> write -f
IP-INTERFACE/{ { any-shelf any-slot 0 } 1 } written
```

When you write the profile of the soft interface, the system displays a message:

```
LOG notice, Shelf 1, Controller-1, Time: 11:42:57--
Soft ip will be effective if the ip-addr of primary controller is
configured.
```

To ensure that the external router can reach the CPE router in Figure 2-1, the external router must specify the soft IP address as the gateway to the CPE router destination address. For example,

```
destination-address = 10.14.16.1/24
gateway-address = 10.99.99.100/24
```

Configuring Gigabit Ethernet redundancy for VLAN bridge circuits

Virtual LAN (VLAN) technology is supported with an optional software license. For details about configuring VLAN, see Chapter 3, “VLAN Configuration.”

A VLAN bridge circuit is a pairing between a unique VLAN ID on the Gigabit Ethernet port and a DSL subscriber connection on a LIM port. To enable the system to maintain the pairing following a controller switchover, you must configure the VLAN circuits using the expression *any-slot* or 0 as the slot number.

For example, the following commands create a new GigE-redundant VLAN bridge circuit with VLAN ID 300.

```
admin> new vlan-ethernet { { 1 0 2 } 300 }
VLAN-ETHERNET/{ { shelf-1 any-slot 2 } 300 } read

 admin> set enabled = yes

admin> set bridging-options bridging-group = 300

admin> set bridging-options bridge = yes

admin> write -f
VLAN-ETHERNET/{ { shelf-1 any-slot 2 } 300 } written
```

To modify an existing VLAN bridge circuit for Gigabit Ethernet redundancy, you must create a new configuration and then delete the old one. For example, the following command shows an existing VLAN bridge circuit with VLAN ID 50:

```
admin> dir vlan
37  07/21/2003 17:38:24  { { shelf-1 first-control-module 2 } 50 }
```

The next commands modify the VLAN bridge circuit to enable Gigabit Ethernet redundancy for the connection:

```
admin> read vlan { { 1 8 2 } 50 }
VLAN-ETHERNET/{ { shelf-1 first-control-module 2 } 50 } read

admin> set interface-address = { { 1 0 2 } 50 }
(New index value; will save as new profile VLAN-ETHERNET/{ { shelf-1
any-slot 2} 50 }.)

admin> write -f
VLAN-ETHERNET/{ { shelf-1 any-slot 2 } 50 } written
```

The following commands list and then delete the older `vlan-ethernet` profile:

```
admin> dir vlan
37  07/15/2003 09:00:30  { { shelf-1 any-slot 2 } 50 }
37  07/21/2003 17:38:24  { { shelf-1 first-control-module 2 } 50 }

admin> delete vlan { {1 8 2 } } 50}
Delete profile VLAN-ETHERNET/{ { shelf-1 first-control-module 2 } 50 }?
[y/n] y
VLAN-ETHERNET/{ { shelf-1 first-control-module 2 } 50 } deleted
```

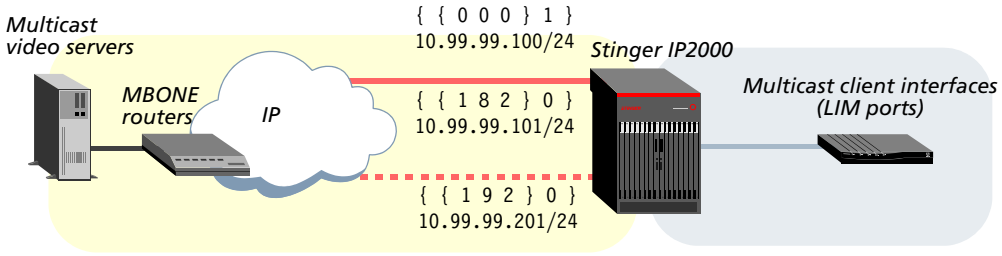
Configuring a redundant LAN MBONE

IP multicast forwarding is supported with an optional software license. For details about configuring it, see Chapter 8, "Forwarding Multicast Video."

To support redundancy for a LAN MBONE interface, you must configure a soft IP interface for the Gigabit Ethernet ports, enable multicast on both ports, and use the `any-slot` expression in the `mbone-lan-interface` parameter setting.

Figure 2-2 shows a Stinger with redundant IP2000 controllers. The Gigabit Ethernet port in slot 8 ({ { 1 8 2 } 0 }), the Gigabit Ethernet port in slot 9 ({ { 1 9 2 } 0 }), and the soft IP interface ({ { 0 0 0 } 1 }), all have IP address assignments on the same subnet and both physical ports enable multicast.

Figure 2-2. Gigabit Ethernet redundancy for a LAN MBONE



The following commands configure the Gigabit Ethernet port in slot 8:

```
admin> read ip-interface { { 1 8 2 } 0 }
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } read

admin> set ip-address = 10.99.99.101/24

admin> set multicast-allowed = yes

admin> write
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } written
```

The next commands configure the Gigabit Ethernet port on slot 9:

```
admin> read ip-interface { { 1 9 2 } 0 }
IP-INTERFACE/{ { shelf-1 second-control-module 2 } 0 } read

admin> set ip-address = 10.99.99.201/24

admin> set multicast-allowed = yes

admin> write -f
IP-INTERFACE/{ { shelf-1 second-control-module 2 } 0 } written
```

The following commands configure a soft IP interface on the same subnet:

```
admin> new ip-interface { { 0 0 0 } 1 }
IP-INTERFACE/{ { any-shelf any-slot 0 } 1 } read

admin> set ip-address = 10.99.99.100/24

admin> write -f
IP-INTERFACE/{ { any-shelf any-slot 0 } 1 } written
```

The following commands enable the multicast forwarding function and specify a redundant LAN MBONE configuration:

```
admin> read ip-global
IP-GLOBAL read

admin> set multicast-forwarding = yes

admin> set multiple-mbone mbone-lan-interface 1 = { { 1 0 2 } 0 }

admin> write -f
IP-GLOBAL written

admin> list multiple-mbone mbone-lan-interface
[in IP-GLOBAL:multiple-mbone:mbone-lan-interface]
mbone-lan-interface[1] = { { shelf-1 any-slot 2 } 0 }
mbone-lan-interface[2] = { { any-shelf any-slot 0 } 0 }
mbone-lan-interface[3] = { { any-shelf any-slot 0 } 0 }
```

```
mbone-lan-interface[4] = { { any-shelf any-slot 0 } 0 }
```

Note With this configuration, the LAN MBONE is supported on the Gigabit Ethernet port of the controller in slot 8 or slot 9, whichever is primary. Following a switchover, each IGMP client must rejoin its group to receive multicast traffic.

Administrative tools for Gigabit Ethernet

The system supports the `gmac` command for administrative information about Gigabit Ethernet ports. If you are managing the system remotely, some of this information is also available through the `ip2kstats` MIB. For details, see "Gigabit Ethernet diagnostics" on page A-2 and "SNMP MIB for GMAC and VLAN statistics" on page A-23. For other commands that can be used to monitor activity on any Ethernet port, such as `etherdisplay`, see the *Stinger Reference*.

VLAN Configuration

3

Virtual LAN (VLAN) technology is supported with an optional software license. VLANs are defined in *IEEE standard 802.1Q (1998) for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks*.

VLANs are typically used for making physically separate nodes appear as if they are part of a LAN. A VLAN helps to divide a network into separate broadcast domains without the latency problems typically seen in routed networks, and without modifying the physical topology. VLANs also allow a network administrator to group users logically irrespective of where they are located physically.

Enter the following command to determine whether the VLAN license is enabled:

```
admin> get base vlan
[in BASE]
vlan-enabled = yes
```

The system sets this parameter to `yes` when the VLAN license is enabled. If the license is not enabled, the system displays an error message if you configure VLAN capabilities. For information about obtaining and enabling Lucent Technologies software licenses, contact your Lucent sales representative.

The IP2000 VLAN implementation

The IP2000 VLAN implementation can support up to 4096 VLAN IDs on the Gigabit Ethernet port. For full compatibility with IEEE 802.1Q standard, however, Lucent recommends that you do not use 0 or 4095 as VLAN IDs. As a result, as many as 4093 logically separate VLAN IDs are supported.

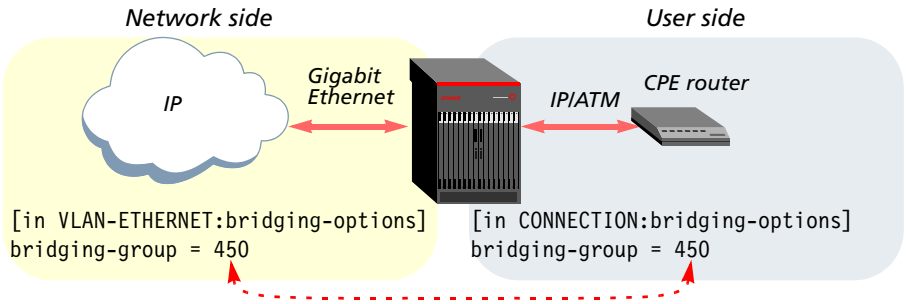
In compliance with IEEE 802.1Q, the Stinger adds 4-byte tags to the header of each Ethernet frame. The tags contain a VLAN ID and IEEE 802.1P priority information. Most Ethernet switches and routers can use these tags to direct the frames only to the specified VLAN. Typically, each VLAN ID on the Gigabit Ethernet interface is paired to form a bridge circuit with one DSL interface. However, management VLANs, used for SNMP or Telnet access from the LAN to the system itself, can also be configured with the current software.

VLAN bridge circuits

A VLAN bridge circuit is a pairing between a unique VLAN ID on the Gigabit Ethernet port and a DSL subscriber connection on a LIM port. The transfer of traffic between the two interfaces within the Stinger unit occurs at Layer 2. The Stinger does not examine Layer 3 information within the data stream.

You create a VLAN bridge circuit by specifying the same `bridging-group` number in both the `vlan-ethernet` and `connection` (or RADIUS) user profile. This is illustrated in Figure 3-1.

Figure 3-1. Bridge circuit between a virtual LAN and bridged WAN interface



When you configure a bridging group, the system creates a separate bridging table that contains only interfaces that share the same group number. When a packet is received on an interface in a bridging group, the system consults only that table for destination interfaces. It will not forward the traffic to interfaces that are not in the same bridging group.

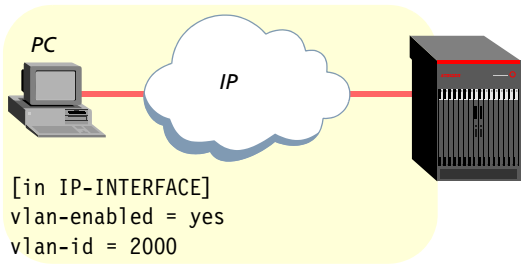
Note With the current software, you cannot bridge multiple LIM-side PVCs in the same system to the same VLAN ID.

Local management VLANs

You can configure a virtual IP interface as a management VLAN, which terminates on the control module and provides access to the system via Telnet or SNMP through the Gigabit Ethernet interface. The system supports up to 16 virtual IP interfaces. For details, see “Defining a local virtual IP interface” on page 4-8.

A management VLAN is not paired with a WAN interface. It is used solely for management access to the Stinger unit, as shown in Figure 3-2.

Figure 3-2. Management VLAN terminating in the Stinger unit



Configuring a VLAN bridge circuit

To configure a VLAN bridge circuit, you must complete the following steps:

1. Verify that bridging is enabled in the `ethernet` profile for the Gigabit Ethernet interface. For background information, see Chapter 2, "Gigabit Ethernet Configuration."
2. Create a `vlan-ethernet` profile with a unique VLAN ID and `bridging-group` number.
3. Configure a `connection` or RADIUS profile that specifies the same `bridging-group` number associated with the VLAN ID.

Note For Stinger systems with redundant controllers, you can configure Gigabit Ethernet redundancy for VLAN bridge circuits to enable the system to maintain VLAN operations across a controller switchover. For details, see "Configuring Gigabit Ethernet redundancy for VLAN bridge circuits" on page 2-7.

Overview of VLAN configuration settings

You define a VLAN in a `vlan-ethernet profile`, which must specify a unique VLAN ID from 0 to 4095, and a `bridging-group` number.

Note To maintain full compatibility with the IEEE 802.1Q standard, Lucent recommends that you do not assign the VLAN ID values of 0, 1 and 4095. However, the system does not prevent you from assigning these values.

You associate a VLAN ID with a DSL interface by specifying the same `bridging-group` number in the `vlan-ethernet` profile and a `connection` or RADIUS profile. With the current software, a VLAN bridge circuit can contain only two interfaces. You cannot map multiple DSL interfaces to the same VLAN ID in the same Stinger unit.

Configuring **`vlan-ethernet`** settings

The index of a `vlan-ethernet` profile specifies the physical address of the Gigabit Ethernet port and a unique VLAN ID. Following are the profile contents, shown for VLAN ID 50:

```
[in VLAN-ETHERNET/{ { shelf-1 first-control-module 2 } 50 }]
interface-address* = { { shelf-1 first-control-module 2 } 50 }
vlan-id = 50
enabled = no
filter-name = ""
pppoe-options = { no no }
[in VLAN-ETHERNET/{ { shelf-1 first-control-module 2 } 50}:bridging-options]
bridging-group = 0
bridge = no
dial-on-broadcast = no
```

Parameter	Setting
`interface-address`	Address of the Gigabit Ethernet port followed by the VLAN ID, using the following format: { *`shelf-n slot-n port-n`* } *`vlan-id`* The *`slot-n`* is 8 (for the first control module) or 9 (the second control module), and *`port-n`* is 2 for the Gigabit port. The *`vlan-id`* value is the IEEE 802.1Q VLAN tag value added to the IP packets transmitted on the Gigabit Ethernet interface. The valid range is from 0 to 4095, but for full compatibility with IEEE 802.1Q, Lucent recommends that you do not use the *`vlan-id`* values of 0, 1 or 4095.
`vlan-id`	VLAN ID. This setting is read-only. You must set it in the index of the `vlan-ethernet` profile.
`enabled`	Enable/disable the `vlan-ethernet` profile.
`filter-name`	*Not used by the IP2000.*
`pppoe-options`	*Not used by the IP2000.*
`bridging-options: bridging-group`	A group number (from 0 to 65535) for the bridge circuit between the VLAN and a WAN interface. The `bridging-group` parameter in this profile must match the `bridging-group` parameter in the `connection` profile that the `vlan-id` maps to.
`bridging-options: bridge`	Enable/disable WAN packet bridging on the interface. With WAN bridging, the system can provide a connection between segments that are connected by a telecommunications link. Set this parameter to `yes` to enable WAN bridging.
`dial-on-broadcast`	*Not used by the IP2000.*

Configuring DSL subscriber **`bridging-options`** settings

To pair a DSL interface with a VLAN ID, the connection or RADIUS profile must enable bridging and specify the right ID for the destination VLAN. The far-end device can be an IP router or a bridge. It is only within the Stinger unit that the VLAN ID is inserted in the packet stream and the bridging occurs to the Gigabit Ethernet interface.

For information about configuring a PVC to an IP router, see "Configuring IP connection interfaces for CPE devices" on page 4-31. For information about ATM PVCs, see the *Stinger ATM Configuration Guide*.

In addition to the usual settings, you must also specify the following parameters, shown with default settings, in the DSL interface `connection` profile:

```
[in CONNECTION/"":bridging-options]
bridging-group = 0
bridge = no
dial-on-broadcast = no
```

Parameter	RADIUS attribute	Setting
`bridging-group`	Ascend-BIR-Bridge-Group (7)	Number from 0 to 65535, used to group this subscriber interface with a VLAN. Note that you cannot group multiple LIM-side PVCs in the same system with the same VLAN.
`bridge`	Ascend-Bridge (230)	Enable/disable WAN packet bridging on the interface.
`dial-on-broadcast`	N/A	*Not used by the IP2000.*

Enabling bridging on the Gigabit Ethernet interface

You must enable bridging on the Gigabit Ethernet port once, to provide bridging support for all configured VLAN IDs. For details about the other settings in the `ethernet` profile, see Chapter 2, "Gigabit Ethernet Configuration."

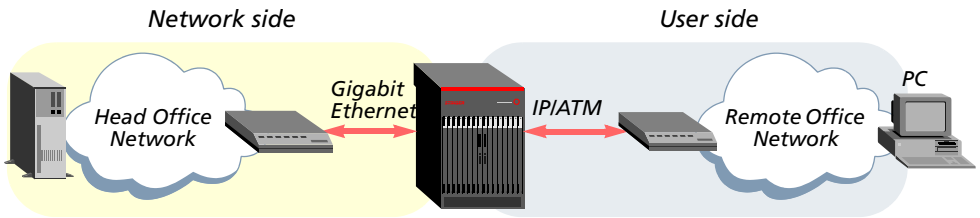
The following commands enable bridging on the physical interface:

```
admin> read ethernet { 1 8 2 }
ETHERNET/{ shelf-1 first-control-module 2 } read

admin> set bridging-enabled = yes

admin> write
ETHERNET/{ shelf-1 first-control-module 2 } written
```

Sample VLAN bridge circuit configuration

In the sample setup shown in Figure 3-3, the remote side of the VLAN bridge circuit connects to a CPE on a remote office LAN. The Gigabit Ethernet side connects to a router that can access the LAN of the head office.

Figure 3-3. Sample VLAN bridge circuit



In this sample setup, the Stinger receives traffic from the CPE, packetizes it and inserts the VLAN ID in Ethernet frame headers, and bridges the packet stream across its Gigabit Ethernet port. It does not terminate the packets by passing them up to the IP router software.

The external router on the Gigabit Ethernet network interprets the VLAN ID and directs the packet stream at Layer 2 to the specified VLAN, which is the head office network.

The following commands configure the Gigabit Ethernet side of the VLAN bridge circuit with `vlan-id` 50 and the `bridging-group` number 34590:

```
admin> new vlan-ethernet { { 1 8 2 } 50}
VLAN-ETHERNET/{ { shelf-1 first-control-module 2 } 50} read

admin> set enabled = yes
```

```
admin> set bridging-options bridging-group = 34590

admin> write
VLAN-ETHERNET/{ { shelf-1 first-control-module 2 } 50} written
```

The next command identifies the nailed group for subscriber port (in this example, port 1 of the LIM in slot 2 of the unit):

```
admin> which -n { 1 2 1 }
Nailed group corresponding to port { shelf-1 slot-2 1 } is 51
```

The following commands configure a `connection` profile on that interface for the LIM side of the VLAN bridge circuit:

```
admin> new CONNECTION dslcpe
CONNECTION/dslcpe read

admin> set active = yes

admin> set encapsulation-protocol = atm

admin> set bridging-options bridging-group = 34590

admin> set bridging-options bridge = yes

admin> set atm-options vpi = 8

admin> set atm-options vci = 100

admin> set atm-options nailed-group = 51

admin> write -f
CONNECTION/dslcpe written
```

Following is a comparable RADIUS profile for the LIM interface side of the VLAN bridge circuit:

```
permconn-st-1 Password = "ascend"
    Service-Type = Outbound,
    Framed-Protocol = ATM-1483,
    User-Name = "dslcpe",
    Ascend-Route-IP = Route-IP-Yes,
    Ascend-ATM-Group = 51,
    Ascend-ATM-Vpi = 8,
    Ascend-ATM-Vci = 100,
     Ascend-Bridge = Bridge-Yes,
    Ascend-BIR-Bridge-Group = 34590
```

The LAN session should come up, displaying log messages such as the following:

```
LOG notice, Shelf 1, Slot 2, Time: 20:24:02--
Line 1 up
LOG info, Shelf 1, Controller-1, Time: 20:24:02--
[1/2/1/0] Assigned to port [MBID 1]
LOG info, Shelf 1, Slot 2, Time: 20:24:02--
[1/2/1/0] LAN session up: <dslcpe> [MBID 1] [dslcpe]
```

The following command shows the active session:

```
admin> users
SessionID    Line/Chan    Slot:Item    Tx/Rx Rate  Svc  Address    Username
385031879    1.02.01/000  1:02:01/000  8000K/832K  ATM  0.0.0.0    dslcpe
<end user list> 1 active user(s)
```

For information about low-level diagnostic commands for viewing bridge tables, see Appendix A, "IP2000 Diagnostics."

Configuring a local management VLAN

You can configure a virtual IP interface as a management VLAN, to isolate management traffic and provide additional security for the logins to the system. Like any VLAN configuration, this requires that bridging is enabled on the Gigabit Ethernet interface. For details, see "Enabling bridging on the Gigabit Ethernet interface" on page 3-5.

Overview of `ip-interface` VLAN settings

Following are the parameters, shown with default settings for a virtual IP interface, for configuring a management VLAN:

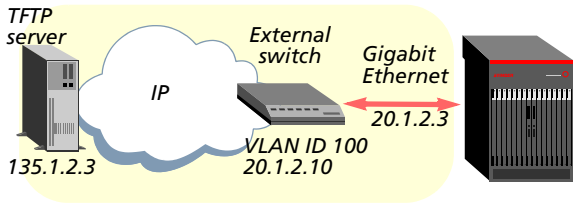
```
[in IP-INTERFACE/{ { shelf-1 first-control-module 2 } 1 }]
interface-address* = { { shelf-1 first-control-module 2 } 1 }
ip-address = 0.0.0.0/0
vlan-enabled = no
vlan-id = 0
```

Parameter	Setting
`interface-address`	Address of the Gigabit Ethernet interface followed by the virtual IP interface number, using the following format: { *shelf-n slot-n port-n* } *item-n* } The *slot-n* is 8 (for the first control module) or 9 (the second control module), and *port-n* is 2 for the Gigabit port. The *item-n* value is a number from 1 to 16, identifying the virtual IP interface.
`ip-address`	IP address of the virtual IP interface
`vlan-enabled`	Enable/disable IEEE 802.1Q VLAN tagging on the virtual IP interface.
`vlan-id`	VLAN ID for this virtual interface. This is the IEEE 802.1Q tag value to be added to the IP packets transmitted on the virtual interface. The valid range is from 0 to 4095, but for full compatibility with IEEE 802.1Q, Lucent recommends that you do not use the *vlan-id* values of 0, 1 or 4095.

Sample management VLAN configuration

In the sample setup shown in Figure 3-4, the IP2000 Gigabit Ethernet interface supports a switch configured with a VLAN ID of 100. For information about displaying the VLAN interface after creating it, see the `ifmgr` command in Appendix A, "IP2000 Diagnostics."

Figure 3-4. Sample management VLAN



The following commands configure a management VLAN on a virtual IP interface of the Gigabit Ethernet port:

```
admin> read ip-interface { { 1 8 2 } 1 }
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 1 } read

admin> set ip-address = 20.1.2.3/24

admin> set vlan-enabled = yes

admin> set vlan-id = 100

admin> write -f
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 1 } written
```

While logged into the Stinger unit, you should be able to `ping` the external switch:

```
admin> ping 20.1.2.10
PING 20.1.2.10 (20.1.2.10): 56 data bytes
64 bytes from 20.1.2.10: icmp_seq=0 ttl=255 time=10 ms
64 bytes from 20.1.2.10: icmp_seq=1 ttl=255 time=0 ms
64 bytes from 20.1.2.10: icmp_seq=2 ttl=255 time=0 ms
64 bytes from 20.1.2.10: icmp_seq=3 ttl=255 time=0 ms
64 bytes from 20.1.2.10: icmp_seq=4 ttl=255 time=0 ms
64 bytes from 20.1.2.10: icmp_seq=5 ttl=255 time=0 ms
64 bytes from 20.1.2.10: icmp_seq=6 ttl=255 time=0 ms
64 bytes from 20.1.2.10: icmp_seq=7 ttl=255 time=0 ms
^C
--- 20.1.2.10 ping statistics ---
8 packets transmitted, 8 packets received, 0% packet loss
round-trip min/avg/max = 0/1/10 ms
```

The following commands set up a static route to reach the TFTP server across the Gigabit Ethernet link:

```
admin> new ip-route tftp-server
IP-ROUTE/tftp-server read

admin> set dest-address = 135.1.2.3

admin> set gateway-address = 20.1.2.10

admin> write -f
IP-ROUTE/tftp-server written
```

You should now be able to download software from the TFTP server to the Stinger flash memory:

```
admin> load cm-v2 n 135.1.2.3 stngrcm2.ffs
loading code from 135.1.2.3
file stngrcm2.ffs...
done.
```

Administrative tools for VLAN

Commands that provide administrative information about VLAN are available only in the debug environment. If you are managing the system remotely, some of this information is also available through the `ip2kstats` MIB. For details, see "VLAN-related diagnostics" on page A-13 and "SNMP MIB for GMAC and VLAN statistics" on page A-23.

IP Routing

4

This chapter describes IP routing features that are typically configured on a Stinger IP2000. Some parameters in IP-related profiles are not used by the IP2000, or are not relevant to its primary applications. Those parameters are not described in this chapter, but are documented in the *Stinger Reference*.

Introduction to the IP router software

When you reset the system, an IP routing table is constructed that contains all the routes known to the system, including the following:

- Routes for the local Ethernet interfaces (configured `ip-interface` profiles)
- Routes for active WAN IP sessions
- Routes for inactive WAN IP sessions (configured `connection` profiles)
- Routes defined in `ip-route` profiles or RADIUS route profiles

If the Routing Information Protocol (RIP) is enabled on one or more interfaces, the system adds routes as it learns them from routing-update packets. In addition, the system is continuously updating its routing table by adding routes for links that become active and removing routes for inactive sessions. If a nailed connection goes down, the system removes the route from its routing table.

Routes and interfaces

An IP route specifies a destination address, a gateway to the network, and an interface that leads to the gateway. It can also specify metrics and other values associated with the route.

A route defined in a profile is a *static route*. A *dynamic route* is learned from RIP updates sent by other routers. Dynamic updates provide access to many more routes than those actually configured in the system, and are updated automatically as routes change. However, dynamic updates cause additional routing overhead, so they are disabled by default.

An *interface* is a point of ingress to or egress from the system. For example, a local interface is an Ethernet port and a WAN interface is a nailed or switched connection. An *IP interface* is the logical IP address that enables IP data to be sent and received.

Displaying the routing table

For details about the `netstat` command, see the *Stinger Reference*. The following command displays the system's routing table:

```
admin> netstat -r
Destination       Gateway         IF          Flg   Pref Met     Use        Age
0.0.0.0/0         1.112.26.1     ie0         SGP     60   1     343        2274
127.0.0.0/8       -               bh0         CP       0   0       0        2274
127.0.0.1/32      -               local       CP       0   0       0        2274
127.0.0.2/32      -               rj0         CP       0   0       0        2274
1.112.0.0/16      -               ie0         C        0   0    6497        2274
1.112.26.146/32   -               local       CP       0   0    3635        2274
224.0.0.0/4       -               mcast       CP       0   0     179        2274
224.0.0.1/32      -               local       CP       0   0       0        2274
224.0.0.2/32      -               local       CP       0   0       0        2274
224.0.0.9/32      -               local       CP       0   0       0        2274
255.255.255.255/32-               ie0         CP       0   0      61        2274
Total Routes = 11      Hidden Routes = 0
```

For each route in the table, the `Destination` and `Gateway` fields show the destination address and the address of the next-hop router used to reach that destination. The zero destination address is the default route. If the system does not find a route for a packet's destination, it forwards the packet to the default route rather than dropping the packet. Note that the system uses the most specific route (having the longest prefix) that matches a given destination. Direct routes do not show a gateway address.

An asterisk (*) in the flags column indicates a hidden route, which is not included in routing updates sent by the system and is not used for forwarding packets. Hidden routes are used only for display purposes.

The `IF` field shows the name of the interface through which a packet addressed to the entry's destination will be sent. The route to the `mcast` interface name encapsulates the multicast forwarder for the entire class D address space. (For more information, see Chapter 8, "Forwarding Multicast Video.")

Routes to the local unit display the `local` interface name. Packets to the 224.0.0.1 and 224.0.0.2 interfaces can be multicast and received like normal multicast packets, but upon receiving such a packet, the router does not forward it to another link layer device. Effectively, these packets have a maximum transmission unit (MTU) of 1.

Displaying the interface table

To display the interface table, use the `–i` option on the `netstat` command line:

```
admin> netstat -i
Name       MTU   Net/Dest          Address         Ipkts    Ierr Opkts    Oerr
ie0        1500  1.112.0.0/16      1.112.26.146     5542    0     1636     0
ie1        1500  -                   -                 0    0        0     0
ie1-1      1500  -                   -                 0    0        0     0
lo0        1500  127.0.0.1/32        127.0.0.1      1629    0     1629     0
rj0        1500  127.0.0.2/32        127.0.0.2         0    0        0     0
bh0        1500  127.0.0.3/32        127.0.0.3         0    0        0     0
wanabe     1500  127.0.0.3/32        127.0.0.3         0    0        0     0
local     65535  127.0.0.1/32        127.0.0.1      1892    0     1892     0
mcast     65535  224.0.0.0/4         224.0.0.0       180    0      180     0
tunnel0   1500   1.112.0.0/16      1.112.26.146        0    0        0     0
vr0_main  1500   1.112.26.146/32   1.112.26.146        0    0        0     0
sip0      65535  -                   -                 0    0        0     0
```

The entries named `ie0` or `ie`*`N-N-N`*`[-`*`N`*`]` represent Ethernet interfaces. *N-N-N-N* represents the shelf number, slot number, item number, and logical-item number of the interface. When the logical-item number is zero (the physical interface), it does not appear in the interface name. The same sequence of numbers forms the address used to index the `ip-interface` profile. For example, the default profile for 1-8-2 is indexed as follows:

```
ip-interface { { 1 8 2 } 0 }
```

When the logical-item number is *not* zero, it does appear in the interface name. Again, the sequence of numbers is identical to the profile index. For example, suppose an `ip-interface` profile has the following index:

```
ip-interface { { 1 8 2 } 3 }
```

This profile has the following interface name:

```
ie1-8-2-3
```

The other names in the interface table have the following significance:

- The `lo0` (loopback) interface is the local loopback.
- The `rj0` (reject) and `bh0` (blackhole) interfaces are used in the pool-summary feature.
- The `wanabe` interface is an inactive RADIUS dial-out profile.
- The `local` interface is the local machine.
- The `mcast` interface is the multicast interface, which represents the multicast forwarder for the entire class D address space. For details, see Chapter 8, “Forwarding Multicast Video.”
- The `tunnel` interface is a single pseudo-interface that is used only when the system is terminating tunnels. (The number terminating the tunnel interface name is an internal number that can change from one software version to the next.)
- The `vr0_main` interface represents the router itself.
- The `sip0` interface is the soft IP interface. For details, see “Defining a soft interface for increased accessibility” on page 4-8.

- The numbered WAN (`wan`*N*) interfaces are WAN connections, which are entered in the interface table as they become active.

IP2000 performance statistics

The IP2000 controller collects statistics on the number of packets and octets transmitted and received on each LIM interface. These counters are represented in the output of the `netstat -i` and `ifstat` commands, and are accessible to an external management utility.

The per-interface statistics for connections terminated on the IP2000 are cleared on the LIM when they are displayed on the controller by using the `netstat -i` or `ifstat` command.

For example, in the following output, the `Ipkts`, `Ierr`, `Opkts`, and `Oerr` statistic counters display the sum of the packets in transit as seen by the IP2000 added to the current packets counted by the TAOS interface manager.

```
admin> netstat -i
Name     MTU   Net/Dest         Address         Ipkts   Ierr   Opkts Oerr
ie0      1500  10.1.26.0/24   10.1.26.1     605504    0         0    0
ie1      1500  15.1.1..0/24   15.1.1.1           0    0         0    0
lo0      1500  127.0.0.1/32   127.0.0.1          8    0         8    0
rj0      1500  127.0.0.2/32   127.0.0.2          0    0         0    0
bh0      1500  127.0.0.3/32   127.0.0.3          0    0         0    0
wanabe   1500  127.0.0.3/32   127.0.0.3          0    0         0    0
local   65535  127.0.0.1/32   127.0.0.1      58935    0     58935    0
mcast   65535  224.0.0.0/4    224.0.0.0          0    0         0    0
tunnel0  1500  10.1.26.0/24   10.1.26.1          0    0         0    0
vr0_main 1500  10.1.26.1/32   10.1.26.1          0    0         0    0
sip0     65535  -               -                 0    0         0    0
wan11    1524  200.200.200.254 2.2.2.1           7    0        10    0
```

Similarly, in the following output, the `in_oct`, `in_errs`, `out_octet`, and `out_err` statistic counters display the sum of the packets in transit as seen by the IP2000 added to the current packets counted by the TAOS interface manager.

```
admin> ifstat 1
in_oct 0 in_errs 0 out_octet 0 out_err 0
```

The MIB II interface stat counters also now display the correct values when viewed from an external management utility.

IP address syntax

The system uses dotted decimal format (not hexadecimal) for IP addresses. If no subnet mask is specified, the system uses a default mask based on the address class. For example, a class C address, such as 198.5.248.40, has 24 network bits, leaving 8 bits for the host portion of the address. If no subnet mask is specified for a class C address, the system uses the default mask of 24 bits.

A subnet address includes a prefix length, which specifies the number of network bits in the address. For example, the following address specifies a 29-bit subnet:

```
ip-address = 198.5.248.40/29
```

In this address, 29 bits of the address are used to specify the network. The three remaining bits are used to specify unique hosts on the subnet. With three bits used to

specify hosts on a 29-bit subnet, eight different bit combinations are possible. Of those eight possible host addresses, two are reserved:

000 — Reserved for the network (base address)
001
010
100
110
101
011
111 — Reserved for the broadcast address of the subnet

Note Be careful with zero subnets (subnets with the same base address as a class A, B, or C network). Early implementations of TCP/IP did not allow them. For example, the subnet 192.32.8.0/30 was illegal because it had the same base address as the class C network 192.32.8.0/24, while the subnet 192.32.8.4/30 was legal. Modern implementations of TCP/IP support zero subnets, and the Stinger implementation of RIP treats these subnets the same as any other network. However, you must treat zero subnets consistently throughout your network. Otherwise, you will encounter routing problems.

Table 4-1 shows subnet masks and prefix lengths for a class C network number.

Table 4-1. Decimal subnet masks and corresponding prefix lengths

Subnet mask	Number of host addresses	Prefix length
255.255.255.0	254 hosts + 1 broadcast, 1 network base	/24
255.255.255.128	126 hosts + 1 broadcast, 1 network base	/25
255.255.255.192	62 hosts + 1 broadcast, 1 network base	/26
255.255.255.224	30 hosts + 1 broadcast, 1 network base	/27
255.255.255.240	14 hosts + 1 broadcast, 1 network base	/28
255.255.255.248	6 hosts + 1 broadcast, 1 network base	/29
255.255.255.252	2 hosts + 1 broadcast, 1 network base	/30
255.255.255.254	Invalid mask (no hosts)	/31
255.255.255.255	1 host—a host route	/32

The broadcast address of any subnet has the host portion of the IP address set to all ones. The network address (or base address) represents the network itself, because the host portion of the IP address is all zeros. For example, supposing the IP configuration assigns the following address to a remote router:

```
198.5.248.120/29
```

The Ethernet network attached to that router has the following address range:

```
198.5.248.120 – 198.5.248.127
```

A host route is a special-case IP address with a prefix length of /32. For example:

```
198.5.248.40/32
```

Host routes are to a single host, rather than to a router or subnet.

Configuring ip-interface profiles for Ethernet ports

The system creates an ip-interface profile for an Ethernet port when it first detects the presence of the port. For example, the following output shows the default ip-interface profiles for the soft interface (the profile with the zero index) and the IP2000 controller:

```
admin> dir ip-interface
    21  07/24/2003 13:55:31  { { any-shelf any-slot 0 } 0 }
    31  07/24/2003 22:46:34  { { shelf-1 first-control-module 1 } 0 }
    21  07/24/2003 13:57:01  { { shelf-1 first-control-module 2 } 0 }
    36  07/24/2003 17:34:13  { { shelf-1 first-control-module 2 } 1 }
    21  07/24/2003 13:55:31  { { shelf-1 second-control-module 1 } 0 }
    21  07/24/2003 13:57:01  { { shelf-1 second-control-module 2 } 0 }
```

The profile for the Gigabit Ethernet interface on the first IP2000 controller (in slot 8) uses the following index:

```
{ { shelf-1 first-control-module 2 } 0 }
```

This index consists of a physical address and a logical-item number in the following format:

{ { *shelf-num slot-num item-num* } *logical-item-num* }

The logical item addresses a specific logical interface. It is zero except when multiple (virtual) interfaces have been configured on the physical port. For more details, see "Defining a local virtual IP interface" on page 4-8.

Overview of typical local interface settings

For information about enabling IP multicast forwarding on the Gigabit Ethernet interface, see Chapter 8, "Forwarding Multicast Video."

Following are the parameters, shown with default settings, used to configure the IP2000 Gigabit Ethernet port as an IP interface:

```
[in IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 }]
interface-address* = { { shelf-1 first-control-module 2 } 0 }
ip-address = 0.0.0.0/0
rip-mode = routing-off
rip2-use-multicast = yes
directed-broadcast-allowed = yes
vlan-enabled = no
vlan-id = 0
```

Parameter	Setting
interface-address	Address of the interface in the Stinger unit, or, if the item number is not zero, the virtual interface address.
ip-address	IP address of the LAN interface. If the LAN IP address includes a subnet specification, you must create a static route to another LAN router to enable the system to reach local networks beyond its own subnets. For details, see "Configuring ip-route profiles" on page 4-29.

Parameter	Setting
rip-mode	Enable/disable RIP updates on the interface. You can enable RIP to receive routing table updates, send them, or both. Running RIP-2 and RIP-v1 on the same network in such a way that the routers receive each other's advertisements is *not recommended*.
rip2-use-multicast	Enable/disable use of the multicast address (224.0.0.9) rather than the broadcast address for RIP updates.
directed-broadcast-allowed	Enable/disable forwarding of directed broadcast traffic onto the interface and its network.
vlan-enabled	Enable VLAN for this IP interface, to isolate management traffic to the unit. See Chapter 3, "VLAN Configuration."
vlan-id	See Chapter 3, "VLAN Configuration."

Configuring a local IP interface

The following command lists the ip-interface profiles created by the system for a Stinger unit with redundant IP2000 controllers:

```
admin> dir ip-interface
    21  07/24/2003 13:55:31  { { any-shelf any-slot 0 } 0 }
    31  07/24/2003 22:46:34  { { shelf-1 first-control-module 1 } 0 }
    21  07/24/2003 13:57:01  { { shelf-1 first-control-module 2 } 0 }
    21  07/24/2003 13:55:31  { { shelf-1 second-control-module 1 } 0 }
    21  07/24/2003 13:57:01  { { shelf-1 second-control-module 2 } 0 }
```

The next command assigns an IP address to the Gigabit Ethernet port of the first controller (installed in slot 8):

```
admin> read ip-interface { { 1 8 2 } 0 }
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } read

admin> set ip-address = 1.1.1.2/29

admin> write
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } written
```

In this example configuration, the Gigabit Ethernet interface is connected to the 1.1.1 subnet. RIP is off by default, so to enable the interface to communicate with routers on other local subnets, either the system must have a static route configuration to another router in its own subnet, or the interface must enable RIP. For an example of configuring a static route, see "Configuring ip-route profiles" on page 4-29.

The following commands configure the interface to receive RIP-2 updates on the multicast address (the multicast address is the default):

```
admin> read ip-interface { { 1 8 2 } 0 }
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } read

admin> set rip-mode = routing-recv-v2

admin> write
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } written
```

You can verify that the system can transfer IP packets across the interface by pinging another host on the same network segment, as shown in the following example:

```
admin> ping 1.1.1.19
PING 1.1.1.19: 56 Data bytes
64 bytes from 1.1.1.19: icmp_seq=0 ttl=255 time=0 ms
64 bytes from 1.1.1.19: icmp_seq=3 ttl=255 time=0 ms
^C
```

Defining a local virtual IP interface

You can configure up to 16 `ip-interface` profiles for each IP2000 module as a whole, with each profile specifying one IP address. For details about using a virtual IP interface for a management VLAN, see "Configuring a local management VLAN" on page 3-7.

The system creates the default profile for an interface and assigns it the zero logical-item number. To configure another IP address on a LAN interface, create an `ip-interface` profile with a nonzero logical-item number in its interface address. For example, the following commands create a virtual interface for the Gigabit Ethernet port:

```
admin> read ip-interface { { 1 8 2 } 1 }
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 1 } read

admin> set ip-address = 1.1.1.1/29

admin> write
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 1 } written
```

The logical-item numbers do not have to be consecutive, but they must each be unique.

Note The default `ip-interface` profile (with the zero logical-item number) must have an IP address configured. Otherwise, none of the other `ip-interface` profiles for the same port can function. (Do not delete the default profile and expect your other configurations to work.)

Defining a soft interface for increased accessibility

You can configure a soft IP interface, which is an internal IP interface that is always active and reachable, as long as one of the system's IP interfaces is up. The `ip-interface` profile with the zero index is reserved for the soft interface.

Note Do not specify the IP address of a physical LAN interface as the soft interface address.

The following commands set the soft interface IP address to 1.1.1.128:

```
admin> read ip-interface { { 0 0 0 } 0 }
IP-INTERFACE/{ { any-shelf any-slot 0 } 0 } read

admin> set ip-address = 1.1.1.128

admin> write
IP-INTERFACE/{ { any-shelf any-slot 0 } 0 } written
```

If RIP is enabled, the system advertises the soft interface address as a host route (with a prefix length of 32 bits) using the loopback interface. If RIP is not enabled, routers one hop away from the unit must have a static route to the soft interface address.

To verify that other hosts in your network have a route to the soft address, run `ping` or `traceroute` from the other hosts. For example:

```
host1% ping 1.1.1.128
PING 1.1.1.128 (1.1.1.128): 56 Data bytes
64 bytes from 1.1.1.128: icmp_seq=0 ttl=255 time=0 ms
64 bytes from 1.1.1.128: icmp_seq=7 ttl=255 time=0 ms
^C
```

Disabling directed broadcasts to protect against denial-of-service

Denial-of-service attacks known as *smurf* attacks typically use ICMP Echo Request packets with a spoofed source address and packets directed to IP broadcast addresses. These attacks are intended to degrade network performance, possibly to the point that the network becomes unusable.

To prevent the IP router from being used as an intermediary in this type of denial-of-service attack launched from another network, you must disable the router from forwarding directed broadcasts it receives from another network. You must explicitly disable directed broadcasts on *all* IP interfaces in the system (including the management interface). In a system with redundant controllers, disable the feature on both controllers, so the unit is still protected following a switchover. The following commands configure the Gigabit Ethernet interface on the first controller:

```
admin> read ip-interface { { 1 8 2 } 0 }
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } read

admin> set directed-broadcast-allowed = no

admin> write
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } written
```

Configuring ip-global network features

The IP router has many configuration settings that affect its operations. The settings that determine its routing policies include security, RIP options, IP route cache options, and other options. These settings are available only in the ip-global profile. They have no counterpart in RADIUS.

Of the many settings in the ip-global profile, some are intended for remote access service and are not directly relevant to IP2000 operations. However, there are many more options you can choose to configure in this profile. For details about all ip-globalparameters and subprofiles, see the *Stinger Reference*.

Setting a system IP address

The system IP address is the source address used for all packets generated by the system. It must be the real address of one of the unit's LAN IP interfaces, or the soft interface address (see "Defining a soft interface for increased accessibility" on page 4-8.) Following is the parameter for specifying a system address:

```
[in IP-GLOBAL]
system-ip-addr = 0.0.0.0
```

With the default zero address, the Stinger unit uses the IP address assigned to the Gigabit Ethernet interface as the source address for packets it generates. However, explicitly setting the system address simplifies access control. For example, most RADIUS servers keep a database of known remote access server (RAS) clients and their authentication keys. If you do not specify a system address, the RADIUS

database must include a complete list of all the system's interface addresses. If you specify a system address, it is used for all RADIUS request packets.

Following is an example of setting the system-ip-addr parameter to the Ethernet interface address:

```
admin> get ip-interface { { 1 8 2 } 0 } ip-address
ip-address = 2.2.2.2

admin> read ip-global
IP-GLOBAL read

admin> set system-ip-addr = 2.2.2.2

admin> write
IP-GLOBAL written
```

Configuring DNS

Domain Name System (DNS) is a TCP/IP service for centralized management of address resolution. You enable DNS lookups by specifying a domain name and the IP addresses of one or more local servers.

Some sites maintain multiple DNS servers, each one dedicated to a particular client or location. In addition, some servers support a list feature that enables them to return multiple addresses for a hostname in response to a DNS query. For information about those DNS features, see the *Stinger Reference*.

Overview of typical DNS settings

Following are the parameters (shown with default settings) for configuring DNS to allow lookups:

```
[in IP-GLOBAL]
domain-name = ""
dns-primary-server = 0.0.0.0
dns-secondary-server = 0.0.0.0
sec-domain-name = ""
```

Parameter	Setting
domain-name	Primary domain name to use for DNS lookups. The system appends this domain name to hostnames when performing lookups.
dns-primary-server	Address of the primary local DNS server to use for lookups.
dns-secondary-server	Address of the secondary local DNS server to use for lookups. Used only if the primary server is not found.
sec-domain-name	Secondary domain name to use for DNS lookups if the hostname is not found in the primary domain.

Specifying domain names for lookups

The following commands specify a primary and secondary domain name for DNS lookups:

```
admin> read ip-global
IP-GLOBAL read

admin> set domain-name = abc.com

admin> set sec-domain-name = eng.abc.com

admin> write
IP-GLOBAL written
```

If a lookup fails with the first domain name, the router tries again with the secondary domain name.

Setting RIP options

The following parameters (shown with default settings) define how the system handles RIP updates:

```
[in IP-GLOBAL]
rip-policy = Poison-Rvrs
summarize-rip-routes = no
rip-trigger = yes
rip-pref = 100
dialout-poison = no
rip-queue-depth = 0
ignore-def-route = yes
suppress-host-routes = no
```

Parameter	Setting
rip-policy	Policy for sending update packets that include routes received on the same interface.
summarize-rip-routes	Enable/disable summarization of subnet information in RIP-v1 updates. This setting has no effect on RIP-2 updates.
rip-trigger	Enable/disable RIP triggering. With a yes setting (the default), RIP updates include only changed routes.
rip-pref	Default preference for routes learned from RIP updates. When choosing the routes to put in the routing table, the unit first compares their preference values, preferring the lowest number. If the preference values are equal, the router compares the metric values, using the route with the lowest metric. Specify a number from 0 through 255. A value of 255 prevents the use of the route. The default preferences for different types of routes are 0 (zero) for connected routes, 10 for OSPF routes, 30 for routes learned from ICMP redirects, and 100 for routes learned from RIP and static routes.
dialout-poison	Enable/disable advertisement of dial-out routes when no trunks are available. Stinger units do not dial out, so leave this parameter at its default setting.
ignore-def-route	Enable/disable exclusion of advertised default routes from the routing table.

Parameter	Setting
rip-queue-depth	Maximum number of RIP packets to be held for processing. Valid values are 0 to 1024. The default (0) means that the router will not drop any RIP packets, no matter how far behind it gets.
suppress-host-routes	Enable/disable suppression of host routes for interfaces with a subnet mask of less than 32 bits.

RIP policy for propagating updates back to the originating subnet

You can specify a split-horizon or poison-reverse policy for outgoing update packets that include routes received on the same interface on which the update is sent. Split-horizon means that the router does not propagate routes back to the subnet from which they were received. Poison-reverse means that it propagates routes back to the subnet from which they were received, but with a metric of 16 (infinite metric).

The following set of commands specifies the split-horizon policy:

```
admin> read ip-global
IP-GLOBAL read

admin> set rip-policy = split

admin> write
IP-GLOBAL written
```

RIP triggering

RIP triggering enables the router to tag routes that have been updated in the routing table and send updates that include only the changed routes. The result is reduced processing overhead for both the TAOS router and its neighbors.

With the default value (yes), the router tags changes to its routing table and includes only the tagged routes in its next update. Changes occur when a call arrives or disconnects, RIP learns a route from another router, or the administrator modifies a route-related profile. The router broadcasts updates 5 to 8 seconds after the first change in the routing table is detected. The delay helps to prevent constant updates during peak traffic conditions.

If rip-trigger is set to no, the router sends full table updates every 20 to 40 seconds. To prevent RIP routers on a network from synchronizing and sending large updates in unison, the full table update is no longer broadcast at fixed 30-second intervals.

Limiting the size of UDP packet queues

When the router is very busy and receives a flood of UDP packets from SNMP requests or RIP updates, a backlog of packets waiting for processing can create enough delay in routing to cause sporadic problems with time-sensitive packets, such as LCP negotiation or frame relay management packets.

To prevent such problems, UDP processing runs at a lower priority than processing of routed packets. On a system busily routing packets, UDP processing might be delayed, and a backlog of UDP packets builds up. The rip-queue-depth parameter in the ip-global profile and the queue-depth parameter in the snmp profile specify the maximum size of this backlog.

When you set one of these parameters to specify a queue depth, the system is more likely to drop UDP packets when it is busy routing packets. However, time-sensitive routed packets are less likely to be delayed and system memory is used more efficiently.

In following sample commands sets both queue depths to 50. Fifty of each type of packet is held for processing, and if additional packets of either type are received when the queue is full, they are dropped.

```
admin> read ip-global
IP-GLOBAL read

admin> set rip-queue-depth = 50

admin> write
IP-GLOBAL written

admin> read snmp
SNMP read

admin> set queue-depth = 50

admin> write
SNMP written
```

The `netstat` command output shows the queue depth of various UDP ports, and the total packets received and total packets dropped on each port. The total packets received count includes dropped packets. In the following example, the SNMP queue depth was set to 32:

```
admin> netstat udp
udp:
Socket  Local Port  InQLen  InQMax    InQDrops    Total Rx
  0        1023       0       1           0           0
  1       route       0      50           0         509
  2        echo       0      32           0           0
  3         ntp       0      32           0           0
  4        1022       0     128           0           0
  5        SNMP      32      32        5837       20849
```

Ignoring default routes when updating the routing table

Lucent Technologies recommends enabling the `ignore-def-route` parameter to prevent routing updates from modifying the default route in the routing table. The following set of commands protects the default route from RIP updates:

```
admin> read ip-global
IP-GLOBAL read

admin> set ignore-def-route = yes

admin> write
IP-GLOBAL written
```

Suppressing host-route advertisements

If you set the `suppress-host-routes` parameter to `yes`, routes are suppressed according to the following rules:

- If a `connection` profile includes a subnet mask of less than 32 bits in the `remote-address` setting, host routes for the interface are suppressed while the session is

being negotiated, and after the session is established, only network routes are advertised for the interface.

- If a `connection` profile includes a subnet mask of /32 in the `remote-address` setting, host routes for the interface are not suppressed. (Pool addresses also have a 32-bit mask, so they are not suppressed.)

The following set of commands configures the router to suppress host routes for connections that specify a subnet mask of less than 32 bits:

```
admin> read ip-global
IP-GLOBAL read

admin> set suppress-host-routes = yes

admin> write
IP-GLOBAL written
```

Configuring and using address pools

An address pool is a range of contiguous addresses on a local IP network or subnet. Pool addresses are available for assignment to incoming connections that request an address. When the call terminates, the address is returned to the pool, making it available again for assignment.

If you designate a subnet for IP address pools, you must make sure that other IP hosts on the local network know the route to that subnet. You must also make sure that the pools do not overlap (do not contain duplicate addresses).

For related information, see "Defining address pools for a virtual router" on page 5-6.

Overview of settings for defining pools

You can define up to 128 address pools locally in the `ip-global` profile. Those pools can be used to assign addresses to connections authenticated locally (in `connection` profiles) or by RADIUS. If you are using RADIUS authentication, you can choose to define address pools in RADIUS instead of, or in addition to, those defined locally. If you have the RADIPAD program installed, you can use it to manage address pools centrally on a single RADIUS server.

Settings in the `ip-global` *profile*

The following parameters (shown with default values) configure address pools locally:

```
[in IP-GLOBAL]
pool-summary = no
pool-base-address = [ 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0+
assign-count = [ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 +
pool-name = [ "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "+
must-accept-address-assign = no
```

Parameter	Setting
`pool-summary`	Set/clear the Pool Summary flag. For details, see "Example of configuring summarized address pools" on page 4-18.

Parameter	Setting
pool-base-address	Base address of a pool of contiguous addresses on a local network or subnet.
assign-count	Number of addresses in the pool.
pool-name	A pool name, required only when TACACS+ authentication is in use. If TACACS+ authentication is not in use, the name is treated as a comment.
must-accept-address-assign	Enable/disable rejection of an assigned IP address by an incoming caller during PPP negotiation.

Settings in RADIUS pseudo-user profiles

You can define address pools in a RADIUS pools pseudo-user profile. The first line of pools pseudo-user profile uses the following format:

```
pools-name Password = "ascend", Service-Type = Outbound-User
```

The *name* argument is the system name (specified by the name parameter in the system profile). Subsequent lines in the profile define IP address pools by using the Ascend-IP-Pool-Definition (217) attribute. The value of the Ascend-IP-Pool-Definition attribute uses the following syntax:

"*pool-num base-addr assign-count*"

Syntax element	Description
pool-num	Pool number. If you use the same number to designate two pools, one locally and one in RADIUS, the RADIUS definition takes precedence. So if you have defined some pools in the ip-global profile and do not wish to override them, start numbering the pools at the next number. For example, if you defined 10 pools in the ip-global profile, start with number 11 in RADIUS. Otherwise, start with 1.
base-addr	The base address in a pool of contiguous addresses on the local network or subnet.
assign-count	Number of addresses included in the pool.

Global RADIUS pools (RADIPAD)

RADIUS IP Address Daemon (RADIPAD) is a program that works with RADIUS authentication to manage IP address pools centrally, so that connections can all acquire an address from a global pool, regardless of which system answers the call.

RADIPAD runs on one RADIUS server on the network. A Stinger unit sends an authentication request to RADIUS, and if the user profile contains an attribute to allocate an IP address from the global pool, RADIUS sends a request to RADIPAD to acquire the address.

The Stinger unit does not communicate directly with RADIPAD, so it does not require additional configuration to use RADIPAD. To configure RADIPAD, you define the global pools of addresses, specify which RADIUS server is running RADIPAD, and (optionally) specify which systems can obtain addresses from those pools. You can then create RADIUS user profiles that acquire an IP address from the global pool.

At startup, `syslog` notes RADIUS requests to release RADIUS-allocated IP addresses. Some versions of the RADIUS server might time out the request, resulting in log messages indicating the release of global-pool addresses.

Defining global pools

Global address pools are defined in a `global-pools` pseudo-user profile on the server running RADIPAD. The first line of a `global-pools` pseudo-user profile uses the following format:

```
global-pools-name Password = "ascend", Service-Type = Outbound-User
```

The *name* argument is a designation for any class of users. You can create multiple global pool profiles for multiple user classes. For example, you could create profiles named `global-pool-ppp`, `global-pool-slip`, and so forth. Subsequent lines in the profile define IP address pools by using the Ascend-IP-Pool-Definition (217) attribute. This is the same attribute described in "Settings in RADIUS pseudo-user profiles" on page 4-15, and it follows the same rules for global pools. In addition, when the Stinger unit assigns an address from a pool managed by the RADIPAD daemon, RADIPAD tries to allocate an address from the pools in order, by pool number, and chooses an address from the first pool with an available IP address.

Specifying the RADIPAD host

Each RADIUS server must specify the host running RADIPAD and (optionally) the systems that can access the global pools. These settings are defined in a `radipa-hosts` pseudo-user profile, which uses the following format in the first line of the profile:

```
radipa-hosts Password = "ascend", Service-Type = Outbound-User
```

Subsequent lines in the profile use the following attribute-value pairs to define which hosts can assign addresses from the pools managed by RADIPAD:

RADIUS attribute	Value
Ascend-Assign-IP-Client (144)	Address of a system that is allowed to access the global address pools managed by RADIPAD. You can specify multiple instances of this attribute. If no client addresses are specified, all units listed in the RADIUS clients file can access RADIPAD pools.
Ascend-Assign-IP-Server (145)	Address of the server running RADIPAD. Only one instance of this attribute can appear in the profile, and it must specify the correct IP address.

For example:

```
radipa-hosts Password ="ascend", Service-Type = Outbound-User
   Ascend-Assign-IP-Server = 10.31.4.34,
   Ascend-Assign-IP-Client = 10.31.4.10,
   Ascend-Assign-IP-Client = 10.31.4.11
```

You can specify only one RADIPAD server, but you can include multiple clients. The sample profile indicates that two systems (10.31.4.10 and 10.31.4.11) can access RADIPAD pools as clients.

Preventing the use of class boundary addresses

If you define address pools that contain more than 254 addresses, be aware that the system allocates the class boundary addresses (*n.n.n*.0 and *n.n.n*.255) as valid connection addresses. According to CIDR, this is permitted because the pool is not a /24 network. However, some client systems, such as Windows, do not tolerate the class boundary addresses well. For example, because Windows assumes a /24 network, it broadcasts NetBIOS over IP name service to the .255 address, which could overwhelm a connection assigned the .255 host address.

To prevent client software from using a host address for broadcasts, you must explicitly apply a filter that prevents the system from using the class boundary addresses. For example, if you are using RADIUS authentication, you can apply a data filter, in the `answer-defaults` profile, that drops packets from any source to pool address *n.n.n*.0 or *n.n.n*.255.

Examples of configuring address pools

For a pool that is not summarized, each assigned address is advertised as its own host route. Such a pool can start at any base address. Addresses do not accept a subnet mask component, because they are always advertised as host routes.

The following commands define three address pools, each containing 50 addresses. Pool 1 contains 10.2.3.4 through 10.2.3.54. Pool 2 contains 11.5.7.51 through 11.5.7.101. Pool 3 contains 12.7.112.15 through 12.7.112.65.

```
admin> read ip-global
IP-GLOBAL read

admin> set pool-base-address 1 = 10.2.3.4

admin> set pool-base-address 2 = 11.5.7.51

admin> set pool-base-address 3 = 12.7.112.15

admin> set assign-count 1 = 50

admin> set assign-count 2 = 50

admin> set assign-count 3 = 50

admin> write
IP-GLOBAL written
```

Following is a comparable RADIUS `pools` profile (for use by a single RADIUS server):

```
pools-taos01 Password = "ascend", Service-Type = Outbound-User
   Ascend-IP-Pool-Definition = "1 10.2.3.4 50",
   Ascend-IP-Pool-Definition = "2 11.5.7.51 50",
   Ascend-IP-Pool-Definition = "3 12.7.112.15 50"
```

Following is a comparable global pools definition (for use with RADIPAD):

```
global-pool-ppp Password ="ascend", Service-Type = Outbound-User
   Ascend-IP-Pool-Definition = "1 10.2.3.4 50",
   Ascend-IP-Pool-Definition = "2 11.5.7.51 50",
   Ascend-IP-Pool-Definition = "3 12.7.112.15 50"
```

Although some client software assumes a default subnet mask of 255.255.255.0 for PPP interfaces, you can define pools on subnets wider than /24. For example, the following commands define an address pool on a /23 subnet:

```
admin> read ip-global
IP-GLOBAL read

admin> set pool-base-address 1 = 10.55.178.1

admin> set assign-count 1 = 510

admin> write
IP-GLOBAL written
```

This pool definition translates to 10.55.178.0/23 (a subnet mask of 255.255.254.0). Following are comparable RADIUS definitions:

```
pools-taos01 Password = "ascend", Service-Type = Outbound-User
   Ascend-IP-Pool-Definition = "1 10.55.178.1 510"

global-pool-ppp Password ="ascend", Service-Type = Outbound-User
   Ascend-IP-Pool-Definition = "1 10.55.178.1 510"
```

Example of configuring summarized address pools

The pool-summary feature reduces routing overhead associated with address pools. Instead of advertising each address assigned from a pool as a host route, the system suppresses the host route advertisements and instead advertises a static route to the pool itself.

To use summarized pools locally or in RADIUS, you must set the pool-summary flag to yes in the ip-global profile, and you must define all pools to be network-aligned.

Setting the pool-summary flag

The following commands enable the pool-summary flag:

```
admin> read ip-global
IP-GLOBAL read

admin> set pool-summary = yes

admin> write
IP-GLOBAL written
```

Defining network-aligned pools

Following are the rules for network-aligned address pools:

- The specified number of addresses in the pool must be two less than the total number of addresses in the pool. (Add 2 to the assign-count value for the total number of addresses in the subnet, and calculate the mask for the subnet on the basis of this total.)

  assign-count + 2 = number of subnet hosts

- The specified base address of the pool must be the first host address. (Subtract 1 from the pool-base-address value for the base address for the subnet.)

  pool-base-address - 1 = network-aligned subnet address

The following commands enable the pool-summary flag and define a network-aligned pool:

```
admin> read ip-global
IP-GLOBAL read

admin> set pool-summary = yes
```

```
admin> set assign-count 1 = 62

admin> set pool-base-address 1 = 10.12.253.1

admin> write
IP-GLOBAL written
```

In the preceding sample configurations, the `assign-count` parameter is set to 62. When you add 2 to this value, you get 64. The subnet mask for 64 addresses is 255.255.255.192 (256 – 64 = 192). The prefix length for a 255.255.255.192 mask is /26.

The `pool-base-address` parameter is set to 10.12.253.1. When you subtract 1 from this value, you get 10.12.253.0, which is a valid network-aligned base address for the 255.255.255.192 subnet mask. (Note that 10.12.253.64, 10.12.253.128, and 10.12.253.192 are also valid zero addresses for the same mask.) The resulting address pool subnet is 10.12.253.0/26.

Following is a comparable RADIUS `pools` profile (for use by a single RADIUS server).

```
pools-taos01 Password = "ascend", Service-Type = Outbound-User
   Ascend-IP-Pool-Definition = "1 10.12.253.1 62"
```

Following is a comparable global pools definition (for use with RADIPAD):

```
global-pool-ppp Password ="ascend", Service-Type = Outbound-User
   Ascend-IP-Pool-Definition = "1 10.12.253.1 62"
```

The system still creates (but does not advertise) a host route for each assigned address in the pool. Host routes take precedence over subnet routes, so packets whose destination matches an assigned IP address from the pool are routed properly. However, because the system advertises the entire pool as a route, and only privately knows which IP addresses in the pool are active, a remote network might improperly send the Stinger unit a packet for an inactive IP address. If that occurs, the packets are routed to the Reject (`rj0`) interface (127.0.0.2). Packets routed to the Reject interface are bounced back to the sender with an `ICMP unreachable` message.

Examples of assigning an address from a pool

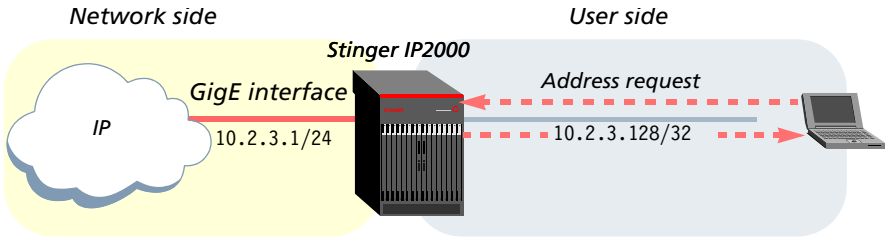
When an incoming call requests an IP address, the Stinger unit assigns one from a pool. A host requests an address if its client software has settings such as those shown in Figure 4-1:

Figure 4-1. Client software settings requesting dynamic address assignment

```
Username=victor
Accept Assigned IP=Yes
IP address=Dynamic (or Assigned or N/A)
Netmask=255.255.255.255 (or None or N/A)
Default Gateway=None or N/A
Name Server=10.2.3.55
Domain suffix=abc.com
Baud rate=38400
Hardware handshaking ON
VAN Jacobson compression ON
```

Figure 4-2 shows a remote host requesting and being assigned an IP address.

Figure 4-2. Remote CPE requiring assigned IP address



The following commands enable dynamic address assignment systemwide:

```
admin> read answer-defaults
ANSWER-DEFAULTS read

admin> set ip-answer assign-address = yes

admin> write
ANSWER-DEFAULTS written
```

During PPP negotiation, a CPE can reject an IP address offered by the router and present the caller's own IP address for consideration. For security purposes, many sites set `must-accept-address-assign` to `yes` to ensure that the Stinger unit terminates such a call, as shown in the following example:

```
admin> read ip-global
IP-GLOBAL read

admin> set must-accept-address-assign = yes

admin> write
IP-GLOBAL written
```

For address assignment to occur, the Stinger unit must have an address available for assignment, the `answer-defaults` profile must enable dynamic assignment, the client profile must specify dynamic assignment, and the client's PPP software must be configured to acquire its IP address dynamically.

The following commands configure a profile to acquire an address from the first pool that has available addresses:

```
admin> new connection victor
CONNECTION/victor read

admin> set active = yes

admin> set encapsulation-protocol = ppp

admin> set ppp recv-password = localpw

admin> set ip-options address-pool = 0

admin> write
CONNECTION/victor written
```

Following is a comparable RADIUS profile:

```
victor Password = "localpw"
   Service-Type = Framed-User,
   Framed-Protocol = PPP,
   Ascend-Assign-IP-Pool = 0
```

Following is a comparable RADIUS profile that acquires an address from any global pool managed by the RADIPAD daemon:

```
victor Password = "localpw"
   Service-Type = Framed-User,
   Framed-Protocol = PPP,
   Ascend-Assign-IP-Pool = 65535,
   Ascend-Assign-ip-global-Pool = "global-pool-ppp"
```

IP pool chaining

Because the addresses within a pool must be contiguous, many sites have defined a large number of pools, with each pool containing only a small range of addresses. For example, the following RADIUS profile defines six pools, each containing 10 addresses:

```
pools-JFAN-TNT Password = "ascend"
    Service-Type = Outbound,
    Ascend-IP-Pool-Definition = "1 11.168.6.10 10",
    Ascend-IP-Pool-Definition = "2 12.168.6.10 10",
    Ascend-IP-Pool-Definition = "3 13.168.6.10 10",
    Ascend-IP-Pool-Definition = "7 17.168.6.10 10",
    Ascend-IP-Pool-Definition = "8 18.168.6.10 10",
    Ascend-IP-Pool-Definition = "9 19.168.6.10 10"
```

In earlier versions of the software, you could allow a connection to acquire an address from any pool (by assigning the pool number 0 in the connection's profile) or from a single specified pool, such as pool 1. IP pool chaining enables you to allow a connection to acquire an address from any pool within a chain.

When IP pool chaining is enabled, contiguous pools are treated as one *pool space* with shared addresses. When the system assigns an address to an end user, it begins searching for an available address in the first pool of the chain and stops when it either finds an available address or encounters a null pool definition. So, the pools within a chain must be defined in a contiguous sequence. For example, the following profile contains two IP pool chains (pools 1, 2, 3 and pools 7, 8, 9), with each pool chain containing 30 addresses:

```
pools-JFAN-TNT Password = "ascend", Service-Type = Outbound
    Ascend-IP-Pool-Chaining = IP-Pool-Chaining-Yes,
    Ascend-IP-Pool-Definition = "1 11.168.6.10 10",
    Ascend-IP-Pool-Definition = "2 12.168.6.10 10",
    Ascend-IP-Pool-Definition = "3 13.168.6.10 10",
    Ascend-IP-Pool-Definition = "7 17.168.6.10 10",
    Ascend-IP-Pool-Definition = "8 18.168.6.10 10",
    Ascend-IP-Pool-Definition = "9 19.168.6.10 10"
```

Note To support IP pool chaining in RADIUS profiles, the RADIUS server must support vendor-specific attributes (VSA) and the system must be configured in VSA compatibility mode. For details, see "Pool chaining in RADIUS" on page 4-24.

IP pool chaining is supported both for RADIUS-defined address pools and for pools defined locally in the `ip-global` profile. For example, the following settings in the `ip-global` profile enable pool chaining and define a pool chain (pools 1 and 2) that contains 252 addresses:

```
[in IP-GLOBAL]
pool-chaining = yes
pool-base-address = [ 172.20.31.1 172.20.33.1 0.0.0.0 153.37.21.1 0.0+
assign-count = [ 126 126 0 30 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0+
```

Pool chaining in local profiles

Whether pool chains are defined locally or in RADIUS, the pool addresses are available for dynamic assignment regardless of where the connection's profile is authenticated.

Overview of local profile settings

Following are the parameters, shown with default settings, relevant to IP pool chaining:

```
[in IP-GLOBAL]
pool-chaining = no
pool-base-address = [ 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0+
assign-count = [ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0+
[in CONNECTION/"":ip-options]
address-pool = 0
```

Parameter	Setting
`pool-chaining`	Enable/disable IP pool chaining. With the `yes` setting, the system treats contiguous IP address pools as a single extended pool space when searching for an available address to assign to a connection.
`pool-base-address`	An array of up to 128 IP addresses to be used as the first address in a pool. These values are used with the `assign-count` values to define address pools locally. A pool chain contains all of the pools defined in sequence within the array, such as 1, 2, 3. To end a pool chain, leave a null value in the array.
`assign-count`	An array of up to 128 numbers that specify the number of addresses in a pool that starts with the corresponding `pool-base-address`.
`address-pool`	Number of an address pool from which to acquire an address. When pool chaining is enabled, a pool number within a chain includes addresses defined in all other pools within the chain. For example, if pools 1, 2, and 3 are in a pool chain, setting this parameter to 1 has the same effect as setting it to 2 or 3.

Example of local pool chain definition

The following commands define five address pools, which form two pool chains. Notice that the pool numbers (their indexes in the `pool-base-address` and `assign-count` arrays) are contiguous within a chain.

```
admin> read ip-global
IP-GLOBAL read
```

```
admin> set pool-chaining = yes
admin> set pool-base-address 1 = 10.1.1.1
admin> set pool-base-address 2 = 11.1.1.1
admin> set pool-base-address 3 = 12.1.1.1
admin> set assign-count 1 = 50
admin> set assign-count 2 = 50
admin> set assign-count 3 = 50
admin> set pool-base-address 7 = 13.1.1.1
admin> set pool-base-address 8 = 14.1.1.1
admin> set assign-count 7 = 50
admin> set assign-count 8 = 50
admin> write
IP-GLOBAL written
```

The following commands enable dynamic address assignment systemwide:

```
admin> read answer-defaults
ANSWER-DEFAULTS read
admin> set ip-answer assign = yes
admin> write
ANSWER-DEFAULTS written
```

The following commands configure profiles to acquire an address from the first pool chain. When the end users initiate a session request, they can acquire an address from 10.1.1.1 to 10.1.1.51, from 11.1.1.1 to 11.1.1.51, or from 12.1.1.1 to 12.1.1.51. If no addresses are available within those ranges, the connection is refused.

```
admin> new connection jfan
CONNECTION/jfan read
admin> set active = yes
admin> set encapsulation-protocol = ppp
admin> set ppp-options recv-password = localpw
admin> set ip-options address-pool = 2
admin> write
CONNECTION/jfan written
admin> new connection ravi
CONNECTION/ravi read
admin> set active = yes
admin> set encapsulation-protocol = ppp
admin> set ppp-options recv-password = localpw
admin> set ip-options address-pool = 1
admin> write
CONNECTION/ravi written
```

Following are comparable RADIUS profiles:

```
jfan Password = "localpw"
    Service-Type = Framed-User,
```

```
    Framed-Protocol = PPP,
    Ascend-Assign-IP-Pool = 2
ravi Password = "localpw"
    Service-Type = Framed-User,
    Framed-Protocol = PPP,
    Ascend-Assign-IP-Pool = 1
```

Pool chaining in RADIUS

Whether pool chains are defined locally or in a RADIUS pool's pseudo-user profile, the pool addresses are available for dynamic assignment regardless of where the connection's profile is authenticated.

Overview of RADIUS profile settings

RADIUS servers use the following attribute-value pairs to define and apply pool chains:

RADIUS attribute	Value
Ascend-IP-Pool-Chaining (85)	Enable/disable IP pool chaining in a pseudo-user profile that defines address pools. If this attribute is set to IP-Pool-Chaining-Yes (1), the system treats contiguous IP address pools as a single extended pool space when searching for an available address to assign to a connection. With a value of IP-Pool-Chaining-No (0), the system treats each address pool as a separate space. **Note** When this attribute is specified in a RADIUS profile, its value overrides the Pool-Chaining setting in the `ip-global` profile.
Ascend-IP-Pool-Definition (217)	Address pool definition in a pseudo-user profile. The value has the following syntax: *`pool-number base-addr assign-count`* The *`pool-number`* value is an integer that identifies the pool. A pool chain contains all of the pools defined in sequence, such as 1, 2, 3. To end a pool chain, leave a gap in the sequence of *`pool-number`* values. The *`base-addr`* value is an IP address to be used as the first address in a pool, and the *`assign-count`* value specifies the number of addresses in a pool.
Ascend-Assign-IP-Pool (218)	Number of the address pool from which the RADIUS user profile should acquire an address. When pool chaining is enabled, a pool number within a chain includes addresses defined in all other pools within the chain. For example, if pools 1, 2, and 3 are in a pool chain, setting this value to 1 has the same effect as setting it to 2 or 3.

To use these attributes, the RADIUS server must support vendor-specific attributes (VSAs) and the system must be configured in VSA compatibility mode. Following are the relevant settings:

```
[in EXTERNAL-AUTH]
auth-type = radius

[in EXTERNAL-AUTH:rad-auth-client]
auth-radius-compat = vendor-specific
```

For details about these settings, see the *Stinger Reference*.

Example of pool chaining in RADIUS

The following pseudo-user profile defines five address pools, which form two pool chains. Notice that the pool numbers are contiguous within a chain.

```
pools-JFAN-TNT Password = "ascend"
    Service-Type = Outbound,
    Ascend-IP-Pool-Chaining = IP-Pool-Chaining-Yes,
    Ascend-IP-Pool-Definition = "1 10.1.1.1 50",
    Ascend-IP-Pool-Definition = "2 11.1.1.1 50",
    Ascend-IP-Pool-Definition = "3 12.1.1.1 50",
    Ascend-IP-Pool-Definition = "7 13.1.1.1 50",
    Ascend-IP-Pool-Definition = "8 14.1.1.1 50"
```

The following commands configure local `connection` profiles to acquire an address from the first pool chain. When the end users initiate a session request, they can acquire an address from 13.1.1.1 to 13.1.1.51, or from 14.1.1.1 to 14.1.1.51. If no addresses are available within those ranges, the connection is refused.

```
admin> new connection hanif
CONNECTION/hanif read

admin> set active = yes

admin> set encapsulation-protocol = ppp

admin> set ppp-options recv-password = localpw

admin> set ip-options address-pool = 7

admin> write
CONNECTION/hanif written

admin> new connection hasnain
CONNECTION/hasnain read

admin> set active = yes

admin> set encapsulation-protocol = ppp

admin> set ppp-options recv-password = localpw

admin> set ip-options address-pool = 8

admin> write
CONNECTION/hasnain written
```

Following are comparable RADIUS user profiles:

```
hanif Password = "localpw"
    Service-Type = Framed-User,
    Framed-Protocol = PPP,
    Ascend-Assign-IP-Pool = 7
```

```
hasnain Password = "localpw"
    Service-Type = Framed-User,
    Framed-Protocol = PPP,
    Ascend-Assign-IP-Pool = 8
```

Configuring DHCP relay to allow CPE clients to obtain an address

RFC 951, *Bootstrap Protocol (BOOTP)*, describes an IP/UDP bootstrap protocol that allows a diskless CPE client to discover its own IP address, the address of a server host, and the name of a configuration file to be loaded into memory and executed.

RFC 2131, *The Dynamic Host Configuration Protocol (DHCP)* describes a framework for automatic configuration of IP hosts, and RFC 2132, *DHCP Options and BOOTP Vendor Information Extensions* describes BOOTP additions that can be used as DHCP options.

When the requesting client machine does not reside on the same IP network as a BOOTP or DHCP server, as is typically the case for DSL subscriber connections, the IP2000 must be configured to operate as a DHCP relay agent. A *relay agent* is an intervening system that transfers messages between the client (the requesting host system) and the server.

RFC 3046, *DHCP Relay Agent Information Option* describes information inserted by the DHCP relay agent when forwarding client-originated DHCP packets to a DHCP server. Servers recognizing the Relay Agent Information option may use the information to implement IP address or other parameter assignment policies. The DHCP Server echoes the option back verbatim to the relay agent in server-to-client replies, and the relay agent strips the option before forwarding the reply to the client.

Overview of DHCP relay configuration settings

For information about DHCP option 82, see "Configuring DHCP Option 82 for use with DHCP relay" on page 4-28. The following parameters (shown with default settings) configure DHCP relay:

```
[in IP-GLOBAL:bootp-relay]
active = no
bootp-servers = [ 0.0.0.0 0.0.0.0 ]

[in IP-GLOBAL:bootp-relay:bootp-servers]
bootp-servers[1] = 0.0.0.0
bootp-servers[2] = 0.0.0.0
```

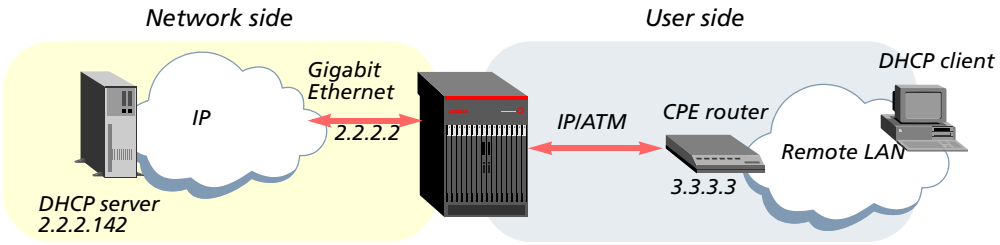
Parameter	Setting
`active`	Enable/disable DHCP relay. When this parameter is set to `yes`, the IP2000 forwards requests from a client on one network (such as a remote interface) to a DHCP server on another network interface (such as the Gigabit Ethernet interface of the IP2000).

Parameter	Setting
`bootp-servers[1]/[2]`	These indexed parameters each specify the IP address of one DHCP server. Only one address is required. If more than one server is specified, the Stinger unit uses the first server until it becomes unavailable. Once the unit starts using the second server, the unit continues using that server until it becomes unavailable, at which time the unit switches back to using the first server again.

Sample DHCP relay configuration

Figure 4-3 shows a high-speed circuit to a remote CPE router that provides a LAN interface to one or more host PCs. To support centralized assignment of the IP addresses of host PCs via DHCP, both the remote CPE router and the IP2000 must be configured as DHCP relay agents.

Figure 4-3. DHCP relay sample setup



In this example, the remote PC is configured to use DHCP to obtain an IP address. The CPE router on the remote LAN is configured statically with the IP address 3.3.3.3 and is configured to enable DHCP relay. The CPE router DHCP relay configuration specifies a DHCP server at 2.2.2.142 (the DHCP server beyond the DHCP relay agent on the IP2000).

The IP2000 is configured as DHCP relay agent with the DHCP server at 2.2.2.142 across its Gigabit Ethernet interface (2.2.2.2). The DHCP server is configured to recognize the CPE router at 3.3.3.3.

The following commands configure the IP2000 as a DHCP relay agent:

```
admin> read ip-global
IP-GLOBAL read

admin> set bootp-relay active = yes

admin> set bootp-relay bootp-servers 1 = 2.2.2.142

admin> write -f
IP-GLOBAL written
```

The next commands configure the `connection` profile for the CPE router:

```
admin> new connection cpe-router
CONNECTION/cpe-router read

admin> set active = yes

admin> set encapsulation-protocol = atm

admin> set ip-options remote-address = 3.3.3.3/30
```

```
admin> set atm-options nailed-group = 251
admin> write -f
CONNECTION/cpe-router read
```

Configuring DHCP Option 82 for use with DHCP relay

DHCP option 82, the relay agent Information option, is used to associate a unique identifier with a broadband device such as a DSL CPE or Integrated Access Device (IAD). The identifier can either be associated with the virtual circuit to the remote device, or with the remote router itself. BOOTP servers that recognize this option can use the option 82 identifier to enforce conditions on address or configuration access. For details about option 82, see RFC 3046, *DHCP Relay Agent Information Option.*

Note DHCP relay is a prerequisite for using option 82. For details, see "Configuring DHCP relay to allow CPE clients to obtain an address" on page 4-26.

DHCP option 82 configuration settings

Following are the parameters, shown with default settings, for configuring DHCP option 82.

```
[in IP-GLOBAL:bootp-relay:relay-agent-information]
circuit-id = { no 0.0.0.0 }
remote-id = { no 0.0.0.0 }
[in IP-GLOBAL:bootp-relay:relay-agent-information:circuit-id]
enable = no
if-ip = 0.0.0.0
[in IP-GLOBAL:bootp-relay:relay-agent-information:remote-id]
enable = no
if-ip = 0.0.0.0
```

Parameter	Setting
`circuit-id:enable`	Enable/disable the circuit identifier suboption of DHCP option 82. If enabled, the IP2000 encodes the `station` value (the hostname) of the `connection` or RADIUS profile that defines the PVC on which the DHCP client-to-server packet was received. This ensures that DHCP responses are sent back to the proper circuit.
`circuit-id:if-ip`	IP address of one of the IP2000 IP interfaces. If no value is specified in this field or the `if-ip` field of an enabled `remote-id` subprofile, the Stinger uses the system address (`ip-global:system-ip-addr`) if that value has been defined.
`remote-id:enable`	Enable/disable the remote identifier suboption of DHCP option 82. If enabled, the IP2000 encodes a globally unique identifier of the remote CPE from which it received a DHCP client-to-server packet, to ensure that DHCP responses are sent back to the proper remote client. The IP2000 can use this field in addition to or instead of the circuit-id field.

Parameter	Setting
remote-id:if-ip	IP address of one of the IP2000 IP interfaces. If both IDs are enabled, only one interface IP address is needed. If no address is specified in this field or in the if-ip field of an enabled circuit-id subprofile, the Stinger uses the system address (ip-global:system-ip-addr) if that value has been defined.

Sample DHCP option 82 configuration

This example builds on the sample DHCP relay configuration described in "Sample DHCP relay configuration" on page 4-27. The connection profile to the CPE router in that sample configuration does not require any changes to support option 82.

For sites that support option 82, the DHCP server configuration typically requires the presence of an ID in DHCP queries. For example, the DHCP server in this example is configured to recognize the CPE router at 3.3.3.3 across interface 7.7.7.7, and to require a circuit ID. If the DHCP request forwarded to the server by the IP2000 does not contain the circuit ID, the server refuses to return an address.

The following commands configure the IP2000 for DHCP option 82:

```
admin> read ip-global
IP-GLOBAL read

admin> list bootp-relay
[in IP-GLOBAL:bootp-relay]
active = yes
bootp-servers = [ 2.2.2.142 0.0.0.0 ]
relay-agent-information = { { no 0.0.0.0 } { no 0.0.0.0 } }
```

The next commands enable the circuit identifier suboption of DHCP option 82 and specify the Gigabit Ethernet address as the ID:

```
admin> set relay-agent-information circuit-id enable = yes

admin> set relay-agent-information circuit-id if-id = 2.2.2.2

admin> write -f
IP-GLOBAL written
```

Configuring ip-route profiles

Any profile that specifies how to reach an IP device or subnet (such as an ip-interface, connection, or RADIUS user profile) specifies a static IP route to that destination. However, you can also configure static routes explicitly, to extend or fine-tune the routing table.

Overview of typical static route settings

You can define static routes in ip-route profiles or in RADIUS. For details about RADIUS pseudo-user and user profile route definitions, using the Framed-Route (22) or Ascend-Private-Route (104) attribute-value pair, see the *TAOS RADIUS Guide and Reference*.

Following are the local parameters (shown with default settings) for configuring a static route:

```
[in IP-ROUTE/""]
name* = ""
dest-address = 0.0.0.0/0
gateway-address = 0.0.0.0
metric = 8
private-route = no
active-route = yes
```

Parameter	Settings
name	Name of the profile (up to 31 characters).
dest-address	Destination IP address. The default value is 0.0.0.0, which represents the default route.
gateway-address	IP address of a next-hop router used to reach the specified destination. A next-hop router is directly connected to the same Ethernet segment, or is one hop away on a WAN link. When the Gigabit Ethernet interface is connected to a subnet and RIP is not enabled on the interface, the system must be informed about the gateway-address of other backbone routers that can route beyond the subnet.
metric	RIP metric (0–15) for the route. Among routes with the same destination address, the higher the metric, the less likely that the system will choose the route.
private-route	Enable/disable including the route in RIP updates.
active-route	Enable/disable entering the route in the routing table. (Setting the parameter to no is a useful way to make a route temporarily inactive, so you can reinstate the route later.)

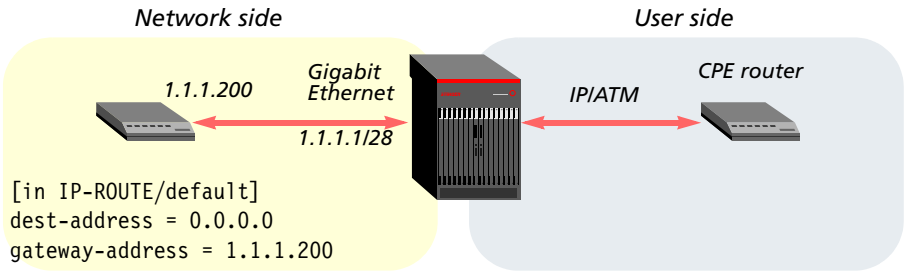
Offloading routing overhead to an external router

To offload routing overhead from the Stinger unit, you can define a default route to a router on the IP2000 Gigabit Ethernet interface. A default route is a special-case static route that acts as a catch-all for packets for which the Stinger unit cannot find a route. A default route has the zero address as its destination and points to a specific gateway address. The system routes all packets with unknown destinations to the specified gateway. If no default route is defined, the system drops those packets.

The system creates an ip-route profile named default, but the profile is not valid until you specify a gateway address, so the route is not active until you assign an address and activate the route. You can create a default route by modifying the default profile, or by creating one or more ip-route profiles that specify a zero destination and a valid gateway address.

Figure 4-4 shows a router that resides on the same subnet as the IP2000 Gigabit Ethernet IP interface. In this example, the system offloads part of its routing overhead by using a default route to the LAN router.

Figure 4-4. Default route to a local IP router



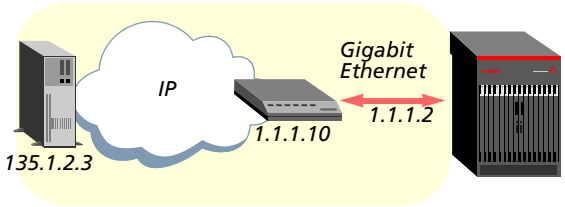
The following commands define a default route to the local router:

```
admin> read ip-route default
IP-ROUTE/default read
admin> set gateway-address = 1.1.1.200
admin> write -f
IP-ROUTE/default written
```

Creating a static route to a subnet

When RIP is turned off on an IP interface, the router cannot reach subnets beyond other routers on that interface unless it has a static route to the subnet. To enable access to subnets beyond the local segment, you must configure a static route. Figure 4-5 shows an example.

Figure 4-5. Static route to a subnet



The following commands configure a static route to the remote subnet:

```
admin> new ip-route subnet
IP-ROUTE/subnet read
admin> set dest-address = 135.1.2.3
admin> set gateway-address = 1.1.1.10
admin> write -f
IP-ROUTE/subnet written
```

Configuring IP `connection` interfaces for CPE devices

The system creates a routing interface for local `connection` profiles when it starts up. For interfaces that use pool addresses or are defined in RADIUS user profiles, the system creates a routing interface when a session becomes active.

The CPE devices described in this section are IP-capable DSL devices that transmit IP over ATM. IP over ATM is sometimes referred to as RFC 1483 traffic.

The CPE devices require a terminating PVC to the Stinger unit. A terminating PVC that is not switched through the system. It terminates on the IP2000 controller and its data stream is passed up to the IP router for further handling.

The profile for a terminating connection must specify the IP address of the far-end router, and it can set a number of other routing-related values. The profile must also specify the ATM characteristics of the connection (for example, a VPI and VCI assignment and a nailed group representing the interface to use). The *Stinger ATM Configuration Guide* describes the ATM aspects of the configuration in detail.

Typical `atm-options` settings for terminating PVCs

For a discussion of ATM settings and quality of service (QoS) contracts, see the *Stinger ATM Configuration Guide*. Following are the ATM-related parameters, shown with default settings, for ATM terminating PVCs:

```
[in CONNECTION/""]

station = ""
active = no
encapsulation-protocol = atm-circuit

[in CONNECTION/"":atm-options]
atm1483type = aal5-llc
vpi = 0
vci = 35
nailed-group = 1
```

Parameter	RADIUS attribute	Setting
`station`	User-Name (1)	Name of the far-end device.
`active`	N/A	Enable/disable the profile.
`encapsulation-protocol`	Framed-Protocol (7)	Encapsulation protocol to use for the connection. Must specify ATM for terminating PVCs.
`atm1483type`	Framed-Protocol (7)	Method of multiplexing Layer-3 packets into ATM cells. For RFC 1483 PVCs that terminate on the IP2000 to be forwarded as IP traffic, only `aal5llc` is available.
`vpi`	Ascend-ATM-Vpi (94)	VPI value for the PVC. For a discussion of valid values, see the *Stinger ATM Configuration Guide*.
`vci`	Ascend-ATM-Vci (95	VCI value for the PVC. For a discussion of valid values, see the *Stinger ATM Configuration Guide*.
`nailed-group`	Ascend-ATM-Group (64)	Nailed-group number of the interface used by the connection. You can obtain the nailed-group assigned to any interface by using the `which -n` command.

Typical `ip-options` settings for terminating PVCs

For information about enabling IP multicast forwarding on client or remote MBONE interfaces, see Chapter 8, "Forwarding Multicast Video."

Following are the IP options (shown with default settings) for configuring an IP routed RFC 1483 connection to a DSL CPE:

```
[in CONNECTION/"":ip-options]
ip-routing-enabled = yes
vj-header-prediction = no
remote-address = 0.0.0.0/0
local-address = 0.0.0.0/0
routing-metric = 1
private-route = no
rip = routing-off
```

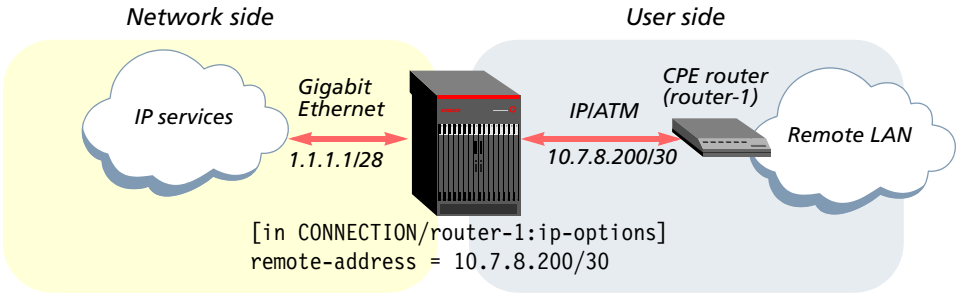
Parameter	RADIUS attribute	Setting
`ip-routing-enabled`	Ascend-Route-IP (228)	Enable/disable IP routing on the interface.
`vj-header-prediction`	Framed-Compression (13)	Enable/disable Van Jacobson prediction for TCP packets on incoming calls using encapsulation protocols that support Van Jacobson compression.
`remote-address`	Framed-IP-Address (8)	IP address of the remote CPE device.
	Framed-IP-Netmask (9)	
`local-address`	Ascend-PPP-Address (253)	Local IP address of a numbered interface connection.
	Ascend-IF-Netmask (153)	
`routing-metric`	Ascend-Metric (225)	RIP metric (1–15) for the specified route. If preference values are equal, the higher the metric, the less likely that the router will use the route.
`private-route`	Ascend-Route-Preference (126)	Include or exclude this route in RIP updates.
`rip`	Framed-Route (22)	Enable/disable RIP.

For details about parameters, see the *Stinger Reference*. For details about the attribute-value pairs used to configure IP options in RADIUS profiles, see the *TAOS RADIUS Guide and Reference*.

Sample RFC 1483 terminating PVC to a CPE router

Figure 4-6 shows a CPE router connection using IP over ATM.

Figure 4-6. Router-to-router IP connection



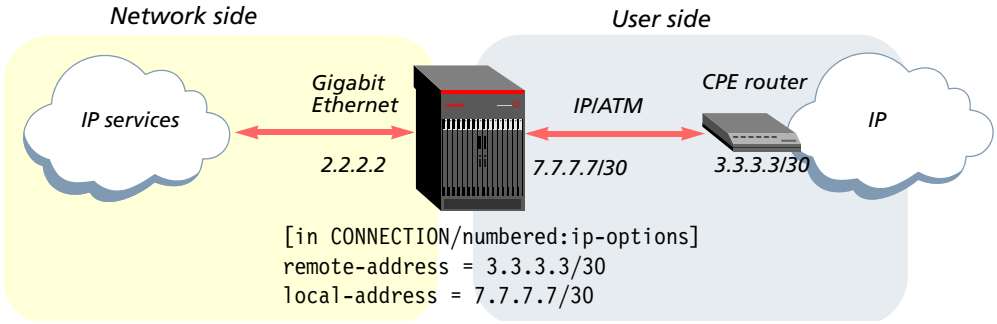
The default settings for the `ip-options` subprofile enable IP routing and Van Jacobson header compression and turn RIP off. Those settings are typically appropriate for a DSL interface, but they are not required.The following example shows configuration of a `connection` profile for the DSL CPE router in Figure 4-6:

```
admin> read connection router-1
CONNECTION/router-1 read

admin> set active = yes

admin> set encapsulation-protocol = atm

admin> set ip-options remote-address = 10.7.8.200/30

admin> set atm-options vpi = 8

admin> set atm-options vci = 100

admin> set atm-options nailed-group = 201

admin> write -f
CONNECTION/router-1 written

permconn-st-1 Password = "ascend"
    Service-Type = Outbound,
    Framed-Protocol = ATM-1483,
    User-Name = "router-1",
    Framed-IP-Address = 10.7.8.200,
    Framed-IP-Netmask = 255.255.255.252,
    Ascend-ATM-Group = 201,
    Ascend-Route-IP = Route-IP-Yes,
    Ascend-ATM-Vpi = 8,
    Ascend-ATM-Vci = 100
```

Example of a numbered interface

A numbered-interface configuration assigns each side of the connection a unique address that applies only to that connection. Figure 4-7 shows a numbered-interface connection. The Stinger unit's real, physical Ethernet interface has the IP address 2.2.2.2. The other two addresses represent the local and remote addresses of the numbered-interface connection.

Figure 4-7. A numbered-interface connection



Some applications such as SNMP use the `local-address` value internally to keep track of the circuit. The `local-address` value must be unique to the connection and to the network.

Note Do not assign a local address that belongs to one of the Stinger unit's real physical LAN interfaces. Doing so causes routing problems.

The following set of commands specifies a `connection` profile for the numbered interface:

```
admin> new connection numbered
CONNECTION/numbered read
admin> set active = yes
admin> set encapsulation-protocol = atm
admin> set ip-options remote-address = 3.3.3.3/30
admin> set ip-options local-address = 7.7.7.7/30
admin> set atm-options vci = 36
admin> set atm-options nailed-group = 211
admin> write
CONNECTION/numbered written
```

Following is a comparable RADIUS profile:

```
permconn-st-2 Password = "ascend"
    Service-Type = Framed-User,
    Framed-Protocol = ATM-1483,
    User-Name = "numbered",
    Ascend-ATM-Group = 211,
    Ascend-Route-IP = Route-IP-Yes,
    Ascend-ATM-Vpi = 0,
    Ascend-ATM-Vci = 36,
    Framed-IP-Address = 3.3.3.3,
    Framed-IP-Netmask = 255.255.255.252,
    Ascend-PPP-Addr = 7.7.7.7,
    Ascend-IF-Netmask = 255.255.255.252
```

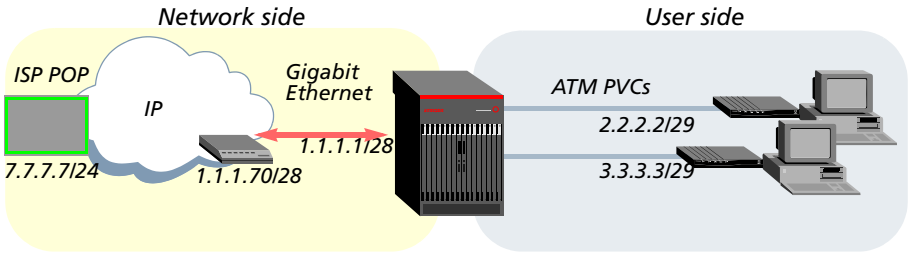
In this example, the interface is assigned a 30-bit subnet, so four bit combinations are available for host assignments. Of the four possible host addresses, the one that is evenly divisible by 4 is the network or base address (the address that specifies zeros in the host bits). This address is added to the routing table. The other host addresses are

assigned a /32 subnet mask and added as host routes. You can suppress advertisement of the host routes associated with a numbered interface by using the `suppress-host-routes` parameter, which is described in the *Stinger Reference*.

Example of forwarding IP-routed PVCs across Gigabit Ethernet

You can forward RFC 1483 PVCs from DSL subscribers onto the Gigabit Ethernet IP interface to be further routed to a specific IP destination such as an Internet service provider (ISP), as shown in Figure 4-8.

Figure 4-8. Forwarding terminating PVCs on the Gigabit Ethernet interface



This configuration requires a terminating PVC for each DSL subscriber. In this example, the Stinger does not maintain a large routing table itself. It uses a static route configuration to forward IP traffic across Gigabit Ethernet to another router, which routes the traffic on toward the ISP point of presence.

The following commands create a `connection` profile for each of the DSL subscribers in Figure 4-8:

```
admin> new connection user-1
CONNECTION/user-1 read
admin> set active = yes
admin> set encapsulation-protocol = atm
admin> set ip-options remote-address = 2.2.2.2/29
admin> which -n { 1 2 1 }
Nailed group corresponding to port { shelf-1 slot-2 1 } is 51
admin> set atm-options nailed-group = 51
admin> write -f
CONNECTION/user-1 written
admin> new connection user-2
CONNECTION/user-2 read
admin> set active = yes
admin> set encapsulation-protocol = atm
admin> set ip-options remote-address = 3.3.3.3/29
admin> which -n { 1 2 2 }
Nailed group corresponding to port { shelf-1 slot-2 2 } is 52
admin> set atm-options nailed-group = 52
admin> write -f
CONNECTION/user-2 written
```

The following command displays the ip-interface profile for the IP2000 Gigabit Ethernet interface, which shows that the address has been specified and RIP is not enabled:

```
admin> get ip-interface { { 1 8 2 } 0 }
[in IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 }]
interface-address* = { { shelf-1 first-control-module 2 } 0 }
ip-address = 1.1.1.1/28
proxy-mode = Off
rip-mode = routing-off
route-filter = ""
rip2-use-multicast = yes
ospf = { no 0.0.0.0 normal 10 40 5 simple ******* 0 1 16777215 type-1 c+
multicast-allowed = no
igmp-options = { 2 125 100 10 2 }
multicast-rate-limit = 100
multicast-group-leave-delay = 0
multicast-group-leave-delay-msec = 0
multicast-service-profile = ""
multicast-max-groups = 0
directed-broadcast-allowed = yes
vrouter = ""
management-only-interface = no
vlan-enabled = no
vlan-id = 0
```

The following set of commands configures a static route to the ISP's destination address, specifying a next-hop router on the Gigabit Ethernet interface:

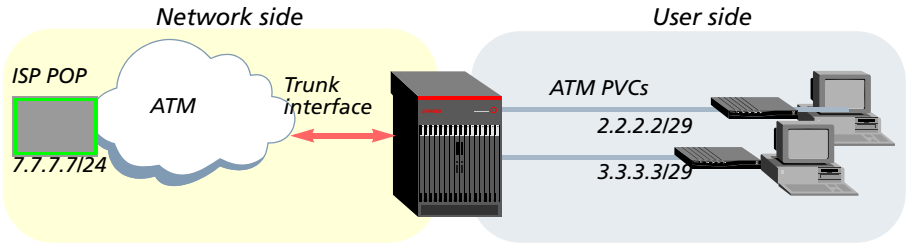
```
admin> read ip-route isp-dest
IP-ROUTE/isp-dest read

admin> set dest-address = 7.7.7.7/24

admin> set gateway-address = 1.1.1.70

admin> set active-route = yes

admin> write -f
IP-ROUTE/isp-dest written
```

With this example configuration, when packets destined for 7.7.7.7/24 are received on the terminating PVCs, the IP2000 consults its own routing table and forwards the packets onto its Gigabit Ethernet interface to the next-hop router specified as the gateway-address.

Example of using IP routing to aggregate PVCs onto a trunk VC

You can use IP routing to aggregate many RFC 1483 PVCs from DSL subscribers onto a single virtual circuit to a specific IP destination such as an ISP. Instead of configuring an ATM circuit for each subscriber, you use PVCs that terminate on the IP2000 and use IP routing to direct the traffic out on a terminating PVC to the ISP. This greatly simplifies provisioning new DSL subscribers that route to the same ISP.

Figure 4-9. Aggregating PVCs onto a single virtual circuit using IP routing



The following commands create a connection profile for each of the DSL subscribers in Figure 4-9:

```
admin> new connection user-1
CONNECTION/user-1 read
admin> set active = yes
admin> set encapsulation-protocol = atm
admin> set ip-options remote-address = 2.2.2.2/29
admin> set atm-options vpi = 0
admin> set atm-options vci = 36
admin> set atm-options nailed-group = 51
admin> write -f
CONNECTION/user-1 written
admin> new connection user-2
CONNECTION/user-2 read
admin> set active = yes
admin> set encapsulation-protocol = atm
admin> set ip-options remote-address = 3.3.3.3/29
admin> set atm-options vpi = 0
admin> set atm-options vci = 37
admin> set atm-options nailed-group = 52
admin> write -f
CONNECTION/user-2 written
```

The next command configures the PVC to the ISP:

```
admin> new connection isp
CONNECTION/isp read
admin> set active = yes
admin> set encapsulation-protocol = atm
admin> set ip-options remote-address = 7.7.7.7/24
admin> set atm-options vpi = 0
admin> set atm-options vci = 35
admin> which -n { 1 17 2}
Nailed group corresponding to port { shelf-1 trunk-module-1 2 } is 802
admin> set atm-options nailed-group = 802
admin> write -f
```

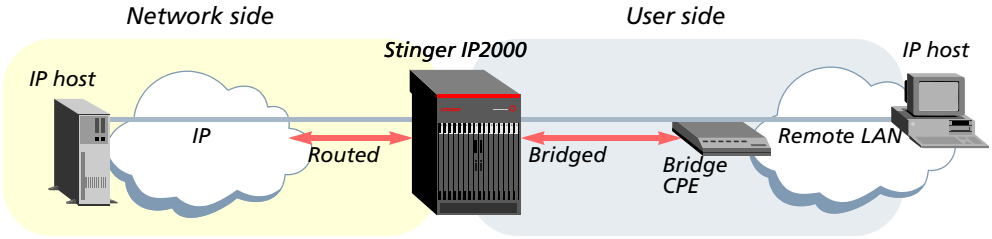
```
CONNECTION/isp written
```

This `connection` profile creates a static route to the ISP's destination address across the trunk interface. When packets destined for 7.7.7.7/24 are received on the terminating PVCs, the IP2000 consults its own routing table and forwards the packets onto the trunk interface to the ISP.

Configuring bridged IP routing (BIR) `connection` interfaces

With bridged IP routing (BIR), a Stinger IP2000 can establish an IP routed connection to an IP host through a customer premises equipment (CPE) bridge device. A BIR connection can use a line interface module (LIM) or trunk interface. A sample setup with a BIR interface on a LIM port is shown in Figure 4-10.

Figure 4-10. BIR interface on a LIM port



On the BIR interface, the system receives IP packets encapsulated in bridged frames. The IP2000 decapsulates the packets and passes them up the protocol stack to the IP router. To the IP host, the session appears to be an ordinary IP connection.

BIR configurations require the use of numbered interfaces, which assign both the remote and local side of the connection a unique IP address. The remote address can specify a subnet or an individual remote IP host. Typically, the local address for the Stinger unit is a unique address on the remote subnet. For details about numbered interfaces, see "Example of a numbered interface" on page 4-34.

Overview of `bir-options` and `ip-options` settings

In addition to the many possible IP routing parameters in `connection` and RADIUS profiles, described in "Configuring IP connection interfaces for CPE devices" on page 4-31, the following parameters apply to BIR interfaces. The parameters are shown with default settings.

```
[in CONNECTION/"":bir-options]
enable = no
proxy-arp = no
```

```
[in CONNECTION/"":ip-options]
ip-routing-enabled = yes
remote-address = 0.0.0.0/0
local-address = 0.0.0.0/0
```

Parameter	RADIUS attribute	Setting
`enable`	Ascend-BIR-Enable (70)	Enable/disable BIR on this interface.

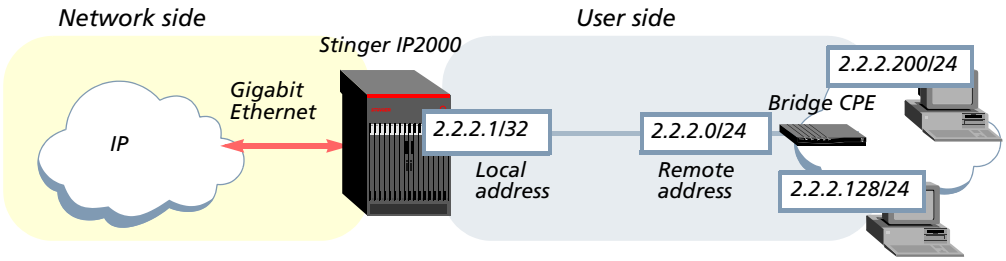
Parameter	RADIUS attribute	Setting
`proxy-arp`	Ascend-BIR-Proxy (71)	Enable/disable proxy Address Resolution Protocol (ARP), which causes the Stinger IP2000 to respond as proxy for ARP requests from local hosts for remote hosts on the far end of the link.
`ip-routing-enabled`	Ascend-Route-IP (228)	Enable/disable IP routing on the interface.
`remote-address`	Framed-IP-Address (8) Framed-IP-Netmask (9)	IP address of the remote device, which can include a subnet specification. If the address does not include a subnet mask, the router assumes the default subnet mask based on address class.
`local-address`	Ascend-PPP-Address (253) Ascend-IF-Netmask (153)	IP address assigned to the local side of a numbered-interface connection. This is a requirement for BIR interfaces.

Sample subnet (BIR/24) configuration

A BIR subnet configuration specifies a remote subnet address, and can be used to transmit bridged data to multiple IP hosts on that subnet.

When the Stinger IP2000 receives a packet destined for a BIR subnet interface, it examines the network bits of the destination address and forwards the packet to the related CPE. For example, Figure 4-11 shows two bridging CPE devices connected to an IP class C subnet. With this example, if the IP2000 receives a packet addressed to 2.2.2.200 or 2.2.2.128, it examines only the first 24 bits of the address, and forwards the packets to the bridge CPE.

Figure 4-11. BIR subnet configuration on LIM interface



The following commands configure a BIR subnet interface through the DSL CPE bridge in Figure 4-11:

```
admin> new connection bir-1
CONNECTION/bir-1 read
admin> set active = yes
admin> set encapsulation-protocol = atm
admin> set ip-options remote-address = 2.2.2.0/24
admin> set ip-options local-address = 2.2.2.1/32
admin> set bir-options enable = yes
```

```
admin> set bir-options proxy-arp = yes
admin> set atm-options atm1483type = aal5-llc
admin> set atm-options vci = 101
admin> which -n { 1 2 1 }
Nailed group corresponding to port { shelf-1 slot-2 1 } is 51
admin> set atm-options nailed-group = 51
admin write
CONNECTION/bir-1 written
```

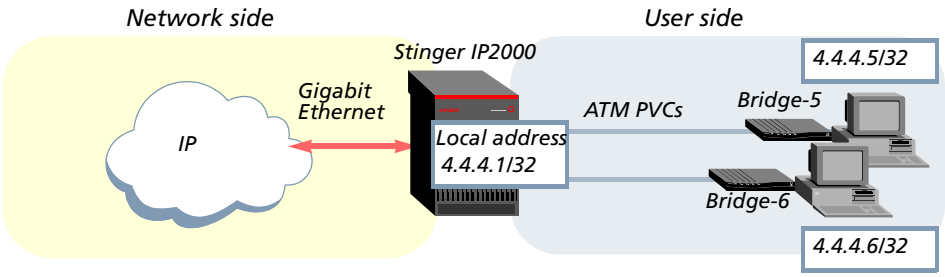
Following is a comparable definition in a RADIUS profile:

```
permconn-cpe-1 Password = "ascend"
    Service-Type = Outbound,
    Framed-Protocol = ATM-1483,
    User-Name = "bir-1",
    Ascend-Route-IP = Route-IP-Yes,
    Framed-IP-Address = 2.2.2.0,
    Framed-IP-Netmask = 255.255.255.0,
    Ascend-PPP-Addr = 2.2.2.1,
    Ascend-IF-Netmask = 255.255.255.255,
    Ascend-ATM-Group = 51,
    Ascend-ATM-Vci = 101,
    Ascend-BIR-Enable = BIR-Enable-Yes,
    Ascend-BIR-Proxy = BIR-Proxy-Yes
```

Sample host route (BIR/32) configurations

When a Stinger IP2000 receives a packet to a BIR/32 interface, it examines the full 32 bits of the destination address and forwards the packet to the related CPE. Figure 4-12 shows two bridging DSL CPE devices, each supporting one host. In this example, the IP hosts have addresses on the same IP network, but that is not a requirement.

Figure 4-12. BIR/32 configurations



In Figure 4-12, the `local-address` value is the same for both BIR interfaces. This is recommended for host routes to the same IP network because it simplifies configuration of the remote hosts, all of which can point to the same local address as the gateway.

The following commands configure a BIR/32 interface through the CPE labeled *Bridge-5* in Figure 4-12:

```
admin> new connection bir-5
CONNECTION/bir-5 read
```

```
admin> set active = yes
admin> set encapsulation-protocol = atm
admin> set ip-options remote-address = 4.4.4.5/32
admin> set ip-options local-address = 4.4.4.1/32
admin> set atm-options atm1483type = aal5-llc
admin> set atm-options vci = 111
admin> set bir-options enable = yes
admin> which -n { 1 2 5 }
Nailed group corresponding to port { shelf-1 slot-2 5 } is 55
admin> set atm-options nailed-group = 55
admin write
CONNECTION/bir-5 written
```

The following commands modify the `connection` profile immediately above to configure a BIR/32 interface through the CPE labeled *Bridge-6*:

```
admin> set station = bir-6
(New index value; will save as new profile CONNECTION/bir-6.)
admin> set ip-options remote-address = 4.4.4.6/32
admin> set atm-options vci = 112
admin> which -n { 1 2 6 }
Nailed group corresponding to port { shelf-1 slot-2 6 } is 56
admin> set atm-options nailed-group = 56
admin write
CONNECTION/bir-6 written
```

Following are comparable definitions in RADIUS profiles:

```
permconn-cpe-5 Password = "ascend"
    Service-Type = Outbound,
    Framed-Protocol = ATM-1483,
    User-Name = "bir-5",
    Ascend-Route-IP = Route-IP-Yes,
    Framed-IP-Address = 4.4.4.5,
    Framed-IP-Netmask = 255.255.255.255,
    Ascend-PPP-Addr = 4.4.4.1,
    Ascend-IF-Netmask = 255.255.255.255,
    Ascend-ATM-Group = 55,
    Ascend-ATM-Vci = 111,
    Ascend-BIR-Enable = BIR-Enable-Yes
permconn-cpe-6 Password = "ascend"
    Service-Type = Outbound,
    Framed-Protocol = ATM-1483,
    User-Name = "bir-6",
    Ascend-Route-IP = Route-IP-Yes,
    Framed-IP-Address = 4.4.4.6,
    Framed-IP-Netmask = 255.255.255.255,
    Ascend-PPP-Addr = 4.4.4.1,
    Ascend-IF-Netmask = 255.255.255.255,
    Ascend-ATM-Group = 56,
```

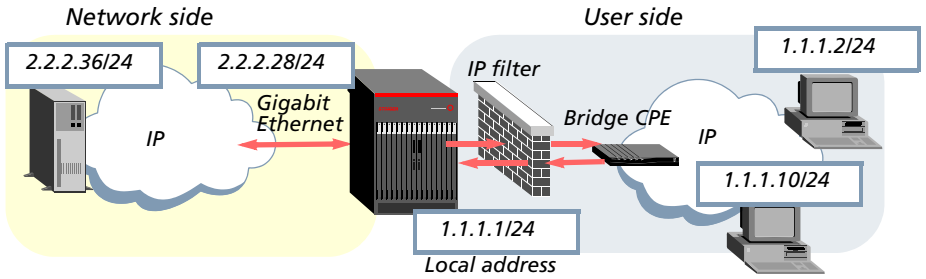
```
Ascend-ATM-Vci = 112,
Ascend-BIR-Enable = BIR-Enable-Yes
```

Sample use of filters with BIR connections

You can apply an IP filter to restrict outbound packets on a BIR interface. However, IP filters are not applied to the inbound packet stream on a BIR interface. For details about defining IP filters, see "Using IP Filters" on page 10-1.

Figure 4-13 shows a sample BIR interface to a subnet that supports two IP hosts.

Figure 4-13. Bidirectional filtering on a BIR interface



The filter defined in this example is applied to the BIR interface. The input filter rules affect packets received on the BIR interface, and output filter rules affect packets destined for the user-side subnet.

The first input filter rule shown below specifies that if the destination IP address in a packet is 2.2.2.0/24, the protocol is 17 (UDP), and the source UDP port is less than 50, the packet is discarded. So, packets that match this rule will not reach the server at 2.2.2.36. The second input filter is an explicit default rule that forwards all other IP packets received on the BIR interface.

```
admin> new filter udp-filter
FILTER/udp-filter read
admin> set input-filters 1 valid-entry = yes
admin> set input-filters 1 forward = no
admin> set input-filters 1 Type = ip-filter
admin> set input-filters 1 ip-filter protocol = 17
admin> set input-filters 1 ip-filter dest-address-mask = 255.255.255.0
admin> set input-filters 1 ip-filter dest-address = 2.2.2.36
admin> set input-filters 1 ip-filter Src-Port-Cmp = less
admin> set input-filters 1 ip-filter source-port = 50
admin> set input-filters 2 valid-entry = yes
admin> set input-filters 2 forward = yes
admin> set input-filters 2 Type = ip-filter
```

The first output filter rule shown below specifies that if the source IP address in a packet is 2.2.2.36/24, the protocol is 17, and the source UDP port is less than 50, the packet is discarded. So, packets that match this rule will not reach the IP hosts across the BIR interface. The second output filter is an explicit default rule that forwards all other IP packets destined for the remote subnet through the BIR interface.

```
admin> set output-filters 1 valid-entry = yes
admin> set output-filters 1 forward = no
admin> set output-filters 1 Type = ip-filter
admin> set output-filters 1 ip-filter source-address-mask = 255.255.255.0
admin> set output-filters 1 ip-filter source-address = 2.2.2.36
admin> set output-filters 1 ip-filter protocol = 17
admin> set output-filters 1 ip-filter Src-Port-Cmp = less
admin> set output-filters 1 ip-filter source-port = 50
admin> set output-filters 2 valid-entry = yes
admin> set output-filters 2 forward = yes
admin> set output-filters 2 Type = ip-filter
admin> write -f
FILTER/udp-filter written
```

The following commands create a BIR profile to the bridge CPE in Figure 4-13, and apply the sample filter:

```
admin> new connection bir-1-1
CONNECTION/bir-1-1 read
admin> set active = yes
admin> set encapsulation-protocol = atm
admin> set ip-options remote-address = 1.1.1.0/24
admin> set ip-options local-address = 1.1.1.1/24
admin> set session-options data-filter = udp-filter
admin> set bir-options enable = yes
admin> set atm-options nailed-group = 101
admin> write -f
CONNECTION/bir-1-1 written
```

Administrative tools for IP routing

The system supports several commands that are useful for locating the sources of problems on an IP network and for communicating with other hosts for management purposes. For examples, see the chapter on working with IP traffic in the *Stinger Adminstration Guide*, and entries in the *Stinger Reference* on commands such as the following:

- arptable
- ipcache
- iproute
- netstat
- nslookup
- ping
- telnet
- traceroute

Virtual Routing

5

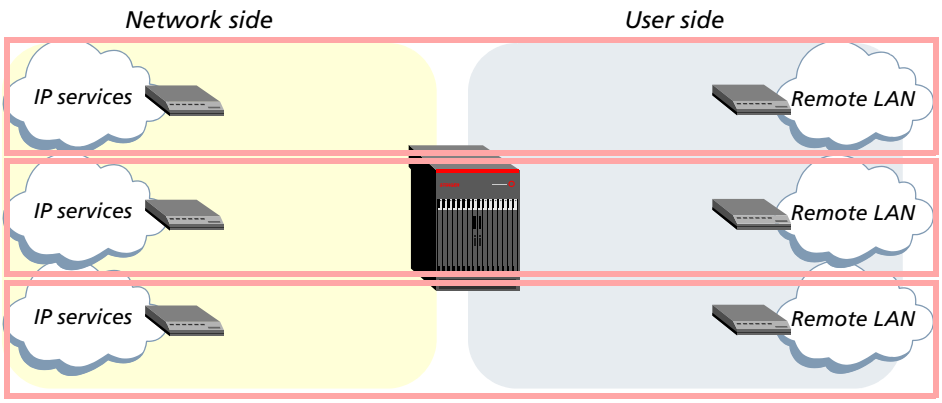
Virtual routing enables you to partition virtual domains within the global IP router. Each virtual domain is defined by a named virtual router. Currently, the Stinger IP2000 controller supports up to 1022 virtual routers in addition to the global router.

If you do not configure virtual routers, the global IP router operates as documented in Chapter 4, "IP Routing." All interfaces that are not explicitly grouped with a defined virtual router are grouped with the global router.

Overview of virtual routing

Virtual routing enables high-density circuit termination with secure logical partitioning and multiple route tables. Virtual routing is particularly useful for remote access server (RAS) functionality. For example, it can securely partition traffic from many CPE devices to different ISPs, with each ISP mapped to a separate virtual domain. Figure 5-1 shows a simplified diagram with three virtual routers configured in the global router. Because each virtual router has its own routing table, traffic within a virtual domain is typically not shared with other domains.

Figure 5-1. Simple diagram of three virtual domains (virtual routers)



How virtual routers affect the routing table

When virtual routers are not defined, the global router maintains a single IP routing table that enables the router to reach any of its many interfaces. In that context, each interface known to the system requires a unique address.

With virtual routers, addresses must be unique within the virtual domain, but not necessarily within the Stinger IP2000. Because each virtual router maintains its own routing table, and because it knows about only those interfaces that explicitly specify the same virtual router, there is no requirement that the private networks maintain unique address spaces, as long as the virtual domains are not interconnected.

Interconnecting virtual domains

Each virtual router has its own associated routing table, ARP table, route cache, and address pools, which cannot be shared with another virtual router. However, it is possible to interconnect two virtual domains by defining an inter-virtual-router route. For details, see "Specifying an inter-virtual-router route" on page 5-9.

Note Since routing traffic between virtual domains is not fast routed, it is strongly recommended to minimize such traffic.

Applicability and limitations

When configuring virtual routing on a Stinger IP2000, consider the following issues related to virtual router applicability and limitations:

- Virtual routing does not apply to switched or bridged connections.
- Only terminated virtual circuits (IP, PPPoA, PPPoE) can be integrated into a virtual domain.
- When the virtual LAN (VLAN) feature is used in its usual bridging capacity, to form a bridge between a VLAN ID on Gigabit Ethernet and a DSL interface, virtual routing does not apply. If the VLAN is not used in a bridge capacity but acts as an IP interface, virtual routing applies.
- Bridged IP routing (BIR) profiles have no special considerations for virtual routing. The `connection` profile terminating the PVC must define the virtual router.
- With the current implementation, multicasting applications must use the global router.
- If multiple virtual IP interfaces are configured on a physical interface, particularly an Ethernet interface, all virtual interfaces must be attached to the same virtual router.
- SNMP management utilities do not currently display information on a per-virtual-router basis.
- Errors and events are not logged on a per-virtual-router basis.
- The `syslog` host defined in the system's `log` profile must be accessible to the global router.
- Servers defined in the `debug`, `trap`, `external-auth`, `ip-global` (for SNTP and multicast), `call-logging`, and `snmp` profiles must be accessible to the global router.

Creating a virtual router

When at least one `vrouter` profile is configured, the `system-ip-address` parameter and the `global-vrouter` parameter in the `ip-global` profile apply to the global router. All interfaces that are not explicitly assigned to another virtual router are grouped with the global router.

For each virtual router in the system, an instance of RIP is created to process routes. The new instance of RIP sends and receives update packets only on the interfaces associated with its particular virtual router and manipulates only that virtual router's routing table. A default instance of RIP is always created for the global router.

When you create a virtual router, the new instance of RIP sends and receives packets only on the interfaces associated with that virtual router and manipulates only that virtual router's routing table. All RIP-related parameters in a `vrouter` profile use default settings that are recommended for most sites.

Overview of `vrouter` profile settings

A `vrouter` profile contains the following parameters, shown with default values:

```
[in VROUTER/"" ]
name* = ""
active = yes
vrouter-ip-addr = 0.0.0.0
pool-base-address = [ 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0+
assign-count = [ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 +
pool-name = [ "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "+
pool-summary = no
share-global-pool = yes
rip-policy = Poison-Rvrs
summarize-rip-routes = no
rip-trigger = yes
domain-name = ""
sec-domain-name = ""
dns-primary-server = 0.0.0.0
dns-secondary-server = 0.0.0.0
client-primary-dns-server = 0.0.0.0
client-secondary-dns-server = 0.0.0.0
allow-as-client-dns-info = True
```

Parameter	Setting
`name`	Unique name for the virtual router, up to 15 characters. All interfaces belonging to a virtual router specify the same virtual router name in the `ip-interface` or `connection` profile.
`active`	Activate the virtual router.
`vrouter-ip-address`	System IP address for the virtual router.

Parameter	Setting
`pool-base-address`	Base address of a pool of contiguous addresses on a local network or subnet. The pool will be exclusively for use by the virtual router. For details about defining address pools, see "Configuring and using address pools" on page 4-14.
`assign-count`	Number of addresses in the pool. The pool will be exclusively for use by the virtual router. For details about defining address pools, see "Configuring and using address pools" on page 4-14.
`pool-name`	A pool name, required only when TACACS+ authentication is in use. The pool will be exclusively for use by the virtual router. For details about defining address pools, see "Configuring and using address pools" on page 4-14.
`pool-summary`	Set/clear the pool summary flag to specify that the address pools will be summarized. For details about defining address pools that can be summarized, see "Configuring and using address pools" on page 4-14.
`share-global-pool`	Enable/disable the virtual router to share the address pools defined in the `ip-global` profile.
`rip-policy`	Policy for the virtual router to use when sending update packets that include routes received on the same interface. For details, see "Setting RIP options" on page 4-11.
`summarize-rip-routes`	Whether the virtual router summarizes subnet information in RIP-v1 advertisements. For details about this feature, see "Setting RIP options" on page 4-11.
`rip-trigger`	Enable/disable RIP triggering for the virtual router. For details about RIP triggering, see "Setting RIP options" on page 4-11.

Note For details about `domain-name` and other DNS parameters, see "Configuring virtual router DNS servers" on page 5-10.

Example of defining a virtual router

The following commands create a virtual router named `vr1` with a system address of 130.200.200.100:

```
admin> new vrouter vr1
VROUTER/vr1 read

admin> set vrouter-ip-addr = 130.200.200.100

admin> write -f
VROUTER/vr1 written
```

The `vr1` virtual router maintains minimal routing and interface tables at this point, as shown in the following sample output:

```
admin> netstat vr1 -rn
Destination    Gateway  IF          Flg     Pref    Met     Use     Age
127.0.0.0/8    -        bh0_vr1     CP      0       0       1       6815
127.0.0.1/32   -        local       CP      0       0       1       6815
127.0.0.2/32   -        rj0_vr1     CP      0       0       1       6815
224.0.0.9/32   -        local       CP      0       0       1       6815
Total Routes = 4    Hidden Routes = 0

admin> netstat vr1 -in
Name          MTU     Net/Dest       Address    Ipkts  Ierr Opkts     Oerr
vr0_vr1       1500    127.0.0.4/32   127.0.0.4  0      0    0         0
lo0_vr1       1500    127.0.0.1/32   127.0.0.1  0      0    0         0
local         65535   127.0.0.1/32   127.0.0.1  0      0    0         0
rj0_vr1       1500    127.0.0.2/32   127.0.0.2  0      0    0         0
bh0_vr1       1500    127.0.0.3/32   127.0.0.3  0      0    0         0
```

The virtual router also maintains its own IP, TCP, UDP, and ICMP statistics. For example:

```
admin> netstat vr1 -s
udp:
       1442 packets received
       0 packets received with no ports
       0 packets received with errors
       0 packets dropped
       32 packets transmitted
tcp:
       0 active opens
       1 passive opens
       0 connect attempts failed
       0 connections were reset
       1 connections currently established
       858 segments received
       0 segments received out of order
       548 segments transmitted
       0 segments retransmitted
       0 active closes
       0 passive closes
       0 disconnects while awaiting retransmission
icmp:
       31 packets received
       0 packets received with errors
       Input histogram:
              30 echo requests
              1 netmask requests

       31 packets transmitted
       0 packets not transmitted due to lack of resources
       Output histogram:
```

```
                30 echo replies
                1 netmask replies

ip:
        0 packets received
        0 packets received with header errors
        0 packets received with address errors
        0 packets received forwarded
        0 packets received with unknown protocols
        0 inbound packets discarded
        0 packets delivered to upper layers
        0 transmit requests
        0 discarded transmit packets
        0 outbound packets with no route
        0 reassemblies timeout
        0 reassemblies required
        0 reassemblies succeeded
        0 reassemblies failed
        0 fragmentation succeeded
        0 fragmentation failed
        0 fragmented packets created
        0 route discards due to lack of memory
        64 default ttl
igmp:
        0 packets received
        0 bad checksum packets received
        0 bad version packets received
        0 query packets received
        0 leave packets received
        0 packets transmitted
        0 query packets sent
        0 resonse packets sent
        0 leave packets sent
mcast:
        0 packets received
        0 packets forwarded
        0 packets in error
        0 packets dropped
        0 packets transmitted
pim:
        0 packets received
        559 packets transmitted
        559 hello packets sent
```

Note Multicast is not currently supported on a per-virtual-router basis, so the IGMP, multicast, and PIM statistics relate only to the global router.

Defining address pools for a virtual router

The following commands define an address pool for the `vr1` virtual router defined in "Example of defining a virtual router" on page 5-4:

```
admin> read vrouter vr1
```

```
VROUTER/vr1 read
admin> set pool-base 1 = 130.100.100.128
admin> set assign-count 1 = 127
admin> write -f
VROUTER/vr1 written
```

Following is a comparable RADIUS pool definition:

```
pools-taos01 Password = "ascend", Service-Type = Outbound-User
    Ascend-IP-Pool-Definition = "1 130.100.100.128 127 vr1"
```

The vr1 virtual router is now maintaining the following pool of addresses:

```
admin> ip-pools vr1
Pool#                Base         Count      InUse
   1          130.100.100.128      127          0
Number of remaining allocated addresses:    0
```

Note The Ascend-IP-Pool-Definition attribute supports a virtual router name as the last syntax element in a pool definition. The value of Ascend-IP-Pool-Definition uses the following syntax:

"*pool-num base-addr assign-count* [*vrouter-name*]"

For background information about address pools, see "Configuring and using address pools" on page 4-14. The process of defining address pools for a virtual router is the same as described in that section.

Assigning interfaces to a virtual router

To assign virtual router membership to an interface, you specify a virtual router name in the interface profile. For a virtual router to be active, at least one IP interface (LAN or WAN) must specify its name.

Overview of interface **vrouter** settings

To assign virtual router membership to an interface in local profiles, set the vrouter parameter. For example:

```
[in IP-INTERFACE/{ { shelf-1 slot-8 2 } 0 } ]
vrouter = vr1
[in CONNECTION/vr1-client]
vrouter = vr1
```

Parameter	RADIUS attribute	Setting
vrouter	Ascend-VRouter-Name (102)	Name of a defined virtual router. Specifying the virtual router name groups the interface with the virtual router. The default null value specifies the global router.

Examples of assigning virtual router membership to interfaces

The following commands assigns a WAN interface to the vr1 virtual router:

```
admin> read connection router-1
CONNECTION/router-1 read
```

```
admin> set active = yes
admin> set encapsulation-protocol = atm
admin> set vrouter = vr1
admin> set ip-options remote-address = 10.7.8.200/30
admin> set atm-options vci = 100
admin> set atm-options nailed-group = 201
admin> write -f
CONNECTION/router-1 written
```

Following is a comparable RADIUS profile:

```
permconn-st-1 Password = "ascend"
    Service-Type = Outbound,
    Framed-Protocol = ATM-1483,
    User-Name = "router-1",
    Framed-IP-Address = 10.7.8.200,
    Framed-IP-Netmask = 255.255.255.252,
    Ascend-ATM-Group = 201,
    Ascend-Route-IP = Route-IP-Yes,
    Ascend-ATM-Vpi = 8,
    Ascend-ATM-Vci = 100,
    Ascend-Vrouter-Name = "vr1"
```

Defining virtual router static routes

You specify a static route associated with a virtual router for one of the following reasons:

- To define a route on a per-virtual-router basis
- To specify an inter-virtual-router route

Overview of static route settings

Following are the virtual router static route parameters (shown here with default values) in `ip-route` profiles:

```
[in IP-ROUTE/""]
vrouter = ""
inter-vrouter = ""
```

Parameter	Setting
`vrouter`	Name of the virtual router that will own this route. The route will be part of the specified virtual router's routing table. If no name is specified (the default), the global router is assumed.
`inter-vrouter`	Name of a virtual router to use as the route's next hop. All packets to the static route's destination network are sent to the specified virtual router for a routing decision. The `gateway-address` parameter must be set to the zero address for this parameter to apply.

In a RADIUS profile, the value of the Framed-Route (22) attribute can specify a virtual router name in the following syntax:

"*dest-addr* [/*prefix*] *gateway-addr metric* [private] [*profile*] [*preference*] [*vrouter-name*]"

Note The fields within the value of the Framed-Route attribute are positional. With the exception of the optional prefix-length specification, if any of the optional fields are specified, the optional fields to the left of that setting must also be specified.

Examples of defining a route on a per-virtual-router basis

When you define a route on a per-virtual-router basis, it appears only in the specified virtual router's routing tables. That virtual router "owns" the route.

Following is an example of defining a static route within the vr1 virtual router domain. This route will appear only in the routing table for vr1.

```
admin> new ip-route rtr1
IP-ROUTE/rtr1 read

admin> set dest = 10.5.6.7/28

admin> set gateway = 10.1.1.1

admin> set vrouter = vr1

admin> write -f
IP-ROUTE/rtr1 written
```

Following is a comparable RADIUS profile:

```
route-taos-1 Password = "ascend", Service-Type = Outbound-User
    Framed-Route = "10.5.6.7/28 10.1.1.1 7 n rtr1 60 vr1"
```

The following sample output shows the new static route that was added to the vr1 virtual router's routing table:

```
admin> netstat vr1 -rn
Destination     Gateway      IF            Flg   Pref Met    Use        Age
10.1.1.0/24     10.1.1.1     wan30         SG     120   7      0          9
10.1.1.1/32     10.1.1.1     wan30         S      120   7      2          9
10.5.6.0/28     10.1.1.1     wan30         SG      60   8      0          9
11.1.1.0/24     11.1.1.1     wan31         SG     120   7      0          9
11.1.1.1/32     11.1.1.1     wan31         S      120   7      1          9
12.1.1.0/24     12.1.1.1     wan32         SG     120   7      0          9
12.1.1.1/32     12.1.1.1     wan32         S      120   7      1          9
127.0.0.0/8     -            bh0_vr1       CP       0   0      0       2274
127.0.0.1/32    -            local         CP       0   0      0       2274
127.0.0.2/32    -            rj0_vr1       CP       0   0      0       2274
```

Specifying an inter-virtual-router route

You can cause one virtual router or the global router to forward traffic to another virtual router for a routing decision by specifying an inter-virtual-router static route.

Note Because routing traffic between virtual domains is not fast routed, it is strongly recommended to minimize such traffic.

In the following example, the static route specifies the `vr1` virtual router as the route's next hop. This route is not defined on a per-virtual-router basis, so it is owned by the global router.

```
admin> new ip-route rtr2
IP-ROUTE/rtr2 read

admin> set dest-address = 11.0.0.0/24

admin> set inter-vrouter = vr1

admin> write -f
IP-ROUTE/rtr2 written
```

Following is a comparable RADIUS route profile:

```
route-taos-1 Password = "ascend", Service-Type = Outbound-User
    Framed-Route = "11.0.0.0/28 0.0.0.0 vr1"
```

The following output shows that the route has been added to the global router's routing table:

```
admin> netstat -rn
Destination      Gateway     IF           Flg  Pref Met    Use     Age
0.0.0.0/0        10.1.6.1    ie0          SGP   60   1      59       4
11.0.0.0/24       -          vr0_vr1       S    60   8       0       4
20.0.0.0/8        -          ie1-12-1      C     0   0      12     234
20.1.1.2/32       -          local        CP     0   0       0    2347
127.0.0.0/8       -          bh0          CP     0   0       0    2378
127.0.0.1/32      -          local        CP     0   0       0    2378
127.0.0.2/32      -          rj0          CP     0   0       0    2378
130.1.1.1/32      -          sip0          C     0   0       0    2378
130.1.1.252/30    -          rj0           C     0   0       0    2378
100.1.6.0/24     100.1.6.221 wanabe       SG    60   1       0       4
101.1.6.0/24      -          ie0           C     0   0    2531    2378
101.1.6.234/32    -          local        CP     0   0    4152    2378
224.0.0.0/4       -          mcast        CP     0   0       0    2378
224.0.0.1/32      -          local        CP     0   0       0    2378
224.0.0.2/32      -          local        CP     0   0       0    2378
224.0.0.5/32      -          local        CP     0   0     732    2378
224.0.0.6/32      -          local        CP     0   0       0    2378
255.255.255.255/32 -         ie0           P     0   0     422    2378
```

Configuring virtual router DNS servers

Virtual router DNS configuration includes settings for primary and secondary DNS servers, domain names, and client DNS servers. The settings direct connections that belong to the virtual router to a particular DNS service. To completely segment the virtual router's DNS information from any other hosts, you can configure and manage DNS information separately for each virtual router. The addresses configured for client DNS servers are presented to dial-in users during IP Control Protocol (IPCP) negotiation.

If DNS information is not found in the `vrouter` profile, the system uses the DNS information in the `ip-global` profile. The DNS list and the local DNS table maintained in RAM are systemwide DNS configurations that are not supported separately for each virtual router.

Overview of virtual router DNS settings

Following are the virtual router-specific DNS parameters (shown with their default settings):

```
[in VROUTER/""]
domain-name = ""
sec-domain-name = ""
dns-primary-server = 0.0.0.0
dns-secondary-server = 0.0.0.0
client-primary-dns-server = 0.0.0.0
client-secondary-dns-server = 0.0.0.0
allow-as-client-dns-info = True
```

Parameter	Setting
`domain-name`	Primary domain name (up to 63 characters) to use for DNS lookups for this virtual router. The system appends this domain name to hostnames when performing lookups.
`sec-domain-name`	Secondary domain name to use for DNS lookups for this virtual router if the hostname is not found in the primary domain.
`dns-primary-server`	Address of the primary local DNS server to use for lookups for this virtual router.
`dns-secondary-server`	Address of the secondary local DNS server to use for lookups for this virtual router. Used only if the primary server is not found.
`client-dns-primary-server`	Address of a client DNS server for dial-in clients of this virtual router.
`client-dns-secondary-server`	Address of a secondary DNS server for dial-in clients of this virtual router.
`allow-as-client-dns-info`	Enable/disable use of local DNS information if the client DNS servers are not found. To isolate local network information for this virtual router, set to `false`.

Example of a typical virtual router DNS configuration

The following commands specify a primary and secondary domain name for DNS lookups for a virtual router named `xyz`:

```
admin> read vrouter xyz
VROUTER/xyz read

admin> set domain-name = xyz.com

admin> set sec-domain-name = eng.xyz.com

admin> write -f
VROUTER/xyz written
```

If a lookup fails in the first domain, the router tries again with the secondary domain name. To enable the system to use DNS to look up addresses, specify DNS server addresses, as shown in the following example:

```
admin> read vrouter xyz
VROUTER/xyz read

admin> set dns-primary-server = 1.2.2.2

admin> set dns-secondary-server = 1.3.3.3

admin> write -f
VROUTER/xyz written
```

If the primary server is unavailable, the system attempts a lookup on the secondary server. The following commands configure a client DNS server for this virtual router:

```
admin> read vrouter xyz
VROUTER/xyz read

admin> set client-dns-primary-server = 1.2.2.2

admin> set client-dns-secondary-server = 1.2.2.96

admin> set allow-as-client-dns-info = false

admin> write -f
VROUTER/xyz written
```

The secondary server is accessed only if the primary one is inaccessible. If both of these client DNS servers are not accessible, the system does not allow the client to access local DNS servers.

Deleting a virtual router

You can delete a virtual router only if no more interfaces are attached to it. If one interface is attached to a virtual router, the system prevents its deletion. To delete a virtual router that has no attached interfaces, delete the `vrouter` profile. For example:

```
admin> delete vrouter vr1
```

Lucent Technologies recommends that you reset the system after deleting a virtual router with active connections. If a system reset is not possible, the recommended course of action before deleting the virtual router is to manually tear down its active connections, and then modify the local `connection`, `ip-interface`, and `ip-route` profiles that point to the virtual router to point instead to the global router or another existing virtual router.

Administrative tools for virtual routers

You can specify a virtual router name on the command line of the network administration commands listed in Table 5-1 to obtain information specific to a particular virtual domain.

Table 5-1. Administrative commands showing optional vrouter arguments

Command	Permissions	Usage with optional vrouter argument
arptable	system	arptable [*vrouter*] [[-a *hostname MAC_address*] | [-d *hostname*] | [-f]]
ipcache	system	ipcache [-r *vrouter*] [cache] [stats]
iproute	system	iproute add [-r *vrouter*] *dest_IPaddress/subnet_mask gateway_IPaddress* [*preference*] [metric] iproute delete [-r *vrouter*] *dest_IPaddress/subnet_mask* [gateway]
netstat	system	netstat [*vrouter*] [-i] [-r] [?] [-n] [-d] [-s *identifiers*] [-z]
vrouter	system	vrouter [dump [full]] [callback]
nslookup	diagnostic	nslookup [-v] [-s *dnssrvr_IPaddress*] [-r *vrouter*] *hostname*
ping	diagnostic	ping [-q | -v] [-i *delay_sec* | -I *delay_msec*] [-s *packetsize*] [-r *vrouter*] [-x *source_IPaddress*] *hostname*
telnet	diagnostic	telnet [-a | -b | -t] [-v *vrouter*] [-l[e] | -r[e]] *hostname* [*portnumber*]
traceroute	diagnostic	traceroute [-n] [-v] [-m *max_ttl*] [-p *port*] [-q *nqueries*] [-w *waittime*] [-r *vrouter*] [-s src_IPaddr] *hostname* [*datasize*]
agrm	debug	agrm -rt [*vrouter*] [-f] agrm -arp [*vrouter*]
ifmgr	debug	ifmgr [-r *vrouter*] -d [*ifNum*] -t ifmgr [up|down] [ifNum|ifName]

For details about the system or diagnostic commands in Table 5-1, see the *Stinger Reference*. For information about the debug-level commands, see Appendix A, "IP2000 Diagnostics.".

OSPF Routing

6

Open Shortest Path First (OSPF) is an Internet routing protocol, developed by the OSPF working group of the Internet Engineering Task Force, and defined in RFC 2328, *OSPF Version 2*.

OSPF was designed for the TCP/IP Internet environment, including explicit support for IP subnets, tagging of externally derived routing information, and use of IP multicast for sending or receiving link-state updates.

This implementation of OSPF conforms to RFC 2328 specifications, but does not support the following features:

- Stinger IP2000 systems do not currently function as an interior gateway protocol (IGP) gateway.
- OSPF routing is not supported on virtual IP interfaces.

Overview of OSPF features supported by the IP2000

This section provides a brief overview of OSPF routing to help you configure the system properly. For details about how OSPF works, see RFC 2328, *OSPF Version 2*. Following are the OSPF functions discussed in this section:

- Multiple authentication schemes, per RFC 1583
- Variable length subnet masks (VLSMs)
- Link state advertisement (LSA) types 1, 2, 3, 4, 5, and 7
- Backup and designated backup router capability on broadcast networks
- Nonbroadcast multiaccess (NBMA) network support over point-to-point links
- Configurable cost metrics

- Hierarchical routing via normal areas, stub areas, and not-so-stubby-areas (NSSAs)
- Shortest path first link-state routing algorithm
- Diagnostics and traps

Authentication

All OSPF protocol exchanges are authenticated by simple authentication by default. Only trusted routers can participate in the autonomous system's routing. A variety of authentication schemes can be used. In fact, different authentication types can be configured for each area. For a discussion of areas, see "Hierarchical routing (areas)" on page 6-5.

Authentication provides added security for the routers that are on the network. Routers that do not have the password are unable to gain access to the routing information, because authentication failure prevents a router from forming adjacencies. (For a discussion of adjacencies, see "Exchange of routing information" on page 6-2.) If both sides of a connection do not support the same authentication method, packet error messages can result.

In addition to null and simple password authentication, Stinger units support the MD5 cryptographic authentication method for OSPF as described RFC 2328. For details about MD5 encryption, see RFC 2328.

Support for variable-length subnet masks

OSPF routers handle variable-length subnet masks (VLSMs). Each route distributed by OSPF has a destination address and subnet mask, and two different subnets of the same IP network can use different size subnet masks. A packet is routed to the best (longest or most specific) match. Host routes are considered to be subnets whose masks are all ones (0xFFFFFFFF).

Note OSPF is useful for networks that use VLSMs. However, to prevent excessive link-state calculations by all OSPF routers on the network, make every effort to assign subnets that are as continuous as possible.

Exchange of routing information

An OSPF router stores its information about the network in a topological database and propagates only changes to the database. Selected neighboring routers form relationships, referred to as *adjacencies*, for the purpose of exchanging routing information. Not every pair of neighboring routers become adjacent. Routers connected by point-to-point networks and virtual links always become adjacent. On multiaccess networks, all routers become adjacent to routers identified as the designated router (DR) and the backup designated router (BDR).

As the adjacency is established, the neighbors exchange databases and build a consistent, synchronized database between them. When an OSPF router detects a change on one of its interfaces, it modifies its topological database and multicasts the change to its adjacent neighbors, which in turn propagate the change to their adjacent neighbors, until all routers within an area have synchronized topological databases. This process provides quick convergence among routers.

A link state advertisement (LSA) is a packet that describes various aspects of an OSPF route. Each LSA is flooded throughout a routing domain. The collected LSAs of all

routers and networks forms the OSPF topological database. Table 6-1 shows the types of LSAs.

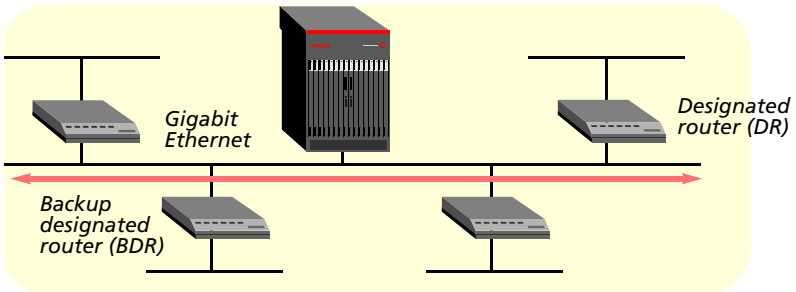
Table 6-1. Description of LSA types

LSA type	Description
Type 1 (RTR) router	Type 1 LSAs describe the collected states of the router's interfaces.
Type 2 (NET) network	Type 2 LSAs describe the set of routers attached to the network.
Types 3 and 4 (Summary LSAs)	Summary LSAs are flooded throughout a single area. Type 3 summary LSAs describe routes to networks. Type 4 summary LSAs describe routes to autonomous system boundary routers.
Type 5 (ASE) AS-external	Type 5 LSAs describe routes to destinations external to the autonomous system (AS). An AS-external-LSA can also describe a default route for the autonomous system. For example, other routers send LSAs to only the designated router by using the All-Designated-Routers multicast address of 224.0.0.6.
Type 7 (ASE) NSSA	NSSAs are like stub areas in that they do not receive or originate type 5 LSAs. However, NSSAs rely solely on default routing for external routes. They employ type 7 LSAs for carrying ASE route information within the area. Type 7 LSAs use a propagate (P) bit to flag the NSSA border router to translate the type 7 LSA into a type 5 LSA, which can then be propagated into other areas.

Designated and backup designated routers on broadcast networks

In OSPF terminology, a broadcast network is any network that has more than two OSPF routers attached and supports the capability to address a single physical message to all the attached routers (broadcast). Neighboring routers are discovered dynamically on these networks using the OSPF Hello protocol, which uses the broadcast capability. Ethernet is an example of a broadcast network. Figure 6-1 shows such a network.

Figure 6-1. OSPF broadcast network on Gigabit Ethernet



To reduce the number of adjacencies each router must form, OSPF calls one of the routers the designated router. As routers begin to form adjacencies, they elect a designated router and then all other routers on the network establish adjacencies, primarily with the designated router. This process simplifies the routing table update procedure and reduces the number of link-state records in the database. The designated router plays other important roles as well to reduce the overhead of OSPF link-state procedures.

To prevent the designated router from becoming a serious liability to the network if it fails, OSPF routers also elect a backup designated router at the same time. Other routers maintain adjacencies with both the designated router and its backup, but the backup router leaves as many of the processing tasks as possible to the designated router. If the designated router fails, the backup immediately becomes the designated router and a new backup is elected.

You choose the designated router on the basis of the processing power, speed, and memory of the system, then assign priorities to other routers on the network, in case both the designated and backup designated routers fail.

Note The Stinger unit can function as a designated router or backup designated router. However, many sites choose to assign a LAN-based router for these roles to dedicate the Stinger unit to WAN processing.

Routing across NBMA interfaces

An OSPF nonbroadcast multiaccess (NBMA) network is any network that has multiple points of access (more than two routers) and does not support broadcast capability. OSPF routers operate on an NBMA network much as they do on a broadcast network, by using the Hello protocol to form adjacencies and identify the designated router. However, because the routers cannot discover their neighboring routers dynamically by means of broadcasts, you must specify some additional parameters. For Stinger units, a WAN link is always point-to-point. However, you can configure the system to interact with NBMA routers via a virtual circuit across an ATM trunk interface.

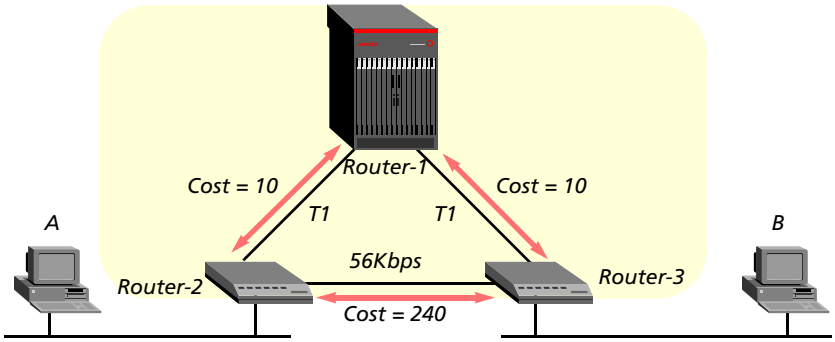
Configurable cost metrics

You assign a cost to the output side of each router interface. The lower the cost, the more likely the interface is to be used to forward data traffic. Costs can also be associated with the externally derived routing data.

The OSPF cost can also be used for preferred-path selection. If two paths to a destination have equal costs, you can assign a higher cost to one of the paths so it can be a backup to be used only when the primary path is not available.

Figure 6-2 shows how costs are used to direct traffic over high-speed links. For example, if Router-2 in Figure 6-2 receives packets destined for Host B, it routes them through Router-1 across two T1 links (`cost=20`) rather than across one 56Kbps B channel to Router-3 (`cost=240`).

Figure 6-2. OSPF costs for different types of links



The Stinger unit has a default cost of 1 for a connected route (Ethernet) and 10 for a WAN link. If two paths have the same destination, the Stinger unit uses the path with the lower cost unless route preferences change the equation.

When assigning costs, remember to account for the bandwidth of a connection. For example, for a single B-channel connection, the cost is 24 times greater than for a T1 link.

Note Be careful when assigning costs. Incorrect cost metrics can cause delays and congestion on the network.

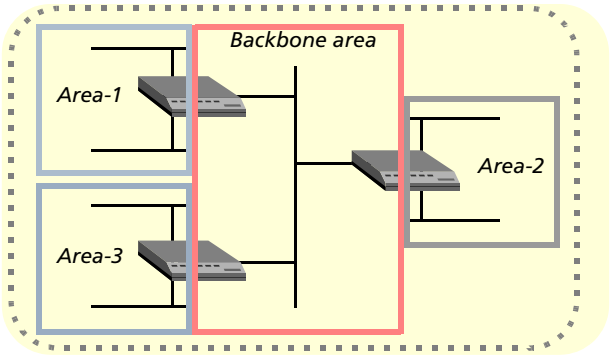
Hierarchical routing (areas)

If a network becomes too large, the size of the database, time required for route computation, and related network traffic become excessive. You can partition an autonomous system into areas to provide hierarchical routing, with a backbone area connecting the other areas. The backbone area is special and always has the area number 0.0.0.0. The backbone consists of networks not contained in any area, their attached routers, and routers that belong to multiple areas.

The backbone must be contiguous. You can use virtual links to connect two backbone routers that have an interface to a common nonbackbone area. OSPF treats two routers joined by a virtual link as if they were connected by an unnumbered point-to-point network. The backbone distributes routing information between areas and has all the properties of an area. The topology of the backbone is invisible to each of the areas, while the backbone itself has no information about area topology.

Each area acts as its own network: All area-specific routing information stays within the area, and all routers within an area must have a synchronized topological database. To tie the areas together, some routers belong to the backbone area and also to one of the other areas. These routers are area border routers (ABRs). In Figure 6-3, all the routers are ABRs.

Figure 6-3. Dividing an OSPF autonomous system into areas



With the ABRs and area boundaries set up correctly, link-state databases are unique to an area. You can configure the Stinger unit to route in normal areas, stub areas, and NSSAs. These different kinds of areas handle the autonomous system external (ASE) routes originated by ASBRs in the following ways.

- Normal areas

  An OSPF normal area allows type 5 LSAs to be flooded throughout the area.

- Stub areas

  Areas that are connected only to the backbone area by one ABR have one exit point and need not maintain information about external routes. To reduce the cost of routing, OSPF supports stub areas in which a default route summarizes all external routes. A stub area allows no type 5 LSAs to be propagated into or throughout the area, and instead depends on default routing to external destinations.

- NSSAs

  NSSAs are like stub areas in that they do not receive or originate type 5 LSAs. However, NSSAs rely solely on default routing for external routes. They employ type 7 LSAs for carrying ASE route information within the area. Type 7 LSAs use a P-bit to flag the NSSA border router to translate the type 7 LSA into a type 5 LSA, which can then be propagated into other areas.

  When the Stinger unit is routing OSPF in an NSSA, it imports ASE routes defined in local or RADIUS profiles as type 7 LSAs. These imported ASE LSAs always have the P-bit enabled, which flags border routers to translate them into type 5 LSAs.

  You can list the router IDs of NSSA border routers that are translating type 7 LSAs to type 5 LSAs, by entering the `ospf translators` command. For example:

```
admin> ospf translators
Area ID     Router ID
0.0.0.1     10.105.0.13
0.0.0.2     12.1.1.1
```

Note For details about the NSSA specification, see RFC 1587, *The OSPF NSSA Option*.

Link-state routing algorithms

The link-state routing algorithms require that all routers within a domain maintain synchronized (identical) topological databases, and that the databases describe the complete topology of the domain. An OSPF router's domain can be an autonomous system or an area within an autonomous system.

OSPF routers create and update a link-state database from information exchanged with other routers. Link-state databases are synchronized between pairs of adjacent routers (as described in "Exchange of routing information" on page 6-2). In addition, each OSPF router uses its link-state database to calculate a self-rooted tree of shortest paths to all destinations. The routing table is built from these calculated shortest-path trees. For example, consider the network topology in Figure 6-4.

Figure 6-4. Sample OSPF topology

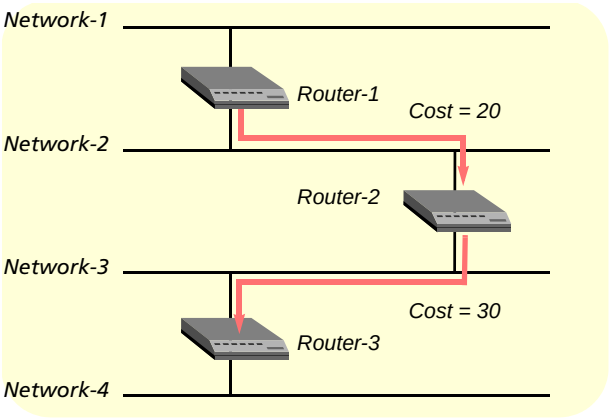


Table 6-2 shows the relevant information in the routers' link-state databases.

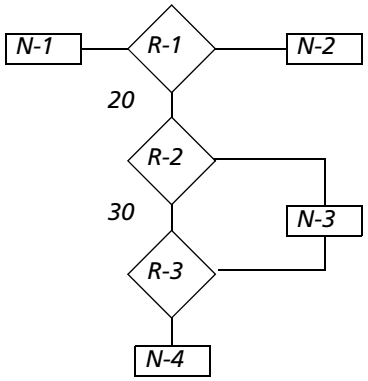
Table 6-2. Link-state databases for OSPF topology in Figure 6-4

Router-1	Router-2	Router-3
Network-1/Cost 0	Network-2/Cost 0	Network-3/Cost 0
Network-2/Cost 0	Network-3/Cost 0	Network-4/Cost 0
Router-2/Cost 20	Router-1/Cost 20	Router-2/Cost 30
	Router-3/Cost 30	

From the link-state database, each router builds a self-rooted shortest-path tree, and then calculates a routing table stating the shortest path to each destination in the autonomous system. (See Table 6-3, Table 6-4, and Table 6-5.) The table also includes externally derived routing information.

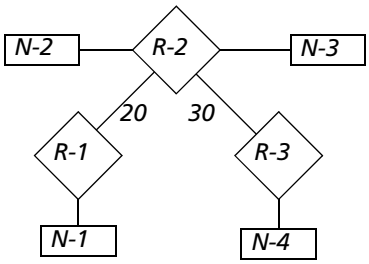
All the routers calculate a routing table of shortest paths, based on the link-state database. Externally derived routing data is advertised throughout the autonomous system but is kept separate from the link-state data. Each external route can also be tagged by the advertising router, enabling the passing of additional information between routers on the boundary of the autonomous system.

Table 6-3. Shortest-path tree and resulting routing table for Router-1



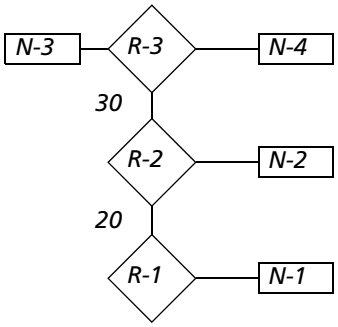
Destination	*Next hop*	*Metric*
Network-1	Direct	0
Network-2	Direct	0
Network-3	Router-2	20
Network-4	Router-2	50

Table 6-4. Shortest-path tree and resulting routing table for Router-2



Destination	*Next hop*	*Metric*
Network-1	Router-1	20
Network-2	Direct	0
Network-3	Direct	0
Network-4	Router-3	30

Table 6-5. Shortest-path tree and resulting routing table for Router-3



Destination	*Next hop*	*Metric*
Network-1	Router-2	50
Network-2	Router-2	30
Network-3	Direct	0
Network-4	Direct	0

Enabling OSPF systemwide

Before the Stinger IP2000 can route OSPF, it must be configured for IP routing. For details about configuring IP routing, see Chapter 4, "IP Routing."

To configure the system to use OSPF routing, you must configure each LAN or WAN interface that will support OSPF routing, and enable the protocol systemwide. The following parameters, shown with default values, enable the protocol and specify a few global settings:

```
[in IP-GLOBAL:ospf-global]
enable = no
```

```
as-boundary-router = yes
ospf-max-lsa = 0
```

Parameter	Setting
`enable`	Enables or disables the OSPF protocol systemwide. Set to `yes` to enable the protocol. If set to `no` (the default), the protocol is disabled systemwide. If you are modifying several OSPF-related profiles, you can use the `no` setting to prevent the OSPF subsystem from reinitializing whenever you write a modified profile. Then set the parameter to `yes` when the modifications are complete. A change to the setting takes effect immediately after you write the profile.
`as-boundary-router`	Enables or disables ASBR calculations related to external routes.
`ospf-max-lsa`	Maximum number of LSAs allowed in the link-state database. Specify a number from 0 through 4,294,967,295. The default setting is 0.

For example, the following commands enable the OSPF protocol:

```
admin> read ip-global
IP-GLOBAL read
admin> set ospf-global enable = yes
admin> write -f
IP-GLOBAL written
```

Configuring OSPF on Gigabit Ethernet

Before the Stinger IP2000 can route OSPF, it must be configured for IP routing. For details about configuring `ip-interface` profiles, see "Configuring ip-interface profiles for Ethernet ports" on page 4-6.

Overview of `ip-interface ospf` settings

Following are the OSPF parameters, shown with default values, for configuring OSPF routing on the controller's GigE interface:

```
[in IP-INTERFACE/{ { any-shelf any-slot 0 } 0 }:ospf]
active = no
area = 0.0.0.0
area-type = normal
hello-interval = 10
dead-interval = 40
priority = 5
authen-type = simple
auth-key = *******
key-id = 0
cost = 1
down-cost = 16777215
ase-type = type-1
```

```
ase-tag = c0:00:00:00
transit-delay = 1
retransmit-interval = 5
non-multicast = no
network-type = Broadcast
poll-interval = 10
profile-type = lan
md5-auth-key = *******
```

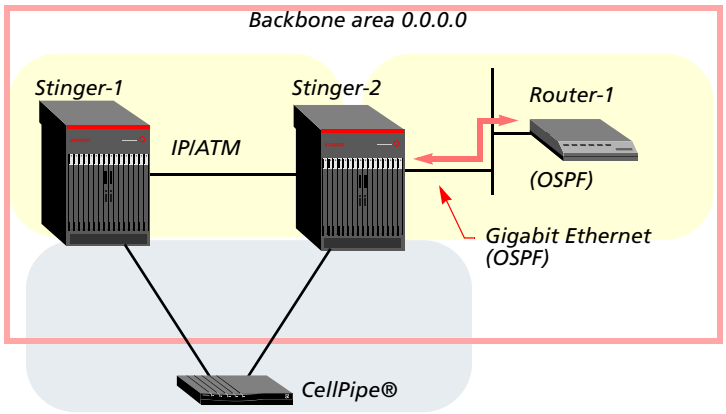
Parameter	Setting	
`active`	Enables or disables OSPF on an interface.	
`area`	OSPF area number in dotted decimal notation. The default `0.0.0.0` represents the backbone area.	
`area-type`	Type of area. The default is the `normal` area type.	
`hello-interval`	Number of seconds between Hello packets.	
`dead-interval`	Number of seconds without receiving a Hello packet before instituting a link-state change.	
`priority`	Priority value, from 0 to 255, used to elect a designated router and backup designated router. A setting of `0` excludes the Stinger from becoming a designated router or backup designated router. The higher the priority value of the Stinger IP2000 relative to other OSPF routers on the network, the better the chances that it will become one of these routers. For details, see "Designated and backup designated routers on broadcast networks" on page 6-3.	
`authen-type`	Type of authentication to use.	
	`none`	No authentication is required.
	`simple`	The router uses the password supplied in the `auth-key` parameter to validate OSPF packet exchanges. This is the default value.
	`md5`	The router uses MD5 encryption and the authentication key ID supplied by the `key-id` parameter to validate OSPF packet exchanges. For related information, see "Authentication" on page 6-2.
`auth-key`	Secret key for authenticating traffic in the router's area. Enter a text string of up to 8 characters. When `authen-type` is set to `md5`, you must set the `md5-auth-key` parameter to specify a key.	
`key-id`	Number from 0 to 255, used to encrypt the secret key when `authen-type` is set to `md5`.	

Parameter	Setting
`cost`	Cost of routing to the interface. The lower the cost assigned to a route, the more likely it is to be used to forward traffic. For details, see "Configurable cost metrics" on page 6-4.
`down-cost`	Cost applied to the interface when it is unavailable. The output cost when the link is physically unavailable but virtually active.
`ase-type`	Type of metric to apply to routes learned from RIP. This parameter applies in a `connection` profile only when OSPF is *not* active. `type-1` — Expresses the metric in the same units as the interface cost. This is the default. `type-2` — Metric is larger than any link-state path.
`ase-tag`	A hexadecimal number attached to an external route, set by default in Stinger systems to `c0:00:00:00`. This is not used by the OSPF protocol itself. It may be used to communicate information between AS boundary routers, and may appear in management utilities to indicate a route is external.
`transit-delay`	Estimated number of seconds required to transmit a Link State Update packet over this interface, taking into account transmission and propagation delays. On a connected route, you can leave the default of `1`.
`retransmit-interval`	Number of seconds between LSA retransmissions for adjacencies belonging to this interface. Its value is also used when retransmitting database description and link-state request packets. On a typical connected route, accept the default setting of `5`.
`non-multicast`	Supports NBMA configuration to a GRF® multigigabit router. See "Sample configuration of NBMA across point-to-point" on page 6-14.
`network-type`	Type of OSPF interface. `broadcast` — A broadcast-capable network, such as Ethernet. `nonbroadcast` — An NBMA network, such as a trunk interface. `point-to-point` — A point-to-point network, consisting of two routers only.
`poll-interval`	Not used on a broadcast network.
`profile-type`	A read-only parameter used internally to verify settings in the profile.
`md5-auth-key`	Secret key to be used for MD5 authentication, up to 16 characters. The default value is `ascend0`. When `authen-type` is set to `md5`, you must supply a value for this parameter.

Sample Gigabit Ethernet interface configuration

Figure 6-5 shows three OSPF routers in the backbone area of an autonomous system. Because all OSPF routers are in the same area, the routers form adjacencies and synchronize their databases. This example shows how to configure the Gigabit Ethernet interface of the unit labeled Stinger-2.

Figure 6-5. OSPF on a LAN interface



All OSPF routers in Figure 6-5 have RIP turned off. Running both RIP and OSPF is unnecessary, and turning RIP off reduces processor overhead. OSPF can learn routes from RIP interfaces, incorporate them in the routing table, assign them an external metric, and tag them as external routes.

Although RFC 2328 does not specify a limitation for the number of routers in the backbone area, keep the number of routers relatively small, because changes that occur in area zero are propagated throughout the autonomous system. Another way to configure the same units is to create a second area (such as 0.0.0.1) in one of the existing OSPF routers, and add the Stinger IP2000 to that area. You can then assign the same area number (0.0.0.1) to all OSPF routers reached through the Stinger IP2000 across a WAN link.

The following sample commands show how to configure `Stinger-2` in Figure 6-5. The commands assign the IP address 10.168.8.17/24 to the local interface and configure the OSPF router in the backbone area:

```
admin> read ip-interface { { 1 8 2 } 0 }
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } read

admin> set ip-address = 10.168.8.17/24

admin> set rip-mode = routing-off

admin> set ospf active = yes

admin> write -f
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } written
```

The following sample commands show how to configure the IP interface for MD5 authentication:

```
admin> read ip-interface { { 1 8 2 } 0 }
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } read

admin> set ospf authen-type = md5
```

```
admin> set ospf md5-auth-key = 12!secret*34key
admin> write -f
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } written
```

Configuring OSPF on an ATM trunk interface

Before the Stinger IP2000 can route OSPF, it must be configured for IP routing. For details about configuring `connection` profiles for IP routing, see "Configuring IP connection interfaces for CPE devices" on page 4-31. In Stinger IP2000 systems, OSPF routing across an ATM trunk interface uses a point-to-point link.

Overview of `connection ospf-options` settings

Following are the parameters, shown with default settings, for configuring OSPF on an ATM trunk interface.

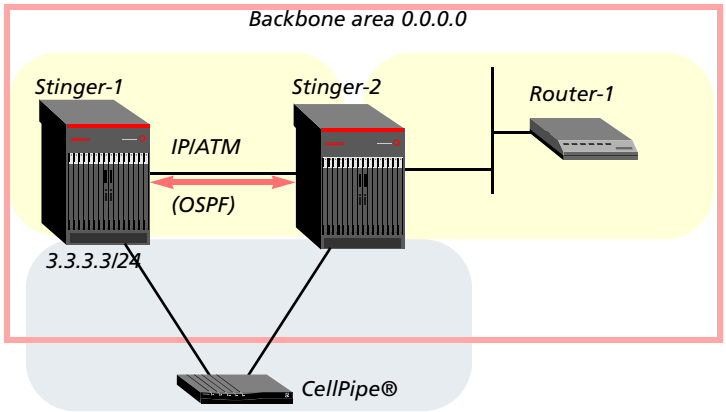
```
[in CONNECTION/"":ip-options:ospf-options]
active = no
area = 0.0.0.0
area-type = normal
hello-interval = 30
dead-interval = 120
priority = 5
authen-type = simple
auth-key = *******
key-id = 0
cost = 10
down-cost = 1000
ase-type = type-1
ase-tag = c0:00:00:00
transit-delay = 1
retransmit-interval = 5
non-multicast = no
network-type = Point-to-Point
poll-interval = 10
profile-type = wan
md5-auth-key = *******
```

These are the same parameters described for enabling OSPF on the Gigabit Ethernet interface. For definitions, see "Overview of ip-interface ospf settings" on page 6-9, or the *Stinger Reference*.

Sample OSPF point-to-point configuration

This example shows how to configure a `connection` profile in the system labeled Stinger-2 in Figure 3-6, to enable it to route OSPF across the ATM cloud to Stinger-1. In this example, the unit labeled Stinger-1 uses the IP address 3.3.3.3/24.

Figure 6-6. OSPF over ATM point to point



The following commands configure OSPF in the unit labeled Stinger-2 in Figure 6-6:

```
admin> read conn stinger1-atmvc
CONNECTION/stinger1-atmvc read
admin> set active = yes
admin> set encapsulation-protocol = atm
admin> set ip-options remote-address = 3.3.3.3/24
admin> set ip-options rip = routing-off
admin> set ip-options ospf active = yes
admin> set atm-options atm1483type = aal5-llc
admin> set atm-options vci = 101
admin> set atm-options nailed-group = 802
admin> write -f
CONNECTION/stinger1-atmvc written
```

Sample configuration of NBMA across point-to-point

With the current software version, NBMA is supported only on a point-to-point WAN link. Full support for multiaccess is not provided.

Overview of additional NBMA settings

In addition to the standard settings for OSPF point-to-point, the following parameters, shown with default settings, must be configured to support NBMA:

```
[in CONNECTION/"":ip-options:ospf-options]
non-multicast = no
network-type = Point-to-Point
[in CONNECTION/"":ip-options]
local-address = 0.0.0.0/0
[in OSPF-NBMA-NEIGHBOR/""]
name* = ""
host-name = ""
```

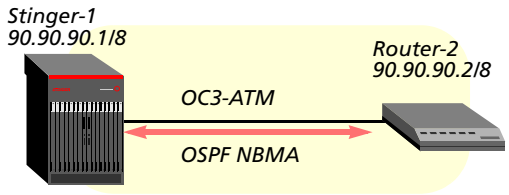
```
ip-address = 0.0.0.0
dr-capable = no
```

Parameter	Setting
non-multicast	For an NBMA connection to a GRF® multigigabit router, the non-multicast parameter must be set to yes. This causes the translation of the multicast traffic to directed traffic. This setting is required only when connecting to a GRF® multigigabit router.
network-type	For NBMA, the network-type parameter must be set to NonBroadcast.
local-ip-address	When network-type is set to NonBroadcast, a local-ip-address value must be provided or the system reports a configuration error such as the following: OSPF CONFIG ERROR: NBMA profile router-2 has illegal 0.0.0.0 address
ospf-nbma-neighbor: name	Name of the ospf-nbma-neighbor profile.
ospf-nbma-neighbor: host-name	Name of the local connection profile that defines the connection to the neighboring router.
ospf-nbma-neighbor: ip-address	IP address of the neighboring router.
ospf-nbma-neighbor: dr-capable	Whether the neighboring router can be the designated router (yes or no).

Example of an NBMA configuration

Figure 6-7 shows a Stinger IP2000 connecting point-to-point with another router that operates in an OSPF NBMA network.

Figure 6-7. OSPF NBMA over ATM point to point



Following is a connection profile on the system labeled Stinger-1 to enable NBMA on the link to Router-2:

```
admin> new connection router-2
CONNECTION/router-2 read
admin> set active = yes
admin> set encapsulation-protocol = atm
admin> set ip-options remote-address = 90.90.90.2/8
admin> set ip-options local-address = 90.90.90.1/8
```

```
admin> set ip-options ospf-options active = yes
admin> set ip-options ospf-options authen-type = simple
admin> set ip-options ospf-options auth-key = mykey
admin> set ip-options ospf-options network-type = NonBroadcast
admin> set telco-options nailed-groups = 851
admin> set mp-options enabled = no
admin> set atm-options atm1483type = aal5-llc
admin> set atm-options vci = 101
admin> set atm-options nailed-group = 851
admin> write -f
CONNECTION/router-2 written
```

The following profile enables the unit to form an adjacency with Router-2:

```
admin> new ospf-nbma-neighbor router-2
OSPF-NBMA-NEIGHBOR/router-2 read
admin> set host-name = router-2
admin> set ip-address = 90.90.90.2/8
admin> write -f
OSPF-NBMA-NEIGHBOR/router-2 written
```

Configuring global route options that apply to OSPF

The `ip-global` profile contains several settings that apply only when OSPF routing is in use. Following are the relevant parameters, shown here with their default settings:

```
[in IP-GLOBAL]
pool-ospf-adv-type = type-1
ospf-pref = 10
ospf-ase-pref = 150
ospf-global = { no yes 0 }
rip-tag = c8:00:00:00
rip-ase-type = 1
```

Parameter	Setting
`pool-ospf-adv-type`	Type of ASE metric applied to summarized pools imported into OSPF as external routes.
`ospf-pref`	Preference value for routes learned from OSPF. Valid values are 0 to 255. The default value is 10.
`ospf-ase-pref`	Preference value for routes learned from RIP, ICMP, or another non-OSPF protocol. Specify a value from 0 through 255. By default, routes learned dynamically from another routing protocol are assigned a preference value of 150.
`rip-tag`	Hexadecimal number associated with routes learned from RIP. OSPF border routers can use the tag to filter a record.

Parameter	Setting
rip-ase-type	Type of ASE metric applied to routes learned from RIP.

Example of importing a summarized pool as an ASE

The following commands configure a summarized pool and import it to OSPF with a type 1 OSPF metric:

```
admin> read ip-global
IP-GLOBAL read
admin> set pool-summary = yes
admin> set pool-base-address 1 = 10.12.253.1
admin> set assign-count 1 = 62
admin> set pool-ospf-adv-type = type-1
admin> write -f
IP-GLOBAL written
```

When pool-summary is set to yes and OSPF is enabled, the OSPF subsystem uses the pool-ospf-adv-type parameter to determine how to import summarized routes into OSPF. If this parameter is set to type-1, the metric for the route to a summarized pool is expressed in the same units as the link-state metric (interface cost).

If pool-ospf-adv-type is set to type-2, the unit considers the routing between autonomous systems as the major cost of routing a packet, and conversion of external costs to internal link-state metrics is unnecessary. If the parameter is set to internal, the summarized pool addresses are imported into OSPF as intra-area routes, which enables them to work properly with stub areas.

Example of setting ASE preferences

The ospf-pref and ospf-ase-pref parameters determine the preference values assigned to routes learned from other OSPF routers and those imported from other dynamic routing protocols. The default settings place a much lower preference on OSPF routes, which means that the routes learned from other protocols (ASE routes) are more likely to be used. The following commands decrease to 100 the preference assigned to ASE routes (the default is 150):

```
admin> read ip-global
IP-GLOBAL read
admin> set ospf-ase-pref = 100
admin> write -f
IP-GLOBAL written
```

Configuring ip-route OSPF options

For details about configuring static routes, see “Configuring ip-route profiles” on page 4-29. The following parameters in the ip-route profile (shown with sample settings), apply only when OSPF is enabled:

```
in IP-ROUTE/""]
cost = 1
ase-type = type-1
```

```
ase-tag = c0:00:00:00
ase7-adv = N/A
```

Parameter	Setting
cost	Cost of routing to the interface. The lower the cost, the more likely the interface is to be used to forward traffic. See "Configurable cost metrics" on page 6-4.
ase-type	Type of metric to apply to routes learned from RIP. The default value of type-1 expresses the metric in the same units as the interface cost. With the value of type-2, the metric is larger than any link-state path.
ase-tag	Hexadecimal number that appears in management utilities and flags this route as external. It can also be used by border routers to filter this record.
ase7-adv	*Currently not used.*

Example of configuring a type 7 LSA in an NSSA

For background information about NSSAs, see "Hierarchical routing (areas)" on page 6-5. To configure the Stinger unit to route OSPF in an NSSA, *all* OSPF interfaces in the Stinger IP2000 must specify the NSSA area-type.

To configure a type 7 LSA, you must specify a static route in an ip-route profile. Following are the related parameters (shown with sample settings):

```
[in IP-ROUTE/external]
name* = external
dest-address = 10.4.5.0/22
gateway-address = 10.4.5.7
metric = 0
cost = 1
preference = 100
third-party = no
ase-type = type-1
ase-tag = c0:00:00:00
private-route = no
active-route = yes
ase7-adv = n/a
```

The following procedure configures the Stinger unit to route in an NSSA and import a type 7 LSA that specifies an external route across the WAN link:

1 Assign an NSSA area type to the IP interface that is running OSPF. For example:

```
admin> read ip-interface { { 1 8 2 } 0 }
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } read

admin> set ospf area-type = nssa

admin> write -f
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } written
```

2 Configure the WAN link that represents an ASE route. For example:

```
admin> read connection ase-link
CONNECTION/ase-link read
admin> set ip-options remote = 10.4.5.7/22
admin> set ip-options rip = routing-off
admin> set ip-options ospf active = yes
admin> write -f
CONNECTION/ase-link written
```

3 Configure a static route to the remote site. For example:

```
admin> new ip-route type7
IP-ROUTE/type7 read
admin> set dest = 10.4.5.0/22
admin> set gateway = 10.4.5.7
admin> write -f
IP-ROUTE/type7 written
```

Example of assigning a cost to a static route

The lower the cost assigned to a route, the more likely the router is to choose the route to forward traffic. Typically, you account for the bandwidth of a connection when assigning costs. The Stinger unit has a default cost of 1 for a connected route (Ethernet) and 10 for an ATM VC. If two paths have the same destination, the Stinger unit uses the path with the lower cost.

Note Be careful when assigning costs. Incorrect cost metrics can cause delays and congestion on the network.

In the following example, an administrator assigns a cost of 25 to a static route:

```
admin> new ip-route mylink
IP-ROUTE/mylink read
admin> set dest = 10.1.2.0/24
admin> set gateway = 10.9.8.10
admin> set cost = 25
admin> write -f
IP-ROUTE/mylink written
```

Administrative tools for OSPF routing

The `ospf` diagnostic-level commands display information related to OSPF routing, including the link state advertisements (LSAs); the routing table for border routers; and the OSPF areas, interfaces, statistics, and routing table. For details, see the chapter on monitoring OSPF in the *Stinger Adminstration Guide*.

Broadband RAS Configuration 7

With broadband remote access server (BRAS) support, a Stinger IP2000 can terminate PPP over ATM (PPPoA) and PPP over Ethernet (PPPoE) sessions received on a LIM DSL interface.

When a DSL interface receives PPPoA or PPPoE packet streams, the system switches the inbound traffic to the LIM's ATM internal interface on the basis of an ATM circuit configuration. The IP2000 controller then uses IP routing to forward the packet streams to the appropriate egress interface. For background information about ATM circuit configurations, see the *Stinger ATM Configuration Guide*.

Note PPPoA and PPPoE sessions are supported on DSL interfaces only, but are not currently supported over T1/E1/IMA or IDSL ports. PPPoA and PPPoE are not supported across trunk or Ethernet interfaces.

Recommended `call-type` setting for PPP sessions

The system assigns nailed group 1 to LIM slot 1 port 1 ({ 1 1 1 }), and also assigns nailed group 1 to permanent, leased PPP connections. Because the default `call-type` in Stinger systems is nailed (`ft1`), a conflict occurs when you configure PPPoA or PPPoE connections on any LIM interface, and { 1 1 1 } is enabled.

Following are representative default settings showing the conflicting nailed group assignments:

```
[in DS1-ATM/{ shelf-1 slot-1 1 }:line-config]
nailed-group = 1
```

```
[in CONNECTION/"":telco-options]
call-type = ft1
nailed-groups = 1
```

Note Because these defaults conflict, setting the `call-type` value to `off` is recommended for all PPP connections.

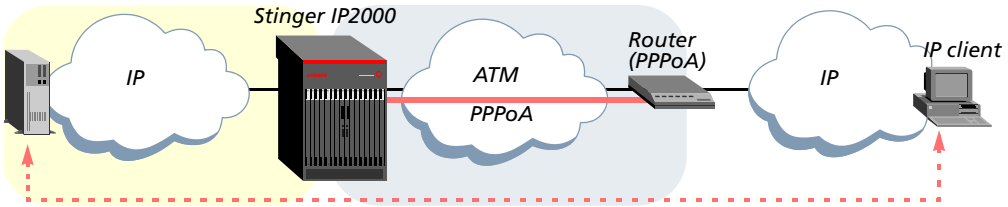
With the default `call-type` setting of `ft1` in a PPP profile, if { 1 1 1 } is enabled (with or without connections), the system generates rolling LOG errors, LOG warning, and LOG information messages. This condition is prevented by setting `call-type` to `off`.

In addition, when `call-type` is set to `off`, the system is able to terminate PPP sessions automatically following a period of client inactivity, and reestablish them when the client becomes active again.

Overview of PPPoA and PPPoE topologies

A PPPoA connection uses the ATM adaptation layer 5 (AAL5) protocol as a framing mechanism across point-to-point virtual circuits, as described in RFC 2364, *PPP over AAL5*. Only VC-multiplexed PPPoA is currently supported. The PPPoA session is between the router and the Stinger unit, as shown in Figure 7-1. The PPPoA session enables an IP client on the far side of the PPPoA router to connect through the Stinger unit to the IP cloud beyond it.

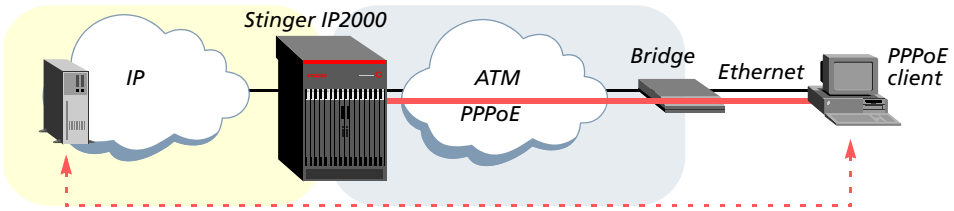
Figure 7-1. PPPoA topology



For PPPoE, the connection uses Ethernet-bridged framing, as defined in RFC 2516, *A Method for Transmitting PPP Over Ethernet (PPPoE)*. PPPoE is negotiated in two phases: a discovery phase and a session phase. During the discovery phase, the PPPoE client negotiates with the Stinger to obtain information it requires. When the session has been established, the client sends PPP packets encapsulated in Ethernet-bridged frames.

The PPPoE sessions shown in Figure 7-2 are between a PPPoE client connected through the bridge, and the Stinger unit. The PPPoE session enables the client to connect through the Stinger unit to the IP cloud beyond it.

Figure 7-2. PPPoE topology



Note Currently, the Stinger IP2000 does not support multiclient PPPoE across a single DSL interface.

Required setup for PPPoA and PPPoE connections

To enable establishment of PPPoA or PPPoE connections using IP routing, you must complete the following steps:

1. Configure the `answer-defaults` profile to accept PPP session requests and require their authentication. You do this once, to enable the system to process subsequent PPP session requests. See "Configuring the answer-defaults profile for PPP sessions" on page 7-3.
2. Configure an ATM circuit `connection` profile between the DSL interface on which the PPPoE or PPPoA connection will be established and the LIM internal interface. See "Terminating traffic on a LIM internal interface" on page 7-5.
3. Configure a `connection` or RADIUS profile for each PPPoE or PPPoA session, as described in "Example of configuring a PPPoA connection" on page 7-6 or "Example of configuring a PPPoE connection" on page 7-8.

If you are using RADIUS to externally authenticate PPP sessions, see the *TAOS RADIUS Guide and Reference*.

Configuring the `answer-defaults` profile for PPP sessions

To control access, PPP sessions typically require password authentication each time a PPP session is opened. Stinger units support RFC 1334 PPP Authentication Protocol (PAP) and RFC 1994 Challenge Handshake Authentication Protocol (CHAP), which must be negotiated with the client on the basis of settings in the `answer-defaults` profile and the client's `connection` or RADIUS profile.

The values in the `answer-defaults` profile are applied before the system locates the `connection` or RADIUS profile associated with the request. If the PPP client's profile contains a similar parameter with a different value, the connection-specific value overrides the `answer-defaults` value when building the session. If no similar value is found in the `connection` or RADIUS profile, the `answer-defaults` setting is used.

Following are some `answer-defaults` parameters, shown with default settings, that affect PPP authentication and session timeout:

```
[in ANSWER-DEFAULTS]
profiles-required = yes

[in ANSWER-DEFAULTS:ppp-answer]
enabled = yes
receive-auth-mode = no-ppp-auth
bi-directional-auth = none
substitute-send-name = ""

[in ANSWER-DEFAULTS:session-info]
idle-timer = 120
max-call-duration = 0
```

Parameter	Setting
`profiles-required`	A setting of `yes` (the default) prevents unauthenticated sessions. If set to `no`, the system builds a temporary profile for session requests for which it cannot locate a configured profile.

Parameter	Setting
enabled	The enabled parameter must be set to yes (the default) for the system to answer PPP session requests.
receive-auth-mode	With the default no-ppp-auth setting, the Stinger unit does not request authentication. If set to a non-default value, the Stinger unit requests an authentication protocol, and the client must accept one of the options the system offers.
bi-directional-auth	Support for bidirectional CHAP. If set to allowed or required, the system negotiates bidirectional CHAP if the client's connection profile specifies the proper settings.
substitute-send-name	System name to send to clients for bidirectional CHAP authentication, if different from the name setting in the system profile.
idle-timer	With a call-type setting of off in a client profile, the system uses the idle-timer value to terminate the session after a default interval of 2 minutes. You can configure a different default interval here or in the client's session-options subprofile, by specifying the maximum number of consecutive seconds a session can remain idle before it is terminated.
max-call-duration	Maximum number of minutes of connect time for a PPP session. The default zero value disables the timer.

Most sites change the default setting of the receive-auth-mode parameter to ensure authentication of a PPP request before a session can be established. For example:

```
admin> read answer-defaults
ANSWER-DEFAULTS read

admin> set ppp-options receive-auth-mode = any-ppp-auth

admin> write -f
ANSWER-DEFAULTS written
```

With this setting, the system accepts session requests that provide any of the supported PPP authentication methods, but it drops requests that do not offer any authentication protocols during session negotiation.

The following commands enable bidirectional authentication for sessions that use CHAP and specify the proper settings in the connection or RADIUS profile:

```
admin> set ppp-answer bidirectional-auth = allowed

admin> write -f
ANSWER-DEFAULTS written
```

With these settings, if a calling device accepts CHAP authentication, the system attempts to negotiate bidirectional CHAP, but does not reject the request if the negotiation fails. However, if bidirectional CHAP is negotiated, authentication must succeed in both directions. For related information, see "Sample PPPoA connection with bidirectional CHAP authentication" on page 7-7.

Terminating traffic on a LIM internal interface

For all installed LIMs that can terminate PPPoA or PPPoE calls, the system creates an `atm-internal` profile for the LIM's internal ATM segmentation assembly and reassembly (SAR) port. The internal interface number is one greater than the highest DSL interface number on the module. For example, the following command output indicates a 48-port LIM in slot 1, a 72-port LIM in slots 2 and 4, and a 24-port LIM in slot 6:

```
admin> dir atm-internal
42  05/09/2003 08:41:34  { shelf-1 slot-1 49 }  1:1:49
42  05/09/2003 08:41:34  { shelf-1 slot-2 73 }  1:2:73
42  05/09/2003 09:30:38  { shelf-1 slot-4 73 }  1:4:73
42  05/09/2003 08:41:34  { shelf-1 slot-6 25 }  1:6:25
38  05/09/2003 08:41:30  { shelf-1 first-control-module 1 } 1:8:1
```

For a DSL interface to handle PPPoA and PPPoE incoming calls, you must configure an ATM circuit between the external DSL interface and the LIM's internal ATM interface. The ATM circuit configuration must specify the nailed group of the internal interface as the second leg of the circuit (in the `atm-connect-options` subprofile).

Note The system prevents configuration of a circuit from one LIM to the ATM internal interface of another LIM, or configurations that attempt to use the internal interface of a LIM in other ways.

To determine the nailed group of a LIM's internal ATM interface, use the `which` command. Following are sample commands for the internal interface of a 72-port LIM in slot 4:

```
admin> which -n { 1 4 73 }
Nailed group corresponding to port { shelf-1 slot-4 73 } is 2271
admin> which -p 2271
The SAR Interface corresponding to nailed group 2271 is:
{ shelf-1 slot-4 73 }
```

The following commands determine the nailed-group number of the first DSL interface in slot 4:

```
admin> which -n { 1 4 1 }
Nailed group corresponding to port { shelf-1 slot-4 1 } is 151
```

The following commands configure the ATM circuit between the DSL interface and the LIM's ATM internal interface:

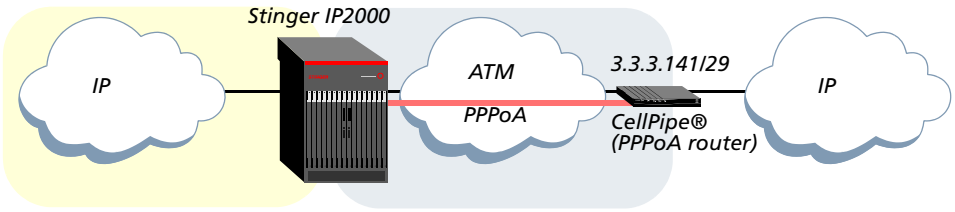
```
admin> new connection cir-1
CONNECTION/cir-1 read
admin> set active = yes
admin> set encapsulation-protocol = atm-circuit
admin> set ip-options ip-routing = no
admin> set mp-options enabled = no
admin> set atm-options vci = 38
admin> set atm-options nailed-group = 151
admin> set atm-connect-options vci = 36
admin> set atm-connect-options nailed-group = 2271
```

```
admin> write -f
CONNECTION/cir-1 written
```

Example of configuring a PPPoA connection

Figure 7-3 shows a Stinger IP2000. Across a DSL interface, a CellPipe® unit is operating as a PPPoA router.

Figure 7-3. Example of a PPPoA session on a DSL interface



Note You can use bidirectional CHAP authentication for PPPoA connection, but it is not required. The sample configuration provides an example of how to use bidirectional authentication.

Overview of PPPoA `connection` settings

For background information about IP routing configurations, see "Configuring IP connection interfaces for CPE devices" on page 4-31.In addition to those settings, following are relevant PPPoA parameters, shown with default settings, including bidirectional CHAP authentication:

```
[in CONNECTION/""]
station* = ""
encapsulation-protocol = atm-circuit

[in CONNECTION/"":ip-options]
ip-routing-enabled = yes
remote-address = 0.0.0.0/0

[in CONNECTION/"":ppp-options]
send-auth-mode = none
bidirectional-auth = none
substitute-recv-name = ""
send-password = ""
recv-password = ""

[in CONNECTION/"":telco-options]
call-type = ft1
```

Parameter	RADIUS attribute	Setting
`station`	User-Name (1)	Name of the PPPoA router. The value is case sensitive and must exactly match the name the router sends during authentication.
`encapsulation-protocol`	Framed-Protocol (7)	Encapsulation protocol. Set to `ppp` for PPPoA clients.
`ip-routing-enabled`	Ascend-Route-IP (228)	Enable/disable IP routing for the interface. IP routing is enabled by default.

Parameter	RADIUS attribute	Setting
`remote-address`	Framed-IP-Address (8) Framed-IP-Netmask (9)	IP address of the PPPoA router, which can include a subnet specification.
`bidirectional-auth`	Ascend-Bi-Directional-Auth (46)	Enable/disable bidirectional CHAP authentication. Used only for bidirectional CHAP.
`send-auth-mode`	Ascend-Send-Auth (231)	Set to `chap-ppp-auth` for bidirectional CHAP. Used only for bidirectional CHAP.
`substitute-rev-name`	Ascend-Recv-Name (45)	Name that must be received from the far end during bidirectional CHAP authentication, if different from the `station` setting. Used only for bidirectional CHAP.
`send-password`	Ascend-Send-Secret (214)	Password the Stinger must send to the far end during bidirectional CHAP, used only for bidirectional CHAP.
`recv-password`	Password (2)	Password the system must receive from the PPPoA router. This setting is used for all PPP authentication methods.
`call-type`	Ascend-Call-Type (177)	Set `call-type` to `off` in PPP profiles. For background information, see "Recommended call-type setting for PPP sessions" on page 7-1.

Sample PPPoA connection with bidirectional CHAP authentication

First, configure an ATM circuit between the DSL interface connecting to the PPPoA router and the LIM's internal interface. See "Terminating traffic on a LIM internal interface" on page 7-5.

Then, verify that the `answer-defaults` profile enables the PPP authentication methods to be used, and configure a `connection` profile to the PPPoA router.

For example, the following commands configure a profile in which bidirectional CHAP authentication is required with the CellPipe® router in Figure 7-3:

```
admin> new connection cellpipe1
CONNECTION/cellpipe1 read
admin> set active = yes
admin> set encapsulation-protocol = ppp
admin> set ip-options remote-address = 3.3.3.141/29
admin> set ppp-options send-auth-mode = chap-ppp-auth
admin> set ppp-options bi-directional-auth = required
admin> set ppp-options send-password = sendpw
admin> set ppp-options recv-password = recvpw
admin> set telco-options call-type = off
admin> write
CONNECTION/cellpipe1 written
```

Following is a comparable RADIUS user profile:

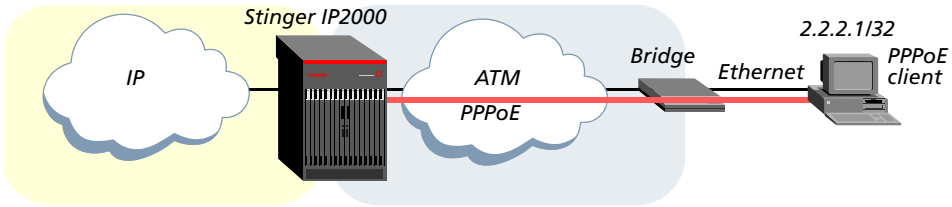
```
cellpipe1 Password = "recvpw"
    Service-Type = Framed-User,
```

```
Ascend-Require-Auth = Require-Auth,
Ascend-Auth-Type = Auth-CHAP,
Ascend-Send-Auth = Send-Auth-CHAP,
Ascend-Bi-Directional-Auth=Bi-Directional-Auth-Required,
Ascend-Send-Secret = "sendpw",
Framed-Protocol = PPP,
Framed-IP-Address = 3.3.3.141,
Framed-IP-Netmask = 255.255.255.248,
Ascend-Call-Type = 0
```

Example of configuring a PPPoE connection

Figure 7-3 shows a Stinger IP2000. Across a DSL interface, a CPE bridging device connects to an Ethernet segment with a PPPoE client.

Figure 7-4. Example of a PPPoE session on a DSL interface



Note You can use bidirectional CHAP authentication for PPPoE connections, but it is not required. For information about using bidirectional CHAP, see "Sample PPPoA connection with bidirectional CHAP authentication" on page 7-7.

Overview of PPPoE **`connection`** settings

For background information about IP routing configurations, see "Configuring IP connection interfaces for CPE devices" on page 4-31. In addition to those settings, following are relevant PPPoE parameters, shown with default setting, including PAP authentication:

```
[in CONNECTION/""]
station* = ""
encapsulation-protocol = atm-circuit

[in CONNECTION/"":ip-options]
ip-routing-enabled = yes
remote-address = 0.0.0.0/0

[in CONNECTION/"":ppp-options]
recv-password = ""

[in CONNECTION/"":pppoe-options]
pppoe = no
bridge-non-pppoe = no

[in CONNECTION/"":telco-options]
call-type = ft1
```

Parameter	RADIUS attribute	Setting
`station`	User-Name (1)	Name of the PPP client system. The value is case sensitive, and must exactly match the name the client presents during authentication.
`encapsulation-protocol`	Framed-Protocol (7)	Encapsulation protocol. Set to `ppp` for PPPoE clients.
`ip-routing-enabled`	Ascend-Route-IP (228)	Enable/disable IP routing for the interface. IP routing is enabled by default.
`remote-address`	Framed-IP-Address (8)	IP address of the remote device, which can include a subnet specification.
	Framed-IP-Netmask (9)	
`recv-password`	Password (2)	Password sent by the PPPoE client.
`pppoe`	Ascend-PPPoE-Enable (74)	Enable/disable processing of PPPoE packets. Must be set to `yes` for a PPPoE connection. If `encapsulation-protocol` is set to `ppp` and `pppoe` is not enabled, the connection is assumed to be PPPoA.
`bridge-non-pppoe`	Ascend-Bridge-Non-PPPoE (75)	*Not currently supported.*
`call-type`	Ascend-Call-Type (177)	Set `call-type` to `off` in PPP profiles. For background information, see "Recommended call-type setting for PPP sessions" on page 7-1.

Sample PPPoE connection using PAP authentication

First, configure an ATM circuit between the DSL interface connecting to the CPE bridging device and the LIM's internal interface. See "Terminating traffic on a LIM internal interface" on page 7-5.

Then, verify that the `answer-defaults` profile enables the PPP authentication methods to be used and configure a `connection` profile to the PPPoE client.

For example, the following commands configure a profile using PPP authentication for the PPPoE client in Figure 7-4:

```
admin> new connection pppoe-1
CONNECTION/pppoe-1 read
admin> set active = yes
admin> set encapsulation-protocol = ppp
admin> set ip-options remote-address = 2.2.2.1/29
admin> set ppp-options recv-password = pppoe1!pw
admin> set pppoe-options pppoe = yes
admin> set telco-options call-type = off
admin> write -f
CONNECTION/pppoe-1 written
```

Following is a comparable RADIUS user profile:

```
pppoe-1 Password = "pppoe1!pw"
    Service-Type = Framed-User,
    Framed-Protocol = PPP,
    Framed-IP-Address = 2.2.2.1,
    Framed-IP-Netmask = 255.255.255.248
    Ascend-PPPoE-Enable = PPPoE-Yes
    Ascend-Call-Type = 0
```

Optional configuration of a LIM ATM internal interface

By default, the ATM internal interface of a LIM is enabled and configured with system-generated values for required settings, so no additional configuration is required. Following are the parameters, shown with default values, for configuring internal ATM interfaces:

```
[in ATM-INTERNAL/{ any-shelf any-slot 0 }]
name = ""
physical-address* = { any-shelf any-slot 0 }
enabled = yes
line-config = { 1 15 }
traffic-shapers = [ { no 1000 1000 2 no 1 } { no 1000 1000 2 no 2 } { no 100+
[in ATM-INTERNAL/{ any-shelf any-slot 0 }:line-config]
nailed-group = 1
vp-switching-vpi = 15
```

Parameter	Setting
`name`	Assigns a name to the interface, up to 15 characters. The name is used only for administrative purposes.
`physical-address`	Physical address of the internal SAR port within the system. This value is set by the system when it creates the `atm-internal` profile, and it is used to retrieve the ATM configuration for the interface.
`enabled`	Enable/disable the interface for use. The interface is enabled by default.
`line-config:nailed-group`	A system-generated unique number that represents the interface in the system. You specify this number in a `connection` or RADIUS profile when creating the ATM circuit for terminating PPPoA or PPPoE connections. **Note** With the current software version, Lucent Technologies does not recommend modifying the system-generated `nailed-group` number assigned by default to the internal interface.
`line-config: vp-switching-vpi`	*VP switching is not supported between LIM interfaces.*
`traffic-shapers`	*Traffic shaping is not currently supported for terminating connections on DSL LIMs (PPPoE/PPPoA).*

Administrative tools for PPP sessions

The system supports several commands that are useful for displaying administrative information about user sessions. For examples, see the *Stinger Adminstration Guide*, and entries in the *Stinger Reference* for commands such as `connection` and `userstat`.

Forwarding Multicast Video

8

IP multicast forwarding is supported with an optional software license. Enter the following command to determine whether the multicast license is enabled:

```
admin> get base igmp
[in BASE]
igmp-np-enabled = yes
```

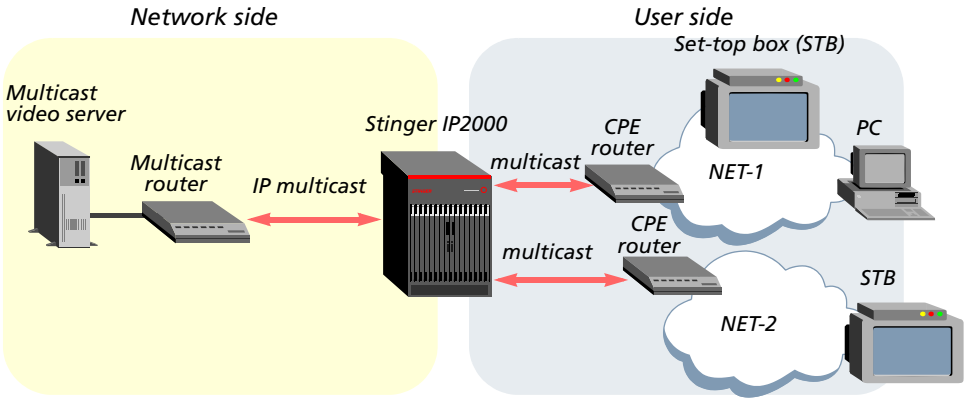
The system sets this parameter to `yes` when the IGMP license is enabled for the IP2000 network processor. If the license is not enabled, the system displays an error message if you configure multicast forwarding on the IP2000. For information about obtaining and enabling Lucent Technologies software licenses, contact your Lucent sales representative.

IP multicast forwarding

IP multicast forwarding enables the Stinger to receive multicast transmissions from multicast backbone (MBONE) routers and forward the transmissions to multiple client interfaces.

A common use for IP multicast is to transmit streaming video across the Internet to applications running on multiple PCs or set-top boxes (STB) for television sets, as shown in Figure 8-1.

Figure 8-1. Multicast video sample setup



To the multicast clients, the Stinger appears to be a multicast router originating the video stream. To the multicast routers, the Stinger appears to be a multicast client, initiating and responding to group management messages via Internet Group Management Protocol (IGMP) version-1 or version-2.

To receive a transmission, the client interfaces must join a specific multicast group. A *multicast group* is a Class D IP address (from 224.0.0.0 to 239.255.255.255). When data is sent to an address in that range, it is multicast to all hosts that have joined that group. The Stinger forwards IGMP messages between clients (hosts) and the multicast router to enable these transactions.

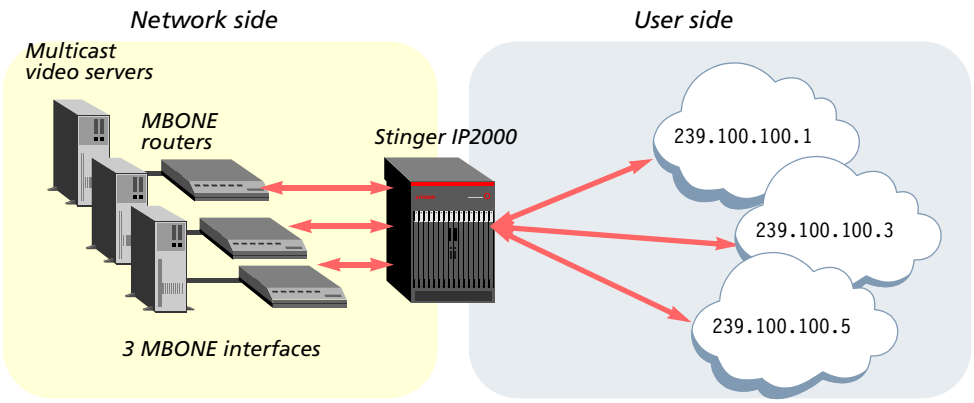
Network-side MBONE interfaces

An MBONE interface, used to receive transmissions from a multicast router, can be either a WAN MBONE on an ATM trunk interface or a LAN MBONE on the IP2000 Gigabit Ethernet interface. On an MBONE interface, the Stinger responds to multicast router queries, and returns responses from clients to the router.

The global IP router can receive multicast data from up to four MBONE interfaces. Multiple MBONE capability is not currently supported for virtual routers.

The multiple MBONE interfaces can be WAN (ATM trunks) or LAN, or any combination of the two interface types, but the total number of MBONE interfaces cannot exceed four. Figure 8-2 shows a Stinger unit with three MBONE interfaces:

Figure 8-2. Multiple MBONE interfaces on trunk or LAN interfaces



Notice about Gigabit Ethernet redundancy for a LAN MBONE

For Stinger systems with redundant controllers, you can configure Gigabit Ethernet redundancy for a LAN MBONE interface to enable the system to maintain MBONE operations across a controller switchover. For details, see "Configuring a redundant LAN MBONE" on page 2-8.

LIM-side multicast client interfaces

The multicast clients must be on LIM interfaces, accessing the Stinger unit through an ATM virtual circuit.

Transmission of multicast data to multicast clients on a local Ethernet interface is not currently supported. For that reason, the following parameters not currently used in the `ip-interface` profile and its `igmp-option` subprofile for the IP2000:

Table 8-1. Unused multicast client settings for LAN interfaces

Unused `ip-interface` settings

`[in IP-INTERFACE/{ { any-shelf any-slot 0 } 0 }]`
`multicast-rate-limit = 100`
`multicast-group-leave-delay = 0`
`multicast-group-leave-delay-msec = 0`
`multicast-service-profile = ""`
`multicast-max-groups = 0`

`[in IP-INTERFACE/{ { any-shelf any-slot 0 } 0 }:igmp-options]`
`robust-count = 2`
`query-interval = 125`
`query-response-interval = 100`
`last-member-query-interval = 10`
`last-member-query-count = 2`

Configuring MBONE interfaces

To configure an MBONE interface, you must complete the following steps:

1. Enable multicast forwarding in the `ip-global` profile.
2. Specify a profile index for each MBONE interface in the `ip-global` profile.
3. Configure the `ip-interface` or `connection` profile for each LAN or WAN MBONE interface. Make sure to set `multicast-allowed` to `yes`.

Note A Stinger IP2000 does not support multicast heartbeat monitoring, so the following `ip-global` settings are not used:

Table 8-2. Unused multicast heartbeat monitoring settings

Unused `ip-global` settings

`[in IP-GLOBAL]`
`multicast-hbeat-addr = 0.0.0.0`
`multicast-hbeat-port = 0`
`multicast-hbeat-slot-time = 0`
`multicast-hbeat-Number-Slot = 0`
`multicast-hbeat-Alarm-threshold = 0`
`multicast-hbeat-src-addr = 0.0.0.0`
`multicast-hbeat-src-addr-mask = 0.0.0.0`

If you need more information about these settings, see the parameter descriptions in the *Stinger Reference*.

Overview of multiple MBONE configuration

The following parameters, shown with their default settings, are used to specify from MBONE interfaces:

```
[in IP-GLOBAL]
multicast-forwarding = no

[in IP-GLOBAL:multiple-mbone:mbone-profile]
mbone-profile[1] = ""
mbone-profile[2] = ""
mbone-profile[3] = ""
mbone-profile[4] = ""

[in IP-GLOBAL:multiple-mbone:mbone-lan-interface]
mbone-lan-interface[1] = { { any-shelf any-slot 0 } 0 }
mbone-lan-interface[2] = { { any-shelf any-slot 0 } 0 }
mbone-lan-interface[3] = { { any-shelf any-slot 0 } 0 }
mbone-lan-interface[4] = { { any-shelf any-slot 0 } 0 }

[in IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } ]
multicast-allowed = no

[in CONNECTION/"":ip-options]
multicast-allowed = no
```

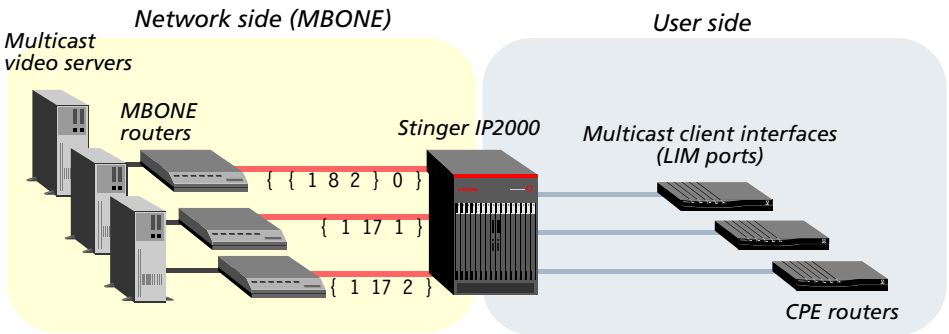
Parameter	Setting
`multicast-forwarding`	Enables or disables multicast forwarding. When you change the value to `yes` and write the profile, the multicast subsystem reads the values in the `ip-global` profile and initiates the forwarding function.

Parameter	Setting
`mbone-profile[N]`	Array of four indexed parameters for specifying the name of a local `connection` profile that provides access to an MBONE router across a trunk interface. This configures up to four WAN MBONE interfaces across ATM trunk ports. The total number of MBONE interfaces specified in either these parameters or the `mbone-lan-interface` parameters, or both, cannot exceed four.
`mbone-lan-interface[N]`	Array of four indexed parameters for specifying the index of an `ip-interface` profile that provides access to an MBONE router across an Ethernet interface. This configures up to four LAN MBONE interfaces across the Gigabit Ethernet port of the IP2000 controller or Ethernet interfaces of T1000 modules. For Stinger systems with redundant controllers, you can configure Gigabit Ethernet redundancy for a LAN MBONE interface to enable the system to maintain MBONE operations across a controller switchover. For details, see "Configuring a redundant LAN MBONE" on page 2-8. The total number of MBONE interfaces specified in either these parameters or the `mbone-profile` parameters, or both, cannot exceed four.
`multicast-allowed`	Enable/disable handling of IGMP requests and responses on the LAN (`ip-interface`) or trunk (`connection`) MBONE interface.

Sample configuration with multiple MBONE interfaces

The sample setup in Figure 8-3 shows three MBONE interfaces, one on the Gigabit Ethernet interface ({ { 1 8 2 } 0 }) and two additional MBONE interfaces on the two trunk interfaces in slot 17.

Figure 8-3. Sample configuration of multiple MBONE interfaces



The following commands configure a LAN MBONE interface on the Gigabit Ethernet port of the IP2000 controller:

```
admin> read ip-interface { { 1 8 2 } 0 }
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } read

admin> set ip-address = 1.1.1.2/28

admin> set multicast-allowed = yes

admin> write -f
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } written
```

The next commands configure two WAN MBONE interfaces on ATM trunk interfaces:

```
admin> new connection mcast1-17-1
CONNECTION/mcast1-17-1 read

admin> set active = yes

admin> set encapsulation-protocol = atm

admin> set ip-options remote-address = 3.3.3.3/29

admin> set ip-options multicast-allowed = yes

admin> set atm-options vci = 101

admin> set atm-options nailed-group = 801

admin write -f
CONNECTION/mcast1-17-1 written

admin> new connection mcast1-17-2
CONNECTION/mcast1-17-2 read

admin> set active = yes

admin> set encapsulation-protocol = atm

admin> set ip-options remote-address = 4.4.4.4/29

admin> set ip-options multicast-allowed = yes

admin> set atm-options vci = 101

admin> set atm-options nailed-group = 802

admin write -f
CONNECTION//mcast1-17-2 written
```

The following commands enable the multicast forwarding function and specify the MBONE interfaces:

```
admin> read ip-global
IP-GLOBAL read

admin> set multicast-forwarding = yes

admin> set multiple-mbone mbone-profile 1 = mcast1-17-1

admin> set multiple-mbone mbone-profile 2 = mcast1-17-2

admin> set multiple-mbone mbone-lan-interface 1 = { { 1 8 2 } 0 }

admin> write -f
IP-GLOBAL written
```

Managing multicast group memberships

To receive a multicast transmission, a client interface must join a specific multicast group (a Class D IP address from 224.0.0.0 to 239.255.255.255). When data is sent to an address in that range, it is multicast to all hosts that have joined that group.

The `mcast-service` profile provides a way to manage which multicast groups can be accessed by a client interface. Each profile specifies a number of multicast groups (up to 256), and a filter type. The filter type determines how the list of multicast groups is used: to allow access only to those groups, or allow access to all groups *except* those listed.

You can configure multiple `mcast-service` profiles, one for each level of multicast services you provide. You can define the profiles locally or via RADIUS. You then apply an `mcast-service` profile to a client interface by specifying the profile name in the client's `connection` or RADIUS profile (see "Configuring multicast client interfaces" on page 8-10).

Number of multicast clients per group

A Stinger IP2000 allows up to 1017 multicast clients per group. The following restrictions apply:

- Multicast groups from `224.0.0.100` through `224.0.0.160` are reserved for internal use.
- Multicast traffic is limited to a maximum of 542Mbps when the number of multicast clients in a group is greater than 251.

If the number of multicast clients in a group exceeds 1017, a message such as the following is displayed:

```
LOG info, Shelf 1, Controller-1, Time: 17:07:07--
  We hit maximum number of client per multicast group for 239.100.100.1
```

Overview of `mcast-service` settings

Following are the `mcast-service` settings, shown with default values:

```
[in MCAST-SERVICE/""]
service-name* = ""
active = no
snmp-trap-enable = no
filter-type = none
filter-list = [ { no 0.0.0.0 } { no 0.0.0.0 } { no 0.0.0.0 } { no 0.0.0.0 }+
[in MCAST-SERVICE/"":filter-list[1]]
active = no
mcast-ip-address = 0.0.0.0
```

Parameter	RADIUS attribute	Setting
`service-name`	Ascend-Multicast-Service-Name (276)	Name assigned to the service profile, up to 31 characters.

Parameter	RADIUS attribute	Setting
`active`	Ascend-Multicast-Service-Active (277)	Enables or disables the profile. *Note:* If the profile is disabled, none of the clients on the interfaces whose profiles point to this service profile are allowed to join any multicast groups. If you want the opposite effect, to allow clients on those interfaces access to all multicast groups, use the `filter-type` setting instead.
`snmp-trap-enable`	Ascend-Multicast-Service-Snmp-Trap (278)	Enables or disables sending a trap for a multicast link up or link down event for all client interfaces associated with this profile. To send these traps, you must set this parameter to `yes`. In addition, you must also enable the traps at the system level by setting the `ascend-multicast-link-trap-enabled` parameter in the `trap` profile. The objects reported in these traps are contained in the `mcastserv.mib`.
`filter-type`	Ascend-Multicast-Service-Filter-Type (279)	Specifies whether access to the multicast groups defined in the filter list will be filtered inclusively or exclusively. With inclusive filtering, client interfaces have access only to those groups specified in the filter list. With exclusive filtering, clients have access to all multicast groups *except* those in the list. If you set this to `none`, access to all multicast groups is allowed.
`filter-list[`*n*`]: active`	Ascend-Multicast-Filter-Active (280)	The filter list contains 256 indexed subprofiles, each of which specifies a multicast group address filter. The `active` parameter enables or disables the filter. When the filter is enabled, access to the address specified in the `mcast-ip-address` parameter is controlled as specified in the `filter-type` setting. If it is disabled, the filter has no effect.
`filter-list[`*n*`]: mcast-ip-address`	Ascend-Multicast-Filter-Address (281)	Class D IP address from 224.0.0.0 to 239.255.255.255. When data is sent to this address, it is multicast to all hosts that have joined that group.

Sample multicast service configurations

In this example, a multicast video server supports two multicast group addresses. Multicast clients can subscribe to either the "bronze" or the "gold" multicast service. Bronze service permits access to the group at 239.255.129.119. Gold service permits access to both 239.255.129.119 and a premium group at 239.255.129.120.

The following commands configure the `mcast-service` profiles:

```
admin> new mcast-service bronze-service
MCAST-SERVICE/bronze-service read

admin> set active = yes

admin> set snmp-trap-enable = yes

admin> set filter-type = inclusive
```

```
admin> set filter-list 1 active = yes
admin> set filter-list 1 mcast-ip-address = 239.255.129.119
admin> write -f
MCAST-SERVICE/bronze-service written
admin> new mcast-service gold-service
MCAST-SERVICE/gold-service read
admin> set active = yes
admin> set snmp-trap-enable = yes
admin> set filter-type = inclusive
admin> set filter-list 1 active = yes
admin> set filter-list 1 mcast-ip-address = 239.225.129.119
admin> set filter-list 2 active = yes
admin> set filter-list 2 mcast-ip-address = 239.255.129.120
admin> write -f
MCAST-SERVICE/gold-service written
admin> dir mcast-service
   802  07/24/2003 20:12:09  bronze-service
   809  07/24/2003 20:13:19  gold-service
```

Following are comparable RADIUS profiles:

```
mcastService-ipstinger-1 password = "pwd"
    Ascend-Multicast-Service-Name = "bronze-service"
    Ascend-Multicast-Service-Active = "Multicast-Service-Yes"
    Ascend-Multicast-Service-Snmp-Trap = "Multicast-Snmp-Trap-Yes"
    Ascend-Multicast-Service-Filter-Type = "Multicast-Filter-Inclusive"
    Ascend-Multicast-Filter-Address = "239.255.129.119"
mcastService-ipstinger-2 password = "pwd"
    Ascend-Multicast-Service-Name = "gold-service"
    Ascend-Multicast-Service-Active = "Multicast-Service-Yes"
    Ascend-Multicast-Service-Snmp-Trap = "Multicast-Snmp-Trap-Yes"
    Ascend-Multicast-Service-Filter-Type = "Multicast-Filter-Inclusive"
    Ascend-Multicast-Filter-Address = "239.255.129.119"
    Ascend-Multicast-Service-Snmp-Trap = "Multicast-Snmp-Trap-Yes"
    Ascend-Multicast-Service-Filter-Type = "Multicast-Filter-Inclusive"
    Ascend-Multicast-Filter-Address = "239.255.129.120"
```

Note You must set the Ascend-Multicast-Filter-Active attribute after each Ascend-Multicast-Filter-Address setting, because it increments the index for the next Ascend-Multicast-Filter-Address setting.

The following command displays information about the multicast service profiles:

```
admin> igmp profiles
IGMP Service Profiles

          Service Name               : gold-service
          SNMP Trap                  : Enabled
          Call logging               : Disabled
          Filter Type                : MCAST_FILTER_INCLUSIVE
```

```
Filter List                 :
       224.255.129.120
       224.225.129.119

Service Name                : bronze-service
SNMP Trap                   : Enabled
Call logging                : Disabled
Filter Type                 : MCAST_FILTER_INCLUSIVE
Filter List                 :
       224.255.129.119
```

Configuring multicast client interfaces

A multicast client interface is an IP-enabled ATM PVC across a DSL interface to a CPE router such as a CellPipe®.

Overview of multicast client `ip-options` settings

In addition to the ATM and IP options needed to configure a terminating PVC, the following parameters, shown with default values, are used to enable IP multicast on a client interface:

```
[in CONNECTION/"":ip-options]
multicast-allowed = no
multicast-rate-limit = 100
multicast-group-leave-delay = 0
multicast-group-leave-delay-msec = 0
multicast-service-profile = ""
multicast-max-groups = 0
```

Parameter	RADIUS attribute	Setting
`multicast-allowed`	Ascend-Multicast-Client (155)	Enable/disable handling of IGMP requests and responses on the interface. The system does *not* forward multicast traffic on the basis of this setting.
`multicast-rate-limit`	Ascend-Multicast-Rate-Limit (152)	The rate at which the Stinger IP2000 accepts multicast packets from clients on the interface. For example, if you set the rate to 5, the system accepts one packet every 5 seconds from multicast clients on the interface. Any subsequent packets received within that 5-second window are discarded. The default value of 100 disables multicast forwarding on the interface—the Stinger acts as a forwarder and handles IGMP packets, but does not accept packets from clients or forward multicast packets from the MBONE router. To enable multicast forwarding on the interface, you must set the rate to a number *less than* 100.

Parameter	RADIUS attribute	Setting
`multicast-group-leave-delay`	Ascend-Multicast-Grp-Leave-Delay (111)	Number of seconds to delay before forwarding a Leave Group message. The sum of (`multicast-group-leave-delay` × 1000) plus `multicast-group-leave-delay-msec` is the number of milliseconds the system waits before forwarding to the MBONE router an IGMP version-2 Leave Group message it receives across a multicast client interface. With the zero default values, the system forwards the Leave Group messages immediately. For client interfaces that support multiple multicast sessions to the same group, set these parameters to a value from 10 to 20. When these parameters have nonzero values, the system sends back a query to make sure there are no active multicast sessions on the interface for that group, and if it receives a response before the specified, specified delay expires, it does not forward the Leave Group message.
`multicast-group-leave-delay-msec`	Ascend-Multicast-GLeave-Delay-Msec	Number of milliseconds to add to the value of (`multicast-group-leave-delay` × 1000) to determine the total delay before forwarding a Leave Group message.
`multicast-service-profile`	Ascend-Multicast-Service-Profile-Name (274)	Name of a configured multicast services profile that filters multicast group access for this client interface. The filtering affects all new client links on the interface, and affects old client links after the expiration of `query-response-interval`. (See "Setting IGMP-v2 timers (local profiles only)" on page 8-11.) For information about configuring a multicast service profile, see "Managing multicast group memberships" on page 8-7.
`multicast-max-groups`	Ascend-Multicast-Max-Groups (275)	Maximum number of accessible multicast group (from 0 to 250) for this client interface. You can set this to a lower number to limit multicast traffic to the interface. This value of this parameter limits the activation of new client links, but does not affect the existing client links.

Setting IGMP-v2 timers (local profiles only)

The system calculates the timeout value for IGMPv1 and IGMPv2 members using the rule described in RFC 2236 *Internet Group Management Protocol, Version 2*. The rule is as follows:

`Timeout-value` = (`query-interval` x `robust-count` + `query-response-interval/10`)

You can control the timeout value by setting the `query-interval`, `robust-count`, or `query-response-interval` parameters for an IGMPv1 or IGMPv2 member.

The following parameters, shown with default values, are used to configure the timers defined in RFC 2236 on multicast client interfaces:

```
[in CONNECTION/"":ip-options:igmp-options]
robust-count = 2
```

```
query-interval = 125
query-response-interval = 100
last-member-query-interval = 10
last-member-query-count = 2
```

Parameter	Setting
`robust-count`	A threshold of packet losses (from 2 to 10) up to which the multicast subsystem will remain robust. If the interface is expected to have a high rate of packet loss, increase this value. IGMP is robust to (`robust-count` minus 1) packet losses. It cannot be set to zero and should not be set to 1. The default is 2.
`query-interval`	Number of seconds (from 0 to 1024) between general queries. You can increase this value from its default of 125 seconds to reduce the number of IGMP queries sent on the interface.
`query-response-interval`	Maximum response time in tenths of a second (from 0 to 1024) inserted into general queries. You can increase this value from its default of 10 seconds to make IGMP traffic less bursty, because host responses will be spread out over a larger interval. The number of seconds response time (this value divided by 10) must be less than the `query-interval` value.
`last-member-query-interval`	Maximum response time in tenths of a second (from 0 to 1024) inserted into group-specific queries sent in response to Leave Group messages. You can reduce this value from its default of 1 second to reduce the time it takes to detect that the last member of a group has left. The response time (this value divided by 10) must be less than the `query-interval` value.
`last-member-query-count`	Number of group-specific queries sent before the multicast router assumes there are no local members.

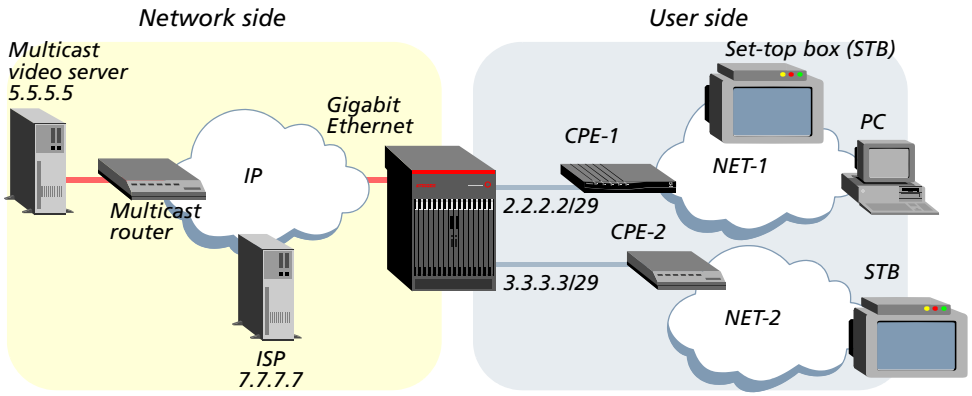
Sample multicast video configuration with filters

In this sample setup, the MBONE interface is the Gigabit Ethernet port of the IP2000. A multicast router on the local IP network sends video transmissions from a multicast video server. In the sample configurations for the setup shown in Figure 8-4, two types of filters are used:

- `mcast-service` profiles, to filter multicast groups and related services
- `filter` profiles, to filter out generic IP data

By introducing generic IP data filters, you can cause the system's IP interfaces to handle only multicast video services and block end users from accessing other services in the network.

Figure 8-4. DSL video application with a local MBONE interface



Configuring the local MBONE interface

The following commands enable the MBONE interface on the Gigabit Ethernet port:

```
admin> read ip-global
IP-GLOBAL read
admin> set multicast-forwarding = yes
admin> set mbone-lan-interface 1 = { { 1 8 2 } 0 }
admin> write -f
IP-GLOBAL written
```

Configuring multicast client PVCs

The profiles in this section apply `mcast-service` profiles to filter multicast groups and their related services. For background information, see "Managing multicast group memberships" on page 8-7.

The following commands configure a PVC for CPE-1 in Figure 8-4:

```
admin> new connection cpe-1
CONNECTION/cpe-1 read
admin> set active = yes
admin> set encapsulation-protocol = atm
admin> set ip-options remote-address = 2.2.2.2/29
admin> set ip-options multicast-allowed = yes
admin> set ip-options multicast-rate-limit = 20
admin> set ip-options multicast-service-profile = bronze-service
admin> set ip-options multicast-max-groups = 1
admin> set atm-options vci = 100
admin> set atm-options nailed-group = 51
admin> write -f
CONNECTION/cpe-1 written
```

The next set of commands configures the CPE-2 client interface in Figure 8-4:

```
admin> new connection cpe-2
CONNECTION/cpe-2 read
```

```
admin> set active = yes
admin> set encapsulation-protocol = atm
admin> set ip-options remote-address = 3.3.3.3/29
admin> set ip-options multicast-allowed = yes
admin> set ip-options multicast-rate-limit = 20
admin> set ip-options multicast-service-profile = gold-service
admin> set ip-options multicast-max-groups = 2
admin> set atm-options vci = 100
admin> set atm-options nailed-group = 52
admin> write -f
CONNECTION/cpe-2 written
```

Following are comparable RADIUS profiles :

```
permconn-st-2 Password = ascend
    Service-Type = Framed-User,
    Framed-Protocol = ATM-1483,
    User-Name = "cpe-1",
    Ascend-Route-IP = Route-IP-Yes,
    Framed-IP-Address = 2.2.2.2,
    Framed-IP-Netmask = 255.255.255.248,
    Ascend-Multicast-Client = Multicast-Yes,
    Ascend-Multicast-Rate-Limit = 20
    Ascend-Multicast-Service-Profile-Name = "bronze-service"
    Ascend-Multicast-Max-Groups = 1
    Ascend-ATM-Group = 51,
    Ascend-ATM-Vpi = 0,
    Ascend-ATM-Vci = 100

permconn-st-2 Password = ascend
    Service-Type = Framed-User,
    Framed-Protocol = ATM-1483,
    User-Name = "cpe-2",
    Ascend-Route-IP = Route-IP-Yes,
    Framed-IP-Address = 3.3.3.3,
    Framed-IP-Netmask = 255.255.255.248,
    Ascend-Multicast-Client = Multicast-Yes,
    Ascend-Multicast-Rate-Limit = 20
    Ascend-Multicast-Service-Profile-Name = "gold-service"
    Ascend-Multicast-Max-Groups = 2
    Ascend-ATM-Group = 52,
    Ascend-ATM-Vpi = 0,
    Ascend-ATM-Vci = 100
```

Applying a filter that restricts the GigE interface to video traffic only

In this example, a filter is applied to the GigE interface to prevent the interface from handling traffic other than video data and related control messages. This filter has the advantage of being easy to define and apply, but it restricts the GigE interface from being used for other applications.

For background information about filters, see Chapter 10, “Using IP Filters.”

The following commands create a new `filter` profile named `mcast-only`, and specify an input filter that forwards inbound multicast data traffic (239.100.100.0):

```
admin> new filter mcast-only
FILTER/mcast-only read
admin> set input-filters 1 valid-entry = yes
admin> set input-filters 1 forward = yes
admin> set input-filters 1 Type = ip-filter
admin> set input-filters 1 ip-filter source-address-mask = 255.255.255.0
admin> set input-filters 1 ip-filter source-address = 239.100.100.0
```

The following commands specify an output filter that allows client access to the video server at IP address 5.5.5.5 (Figure 8-4). Clients might require access to the server, for example, to download boot information for a set-top box.

```
admin> set output-filters 1 valid-entry = yes
admin> set output-filters 1 forward = yes
admin> set output-filters 1 Type = ip-filter
admin> set output-filters 1 ip-filter dest-address-mask = 255.255.255.255
admin> set output-filters 1 ip-filter dest-address = 5.5.5.5
```

Because the default setting of the `forward` parameter is `no`, the next set of commands explicitly drops all other output traffic:

```
admin> set output-filters 2 valid-entry = yes
admin> set output-filters 2 Type = ip-filter
admin> write -f
FILTER/mcast-only written
```

The following commands apply the `mcast-only` filter to the Gigabit Ethernet interface of the IP2000:

```
admin> read ethernet { 1 8 2 }
ETHERNET/{ shelf-1 first-control-module 2 } read
admin> set filter-name = mcast-only
admin> write -f
ETHERNET/{ shelf-1 first-control-module 2 } written
```

An alternative filter to restrict each client interface

If you must use the Gigabit Ethernet interface for other applications as well as multicast video, you cannot restrict the type of traffic allowed on the interface. In that case, you can define filters for individual multicast client interfaces, to restrict the those interfaces from handling traffic other than video data and related control messages. Individual users on the client interface will be unable to access other services through the Stinger.

Note The IP2000 does not currently support filters specified in RADIUS profiles.

This sample filter is specific to the multicast client network connected to CPE-1 in Figure 8-4 (page 8-13), with the CPE IP address 2.2.2.2/29.

The following commands create a new `filter` profile named `conn-input-filter`, and specify an input filter that forwards inbound traffic to the video server at IP address

5.5.5.5 (Figure 8-4). Clients might require access to the server, for example, to download boot information for a set-top box.

```
admin> new filter conn-input-filter
FILTER/conn-input-filter read

admin> set input-filters 1 valid-entry = yes

admin> set input-filters 1 forward = yes

admin> set input-filters 1 Type = ip-filter

admin> set input-filters 1 ip-filter dest-address-mask = 255.255.255.255

admin> set input-filters 1 ip-filter dest-address = 5.5.5.5
```

The next set of commands specifies an input filter that allows only inbound traffic that uses the IGMP protocol (protocol number 2) from a client on the subnet 2.2.2.*x*.

```
admin> set input-filters 2 valid-entry = yes

admin> set input-filters 2 forward = yes

admin> set input-filters 2 Type = ip-filter

admin> set input-filters 2 ip-filter protocol = 2

admin> set input-filters 2 ip-filter source-address-mask = 255.255.255.0

admin> set input-filters 2 ip-filter source-address = 2.2.2.0
```

Because the default setting of the `forward` parameter is `no`, the next set of commands explicitly drops all other inbound traffic from the client subnet:

```
admin> set input-filters 3 valid-entry = yes

admin> set input-filters 3 Type = ip-filter

admin> write -f
FILTER/conn-input-filter written
```

For the CPE-1 `connection` profile definition, see "Configuring multicast client PVCs" on page 8-13. The following commands apply this filter to the individual client interface for CPE-1:

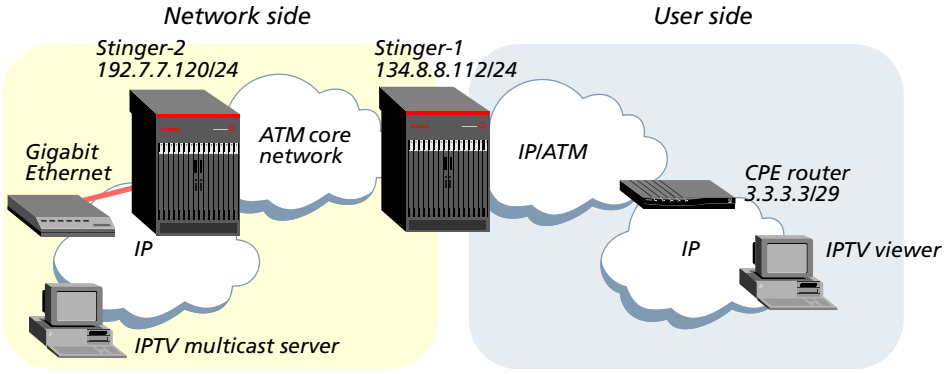
```
admin> read connection cpe-1
CONNECTION/cpe-1 read

admin> set session-options data-filter = conn-input-filter

admin> write -f
CONNECTION/cpe-1 written
```

Sample multicast video configuration with a remote MBONE interface

In this sample setup shown in Figure 8-5, the MBONE interface is configured in Stinger-2, and the multicast client interface is configured in Stinger-1.

The connection between the two Stinger units is an ATM PVC. It can use any ATM medium, and does not use IP processing. In this example, Stinger-2 sends the IPTV multicast data stream across the ATM cloud as a cell stream. Stinger-1 forwards the ATM stream to the multicast CPE router on the basis of an ATM circuit configuration.

Figure 8-5. IPTV video sample configuration



Stinger-1 requires an ATM circuit profile between the LIM interface to the CPE router, and the trunk interface to Stinger-2.

The following command on Stinger-1 displays the nailed-group number of the ADSL interface (LIM slot 4, port 5) to the CPE router in Figure 8-5:

```
admin> which -n { 1 4 5 }
Nailed group corresponding to port { shelf-1 slot-4 5 } is 155
```

The following command on Stinger-1 displays the nailed-group number of the OC3-ATM interface (trunk slot 17, port 2) that connects to the ATM core network in Figure 8-5:

```
admin> which -n { 1 17 2 }
Nailed group corresponding to port { shelf-1 trunk-module-1 2 } is 802
```

The following set of commands on Stinger-1 configures an ATM circuit between the two interfaces:

```
admin> new connection mcast-client-pvc
CONNECTION/mcast-client-pvc read
admin> set active = yes
admin> set encapsulation-protocol = atm-circuit
admin> set ip-options ip-routing-enabled = no
admin> set atm-options nailed-group = 155
admin> set atm-connect-options vci = 100
admin> set atm-connect-options nailed-group = 802
admin> write -f
CONNECTION/mcast-client-pvc written
```

Following is a comparable RADIUS profile:

```
permconn-st-2 Password = "pwd"
    Service-Type = Outbound,
    Framed-Protocol = ATM-CIR,
    User-Name = "mcast-client-pvc",
    Ascend-ATM-Group = 155,
    Ascend-Route-IP = Route-IP-No,
    Ascend-ATM-Vpi = 0,
    Ascend-ATM-Vci = 35,
```

```
    Ascend-ATM-Connect-Vpi = 0,
    Ascend-ATM-Connect-Vci = 100,
    Ascend-ATM-Connect-Group = 802
```

With this `connection` or RADIUS profile, ATM cells received by Stinger-1 from the CPE router are switched to the unit's trunk interface and transmitted across the ATM cloud. On Stinger-2, the PVC terminates and is packetized for transmission on the IP network.

The following command on Stinger-2 displays the nailed-group number of a DS3-ATM interface (trunk slot 17, port 1) that connects to the ATM core network in Figure 8-5:

```
admin> which -n { 1 17 1 }
Nailed group corresponding to port { shelf-1 trunk-module-1 1 } is 801
```

The following set of commands on Stinger-2 terminates the PVC on a trunk interface with VCI 100, and specifies the required IP address for Stinger-1.

```
admin> new connection term-18-100
CONNECTION/term-18-100 read

admin> set active = yes

admin> set encapsulation-protocol = atm

admin> set ip-options remote-address = 3.3.3.3/29

admin> set atm-options vci = 100

admin> set atm-options nailed-group = 801

admin> write -f
CONNECTION/term-18-100 written
```

Following is a comparable RADIUS profile:

```
permconn-st-1 Password = "pwd"
    Service-Type = Outbound,
    Framed-Protocol = ATM-1483,
    User-Name = "term-18-100",
    Framed-IP-Address = 3.3.3.3,
    Framed-IP-Netmask = 255.255.255.252,
    Ascend-ATM-Group = 801,
    Ascend-Route-IP = Route-IP-Yes,
    Ascend-ATM-Vpi = 0,
    Ascend-ATM-Vci = 100
```

After completing the configuration, starting the IPTV client software on the multicast client interface should create the 239.100.100.4 multicast group on Stinger-2. The following command on Stinger-2 checks that the group exists:

```
admin> igmp groups
Group Address          Members     Expire time      Counts
230.0.0.9              14          00:00:31      0 :: 0 S2
                       *(Mbone)                  0 :: 0 S2
```

The following command displays client interfaces (interface 14 representing the remote client interface):

```
admin> igmp clients
   IGMP Clients
    Client             Version        RecvCount      CLU    ALU
```

```
1(Mbone)        2               0       0       0
14              2               0       0       0
```

Administrative tools for IGMP operations

The system supports the `igmp` commands for administrative information about IGMP multicast operations. For examples, see "IGMP diagnostics" on page A-4.

Protocol Independent Multicast Sparse Mode (PIM-SM v2)

9

Traditional multicast routing mechanisms, such as Distance Vector Multicast Routing Protocol (DVMRP) or Multicast OSPF (MOSPF) are intended for use in regions where groups are widely represented or bandwidth is universally plentiful. If these traditional schemes are used when multicast receivers and senders are distributed sparsely across a wide area, data packets and membership reports are sent over many links that do not lead to receivers or senders. For this reason, the traditional schemes can be inefficient for use in wide area networks.

PIM-SM is designed to operate efficiently across wide area networks, where groups are sparsely distributed. Each multicast group has a shared tree through which receivers learn about ("rendezvous with") sources. The rendezvous point (RP) is the root of this per-group shared tree. PIM SM uses RPs and explicit join/prune messages instead of the broadcast and prune mechanism used by PIM-Dense Mode or DVMRP.

This implementation of PIM-SM follows the current IETF drafts for PIM-SM v2 (`draft-ietf-pim-sm-v2-new-07.txt`, March 2003) and candidate bootstrap router (C-BSR) functionality (`draft-ietf-pim-sm-bsr-03.txt`, February 2003).

Note Stinger units support full interoperability with RFC 2362-compliant multicast routers that have not yet implemented the PIM-SM v2 IETF draft recommendations. However, Stinger units do not support interoperability with systems running PIM-SM v1. In addition, PIM-SM cannot be used with multiple MBONE interfaces in a Stinger unit.

PIM-SM features supported with this software version

With this software version, PIM-SM is supported on IP2000 Gigabit Ethernet and trunk interfaces. Stinger units cannot currently operate as PIM Multicast Border Router (PMBR). With the current software, Stinger units support PIM-SM functionality as shown in Table 9-1:

Table 9-1. Current level of support for PIM-SM functionality

PIM-SM capability	Support in this software version
PIM-SM general purpose states	(*,G) state
PIM join/prune messages	Join/prune messages for the (*,G) PIM-SM state
Bootstrap router (BSR)	Stinger units can be configured to act as C-BSR and take part in the BSR election process.
	If the Stinger becomes the elected BSR, it sends bootstrap messages (BSMs) to 224.0.0.13 on its PIM-enabled interfaces.
	If a PIM neighbor comes up and Stinger is the designated router on that interface, Stinger sends the recent BSM to the neighbor.
Designated routers (DR) and hello messages	On each PIM-enabled interface, the Stinger unit can be elected DR on the basis of hello priority.
	Currently, Stinger units do not support register messages and do not have the capability of processing IGMP messages on the LAN interface. For that reason, the Stinger unit must not be elected DR on the LAN interface if the LAN supports IGMP hosts.
RP-group mapping	Static configuration of active group range and their respective RPs.
	Dynamic RP-group mapping using the BSR router mechanism.
Data packet forwarding	Switch to SPT is not supported.
	Data packets are not forwarded from one PIM interface to another. Data packets flow from PIM interfaces to the users only.

Overview of PIM-SM configuration

To enable the system to act as a multicast router in a PIM domain, you must complete the following steps:

1 Enable multicast forwarding and the PIM protocol. If appropriate, specify a C-BSR configuration for the Stinger to participate in BSR elections and act as BSR if elected. See "Enabling multicast and PIM" on page 9-3.

2 Configure static mappings between multicast groups and PIM RPs. This is recommended as a failsafe configuration. See "Configuring static mappings between groups and rendezvous points" on page 9-5.

3 Configure an `ip-interface` or `connection` profile, to enable the system to operate as a PIM router on the Gigabit Ethernet interface, a trunk interface, or both. See "Configuring PIM on the Gigabit Ethernet or trunk interface" on page 9-6.

For examples that show a system configuration that includes all three steps, see "Sample PIM-SM system configuration" on page 9-10.

Enabling multicast and PIM

A PIM BSR is a dynamically elected router within a PIM domain that is responsible for constructing the set of RPs and originating BSMs. A C-BSR is a PIM router configured to participate in the BSR election and to act as BSR if elected. One BSR is elected per PIM domain on the basis of highest priority and address.

For details about the `pim bsr` and `pim rp` commands, see "Administrative tools for PIM-SM routing" on page 9-12.

Overview of settings in the `ip-global` profile

Following are the parameters, shown with default values, for enabling PIM-SM and configuring its BSR capabilities in a Stinger unit:

```
[in IP-GLOBAL]
multicast-forwarding = no

[in IP-GLOBAL:pim-options]
enable = no
cbsr-enable = no
cbsr-ip-address = 0.0.0.0
cbsr-priority = 0
cbsr-interval = 60
```

Parameter	Setting
`multicast-forwarding`	Enables or disables multicast forwarding. This parameter must be set to `yes` for PIM-SM operations. When you change the value to `yes` and write the profile, the multicast subsystem reads the values in the `ip-global` profile and initiates the forwarding function.
`enable`	Enables or disables the PIM routing protocol systemwide. This parameter and the `multicast-forwarding` parameter must be set to `yes` to enable PIM.
`cbsr-enable`	Enables or disables the BSR router mechanism. When set to `yes`, the Stinger unit acts as a candidate BSR and takes part in electing a BSR in the PIM domain. With the `yes` setting, you must specify a `cbsr-ip-address` value. This setting is not used when `enable` is set to `no`.
`cbsr-ip-address`	Local IP address the Stinger unit uses to send BSMs when `cbsr-enable` is set to `yes`. This setting is not used when `cbsr-enable` is set to `no`.
`cbsr-priority`	BSR priority for the Stinger, from 0 (the default) to 255. The priority is used in the election of BSR. The system is more likely to be elected BSR with a higher priority value. To enable the system to exchange BSMs without becoming BSR, leave the default zero setting, or set a low numeric value. This setting is not used when `cbsr-enable` is set to `no`.

Parameter	Setting
cbsr-interval	Number of seconds, from 5 to 900, between transmission of BSMs. The default is to send BSMs every 60 seconds. This setting is not used when cbsr-enable is set to no.

Example showing BSR election and dynamic group-RP mappings

The following commands configure the Stinger unit to act as C-BSR using the IP address of the Gigabit Ethernet interface (1.1.1.101 in this example):

```
admin> read ip-global
IP-GLOBAL read

admin> set multicast-forwarding = yes

admin> set pim-options enable = yes

admin> set pim-options cbsr-enable = yes

admin> set pim-options cbsr-ip-address = 1.1.1.101

admin> write -f
IP-GLOBAL written
```

The following command displays the BSR status immediately after writing the ip-global profile:

```
admin> pim bsr
Stinger BSR State : PENDING_BSR
Details of CURRENT BSR:
BSR IP Address       : 0.0.0.0
BSR Interface        : 0
BSR Priority         : 0
BSR holdtime         : 0
BSR Current Frag Tag : 0
BSR HASH masklen     : 0
```

After the bootstrap interval has elapsed and the system has received a BSM from another router in the domain, a repeat of the pim bsr command shows that the Stinger has become candidate BSR. For example:

```
admin> pim bsr
Stinger BSR State : CANDIDATE_BSR
Details of CURRENT BSR:
BSR IP Address       : 1.1.1.10
BSR Interface        : 1
BSR Priority         : 110
BSR holdtime         : 112
BSR Current Frag Tag : 0
BSR HASH masklen     : 30
```

The following commands modify the ip-global profile to specify the highest BSR priority for the Stinger system:

```
admin> set cbsr-priority = 255

admin> write -f
IP-GLOBAL written
```

Following an exchange of BSMs, the Stinger is elected BSR. For example:

```
admin> pim bsr
Stinger BSR State : ELECTED_BSR
Details of CURRENT BSR:
BSR IP Address       : 1.1.1.101
BSR Interface        : 1
BSR Priority         : 255
BSR holdtime         : 57
BSR Current Frag Tag : 717
BSR HASH masklen     : 30
```

The system is also receiving group-RP mappings dynamically from other routers. For example, the following command displays both static (if any) and dynamic group-RP mappings:

```
admin> pim rp
Group           RP-Address      RPF neighbor    Priority   holdtime
224.0.0.0/8       1.1.1.3         -                 40        75:73
234.0.0.0/8       1.1.1.3         -                 40        75:73
234.3.0.0/16      1.1.1.3         -                 40        75:73
234.4.0.0/16      1.1.1.3         -                 40        75:73
234.5.0.0/16      1.1.1.3         -                 40        75:73
234.6.0.0/16      1.1.1.3         -                 40        75:73
Static Entries:
225.0.0.0/8       1.1.1.10        1.1.1.10          -         -
235.0.0.0/8       1.1.1.10        1.1.1.10          -         -
```

Configuring static mappings between groups and rendezvous points

For PIM-SM to operate properly, all routers in the domain must share the same set of group-to-RP mappings. You can statically configure associations between groups and RPs by using the `pim-group-rp-mapping` profile. These static mappings provide a basic interoperability mechanism if the automatic methods of obtaining mappings should fail.

Each `pim-group-rp-mapping` profile specifies a mapping between an RP (specified as a reachable IP address) and a range of multicast groups (specified as a group and mask). The system uses these mappings to determine an RP for a given group.

Following are the parameters, shown with their default settings, for configuring a static group-to-RP mapping:

```
[in PIM-GROUP-RP-MAPPING/""]
name* = ""
rp-address = 0.0.0.0
group-address = 0.0.0.0/0
group-mask = 0.0.0.0
```

Parameter	Setting
`name`	Text string, up to 31 characters long, that names the mapping between a multicast group and the IP address of a rendezvous point (RP).

Parameter	Setting
rp-address	IP address of the RP. The address must be reachable throughout the domain.
group-address	A multicast group address (a class D IP address). You can specify a full group address or a group range. If you specify a prefix (such as /8 in the value 226.0.0.0/8), the system automatically updates the group-mask parameter with the appropriate decimal value (such as 255.0.0.0). The combined group address and group mask must be unique in the system. You cannot write duplicate mappings for the same group or group range.
group-mask	A mask to be applied to the group-address value to obtain the group prefix mapped to the specified RP. For example, a value of 255.0.0.0 indicates a one-octet group prefix. If no mask is specified, the default mask of 255.255.255.255 is applied.

The following commands create a static mapping for multicast group 231.1.1.1:

```
admin> new pim-group-rp-mapping 231
PIM-GROUP-RP-MAPPING/231 read

admin> set rp-address = 1.1.1.3

admin> set group-address = 231.1.1.1

admin> write -f
PIM-GROUP-RP-MAPPING/231 written
```

The next command shows the static mapping in the group-RP set for the system:

```
admin> pim rp
Group           RP-Address      RPF neighbor    Priority   holdtime
Static Entries:
231.1.1.1/32    1.1.1.3         -               -           -
```

Configuring PIM on the Gigabit Ethernet or trunk interface

The pim-options subprofiles in an ip-interface or connection profile specify settings to enable PIM-SM, and specify the handling of PIM hello and join/prune messages on the interface.

All PIM routers send hello messages periodically on each PIM-enabled interface, and record the hello information received from each PIM neighbor. Hello messages allow a router to learn about the neighboring PIM routers on the interface, and the priority field in these messages is used in DR election on the LAN interface. DR election is not required on point-to-point links, such as on trunk interfaces, so Hello priority is not sent on a trunk interface even if it is configured.

PIM options in the **ip-interface** and **connection** profiles

Following are the parameters, shown with default values, for enabling PIM on the Gigabit Ethernet interface:

```
[in IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 }]
multicast-allowed = no

[in IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 }:pim-options]
enable = no
hello-interval = 30
hello-holdtime = 105
hello-priority-option = yes
hello-priority = 1
join-prune-interval = 60
join-prune-holdtime = 210
lan-delay-option = yes
lan-delay = 5000
override-interval = 2500
```

Note You cannot enable PIM on a virtual Ethernet interface.

Following are the parameters, shown with default values, for configuring PIM on an RFC 1483, *Multiprotocol Encapsulation over ATM Adaptation Layer 5* (MPOA), terminating `connection` profile on a trunk interface:

```
[in CONNECTION/""]
encapsulation-protocol = atm-circuit

[in CONNECTION/"":atm-options]
atm1483type = aal5-llc
vpi = 0
vci = 35
nailed-group = 1

[in CONNECTION/"":ip-options]
ip-routing-enabled = yes
remote-address = 0.0.0.0/0
local-address = 0.0.0.0
multicast-allowed = no

[in CONNECTION/"":ip-options:pim-options]
enable = no
hello-interval = 30
hello-holdtime = 105
hello-priority-option = yes
hello-priority = 1
join-prune-interval = 60
join-prune-holdtime = 210
lan-delay-option = yes
lan-delay = 5000
override-interval = 2500
```

Parameter	Setting
`encapsulation-protocol`	Set to `atm` for MPOA terminating connections.
`atm1483type`	Method of multiplexing Layer-3 packets into ATM cells (AAL5-LLC or AAL5-VC). For details, see the *Stinger ATM Configuration Guide*.

Parameter	Setting
`vpi`	VPI for the MPOA terminating PVC. For details, see the *Stinger ATM Configuration Guide*.
`vci`	VCI for the MPOA terminating PVC. For details, see the *Stinger ATM Configuration Guide*.
`nailed-group`	Nailed-group number of the trunk interface. For details, see the *Stinger ATM Configuration Guide*.
`ip-routing-enabled`	IP routing must be enabled (as it is by default) for MPOA terminating connections.
`remote-address`	IP address of the remote device, which can include a subnet specification. If it does not include a subnet mask, the router software in the Stinger unit assumes a default subnet mask that is based on address class.
`local-address`	IP address assigned to the local side of a numbered-interface connection. This is a requirement for PIM-enabled trunk interfaces.
`multicast-allowed`	Must be set to `yes` for PIM-enabled interfaces.
`enable`	Enables or disables the PIM routing protocol on the interface.
`hello-interval`	Number of seconds between sending hello messages to PIM neighbors on this interface. The valid range is from 1 to 65535 with a default value of 30 seconds. The value must be less than that of the `hello-holdtime` parameter.
`hello-holdtime`	Number of seconds a receiver of hello messages must consider the sender reachable before timing out the sender. The valid range is from 1 to 65535 with a default value of 105 seconds. The value must be greater than that of the `hello-interval` parameter.
`hello-priority-option`	Whether the Stinger unit will participate in DR election on this interface (`yes` or `no`, with a default value of `yes`). Hello-priority is not sent in the hello messages over a trunk interface, even if it is configured, because DR election is not carried out on a point-to-point link.
`hello-priority`	DR election priority for the Stinger unit on the interface. The DR election priority is a 32-bit unsigned number contained in a hello message. A router with a numerically larger priority is preferred in electing a new DR. The valid range for this setting is from 0 to 4,294,967,295, with a default setting of 1. This value is not sent on a trunk interface, even if it is configured, because DR election is not carried out on a point-to-point link. **Note** Currently, the Stinger unit must not be elected DR on the LAN interface if the LAN supports IGMP hosts.

Parameter	Setting
`join-prune-interval`	Number of seconds between sending PIM join/prune messages to PIM neighbors on this interface. A join/prune message consists of a list of groups and a list of joined and pruned sources for each group. The valid range is from 1 to 65535 with a default value of 60 seconds. The value must be less than that of the `join-prune-holdtime` parameter. **Note** Stinger units do not currently support the (S,G) state, so it always sends (*,G) join/prune messages.
`join-prune-holdtime`	Number of seconds a receiver of join/prune messages must consider the list valid before timing out the information. The valid range is from 1 to 65535 with a default value of 210 seconds. The value must be greater than that of the `join-prune-interval` parameter.
`lan-delay-option`	Whether the Stinger unit will expect propagation delay over an Ethernet interface (`yes` or `no`, with a default value of `yes`). The `lan-delay` option is not sent in hello messages on a trunk interface, even if it is configured, because it applies only to LAN interfaces.
`lan-delay`	Number of milliseconds of expected propagation delay over the Ethernet interface. The valid range is from 1 to 65535 with a default value of 5000 milliseconds. This value is not sent in hello messages on a trunk interface, even if it is configured, because it applies only to LAN interfaces.
`override-interval`	A delay interval, in milliseconds, used to randomize when scheduling a delayed join message. The valid range is from 1 to 65535 with a default value of 2500 milliseconds. Because the `override-interval` is sent along with `lan-delay`, this value is not sent in hello messages on a trunk interface, even if it is configured.

Example of enabling PIM on the Gigabit Ethernet interface

The following commands enable PIM on the IP2000 Gigabit Ethernet interface:

```
admin> read ip-interface { { 1 8 2 } 0 }
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } read
admin> set ip-address = 1.1.1.2
admin> set multicast-allowed = yes
admin> set pim-options enable = yes
admin> write -f
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } written
```

Example of enabling PIM on a trunk interface

The following commands enable PIM on an MPOA terminating PVC on a trunk port. A numbered interface (an IP interface that specifies both a remote and local IP

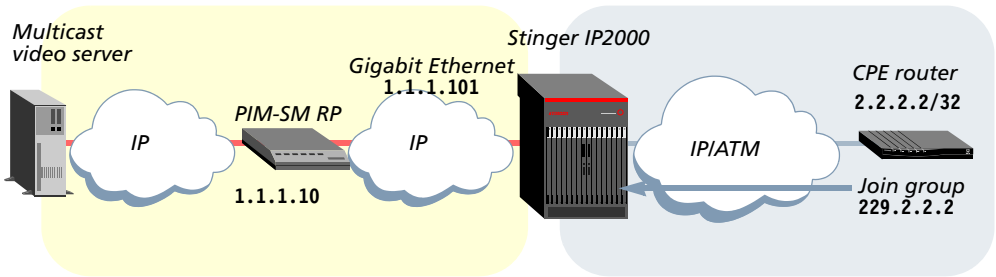
address) is required. For details about numbered interfaces, see "Example of a numbered interface" on page 4-34.

```
admin> new connection pim-trunk
CONNECTION/pim-trunk read
admin> set active = yes
admin> set encapsulation-protocol = atm
admin> set ip-options remote-address = 123.123.123.2/24
admin> set ip-options local-address = 123.123.123.1/24
admin> set ip-options multicast-allowed = yes
admin> set ip-options pim-options enable = yes
admin> set telco-options nailed-groups = 851
admin> set mp-options enabled = no
admin> set atm-options vci = 38
admin> set atm-options nailed-group = 851
admin> write -f
CONNECTION/pim-trunk written
```

Sample PIM-SM system configuration

For details about multicast client configuration, see "Configuring multicast client interfaces" on page 8-10. In the sample setup shown in Figure 9-1, PIM-SM is enabled on the IP2000 Gigabit Ethernet port.

Figure 9-1. PIM-SM on Gigabit Ethernet and trunk interface



The following commands enable PIM-SM and configure the Stinger unit to act as C-BSR using the IP address of the Gigabit Ethernet interface (1.1.1.101 in this example):

```
admin> read ip-global
IP-GLOBAL read
admin> set multicast-forwarding = yes
admin> set pim-options enable = yes
admin> set pim-options cbsr-enable = yes
admin> set pim-options cbsr-ip-address = 1.1.1.101
admin> write -f
IP-GLOBAL written
```

The following commands configure a PVC for multicast client CPE router in Figure 9-1:

```
admin> new connection mcast-client
CONNECTION/mcast-client read

admin> set active = yes

admin> set encapsulation-protocol = atm

admin> set ip-options remote-address = 2.2.2.2/32

admin> set ip-options multicast-allowed = yes

admin> set ip-options multicast-rate-limit = 20

admin> set atm-options vci = 100

admin> set atm-options nailed-group = 51

admin> write -f
CONNECTION/mcast-client written
```

The following commands enable multicast and PIM-SM on the IP2000 Gigabit Ethernet interface:

```
admin> read ip-interface { { 1 8 2 } 0 }
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } read

admin> set ip-address = 1.1.1.101

admin> set multicast-allowed = yes

admin> set pim-options enable = yes

admin> write -f
IP-INTERFACE/{ { shelf-1 first-control-module 2 } 0 } written
```

The following commands configure a group-to-RP mapping specifying the RP shown in Figure 9-1:

```
admin> new pim-group-rp-mapping 229
PIM-GROUP-RP-MAPPING/229 read

admin> set rp-address = 1.1.1.10

admin> set group-address = 229.0.0.0/8

admin> write -f
PIM-GROUP-RP-MAPPING/229 written
```

Following is the relevant group mapping:

```
admin> pim rp
Group           RP-Address      RPF neighbor    Priority   holdtime
Static Entries:
229.0.0.0/8     1.1.1.10        -               -           -
```

The following command shows the PIM neighbor across the Gigabit Ethernet interface:

```
admin> pim nbr
Neighbor        Interface    Priority     Holdtime        DR
1.1.1.10            1         100         105:96          No
```

Administrative tools for PIM-SM routing

The `pim` command displays PIM-related information for active PIM-enabled interfaces in the system. For details, see "PIM-SM diagnostics" on page A-10.

The system provides SNMP MIB support for the PIM protocol as defined in `draft-ietf-pim-mib-v2-01.txt` (the PIMv2 MIB). The PIMv2 MIB is placed in the MIB tree under `experimental 61`. For details, see "PIMv2 MIB support" on page A-29.

In addition, the output of the `netstat -s` command now includes the total PIM statistics for all PIM-enabled interfaces in the system. For example:

```
admin> netstat -s
...
pim:
     25 packets received
     24 hello packet received
     1 C-RP packets received
     38 packets transmitted
     26 hello packets sent
     12 Bootstrap packets sent
```

Using IP Filters

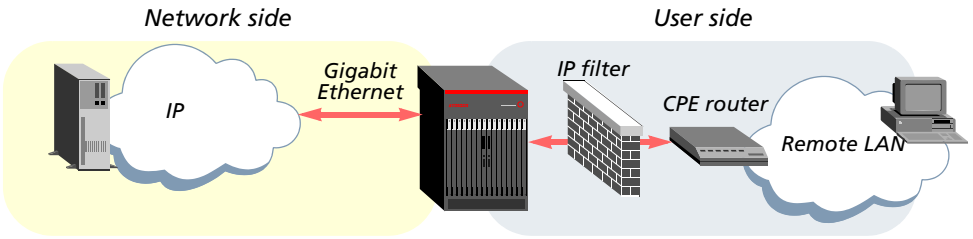
10

Because the optional T1000 module supports a wider range of filtering options than the IP2000 control module, some parameters in the `filter` profile are visible even though they are not used by the IP2000. In particular, generic filters, route filters, and type-of-service (ToS) filters are not used for connections terminating on the IP2000. Those parameters are documented in the *Stinger Reference*.

An IP filter describes IP and related packet fields (such as protocol numbers, logical addresses, and TCP or UDP ports) and specifies whether to forward or drop packets that match the description. A single filter can be applied to many interfaces.

After you apply a filter to an interface, the system monitors the data stream on that interface and determines which traffic is allowed to pass into the system. For example, Figure 10-1 shows a filter applied to the IP interface for a CPE device.

Figure 10-1. IP filter on CPE interface



You can define a filter to monitor inbound packets, outbound packets, or both. The action of forwarding or dropping packets can apply to packets that match the specifications, or to all packets *except* those that match the specifications.

How IP filters work

A `filter` profile can include up to 12 input filters and 12 output filters, each of which specifies a set of comparisons that are made in a defined order, and its own forwarding action—forward or drop. The filters are applied in sequence. The filtering

process stops immediately if conditions in a packet match a single filter specification, and the forwarding action in that filter is applied to the packet. The filtering subsystem uses an implicit default rule at the end of the filtering process to drop all packets that do not match the specified input and output rules.

If a filter specifies rules only in one direction, no filtering is applied to traffic in the other direction. For example, if a filter specifies only input filter rules, all output packets are forwarded without any filtering.

For information about configuring an explicit default rule to forward all packets that do not match the specified input and output rules, see"Explicit default filter rules" on page 10-5.

Overview of `ip-filter` settings

Note Only the `ip-filter` type is supported for the IP2000, and only the parameters in the `ip-filter` subprofile are applicable. For details about other settings, which may be supported on an optional T1000 module, see the *Stinger Reference*. The system does not prevent you from configuring other types of filters, but it displays a warning message if you do. Those filters will have no effect.

Following are the `filter` profile settings for defining IP filters. The parameters are shown with their default values for input filters. The same values apply for output filter specifications—setting the parameters in an input filter affects the inbound data stream, and setting them in an output filter affects the outbound data stream.

```
[in FILTER/""]
filter-name* = ""

[in FILTER/"":input-filters[1]]
valid-entry = no
forward = no
type = gen-filter

[in FILTER/"":input-filters[1]:ip-filter]
protocol = 0
source-address-mask = 0.0.0.0
source-address = 0.0.0.0
dest-address-mask = 0.0.0.0
dest-address = 0.0.0.0
Src-Port-Cmp = none
source-port = 0
Dst-Port-Cmp = none
dest-port = 0
tcp-estab = no
```

Parameter	Setting
`filter-name`	Name of the filter profile, up to 36 characters. You apply a filter to an interface by referring to this name.
`valid-entry`	Enable/disable the input or output filter. With a setting of `no` (the default), the system skips this specification when filtering the data stream. Set this parameter to `yes` for each defined filter you intend to use.

Parameter	Setting
`forward`	Forwarding action for the filter. The default value of `no` causes the system to discard matching packets.
`type`	Type of filter. Only `ip-filter` is supported for the IP2000, and only the parameters in the `ip-filter` subprofile are applicable.
`protocol`	Protocol number. A number of 0 (zero) matches all protocols. A nonzero number is compared to the Protocol field in each packet. For a list of assigned protocol numbers, see RFC 1700, *Assigned Numbers*.
`source-address-mask`	Mask to be applied to the `source-address` value before comparing that value to the source address of a packet.
`source-address`	IP address. After applying the `source-address-mask` value, the system compares the result to the source address in a packet. See "Filtering on source or destination IP addresses" on page 10-4.
`dest-address-mask`	A mask to be applied to the `dest-address` value before comparing that value to the destination address of a packet.
`dest-address`	IP address. After applying the `dest-address-mask` value, the system compares the result to the destination address in a packet. See "Filtering on source or destination IP addresses" on page 10-4.
`src-port-cmp`	Comparison operator to use when comparing the `source-port` value in the filter to the packet's source port value. The `less` (less than) and `gtr` (greater than) operators are not supported when comparing source port values in traffic destined for an external system. See "Filtering on port numbers" on page 10-4.
`source-port`	Port number to be compared against the source port of a packet.
`dst-port-cmp`	Comparison operator to use when comparing the `dest-port` value in the filter to the packet's destination port value. See "Filtering on port numbers" on page 10-4.
`dest-port`	Port number to be compared against the destination port of a packet.
`tcp-estab`	*Not used by the IP2000.* Setting a value for this parameter does not cause the system to display a warning message, and makes no difference to the filtering functionality.

Details of packet comparison passes

The system begins with input or output filter #1, comparing the packet to the filter specification. If the comparison fails (the packet does not match), the system proceeds to filter #2, and so forth. When a comparison succeeds, the filtering process stops and the forwarding action in that filter is applied to the packet. Within each input or output filter, the comparisons proceed as follows:

1. Apply the `source-address-mask` value to the `source-address` value and compare the result to the source address of the packet. If they are not equal, the comparison fails.
2. Apply the `dest-address-mask` value to the `dest-address` value and compare the result to the destination address in the packet. If they are not equal, the comparison fails.
3. If the `protocol` parameter is zero (which matches any protocol), the comparison succeeds. If it is nonzero and not equal to the protocol field in the packet, the comparison fails.
4. If the `src-port-cmp` parameter is not set to `none`, compare the `source-port` number to the source port number of the packet. If they do not match as specified by the `src-port-cmp` parameter, the comparison fails.
5. If the `dst-port-cmp` parameter is not set to `none`, compare the `dest-port` number to the destination port number of the packet. If they do not match as specified by the `dst-port-cmp` parameter, the comparison fails.

If all comparisons fail, the packet does not match the filter. For security purposes, the Stinger IP2000 does not automatically forward nonmatching packets unless the filter explicitly allows nonmatching packets to pass.

Filtering on source or destination IP addresses

When you specify a source or destination address in an IP filter, the system applies the filter's forwarding action to packets received from or sent to that address. If you also specify a subnet mask, the system applies the mask to the address value before comparing the resulting value to the source or destination address in a packet.

To apply the mask, the system translates both the mask and address values into binary format and then uses a logical AND to apply the mask to the address. The mask hides the bits whose positions match those of the binary zeros in the mask. A mask of all zeros (the default) masks all bits. If the address value itself is also all zeros (the default), the filter matches any source or destination address. A mask of all ones (255.255.255.255) masks no bits, so the full address for a single host is compared to the address value.

You can use the address mask to mask out the host portion of an address, for example, or the host and subnet portion, so the specification matches the address to or from any host on a given network.

Filtering on port numbers

IP filters can specify a port number to be compared to the source or destination port (or both) in a packet. A port number of zero matches nothing. TCP and UDP port numbers are typically assigned to services. For a list of well-known port assignments, see RFC 1700, *Assigned Numbers*.

Note For security purposes, Lucent Technologies recommends that you filter all services from outside your domain that are not required. UDP-based services make your network particularly vulnerable to certain types of security attacks.

The specified comparison operator determines when a match occurs. For source port values, filters applied to traffic destined for an external system support the `none` (no comparison is made) or `eql` (equal to) operators.

For other traffic, the following operators can be used to compare source port or

destination port values:

- `none` (no comparison is made)
- `eql` (equal to)
- `less` (less than)
- `gtr` (greater than)

Note The `neq` (not equal to) operator is not supported for port comparisons.

The following commands show an *illegal* rule that uses the unsupported `neq` operator to forward packets with a source port not equal to 50:

```
admin> set input-filters 1 valid-entry = yes
admin> set input-filters 1 Type = ip-filter
admin> set input-filters 1 forward = yes
admin> set input-filters 1 ip-filter protocol = 17
admin> set input-filters 1 ip-filter Src-Port-Cmp = neq
admin> set input-filters 1 ip-filter source-port = 50
```

If the filter containing this unsupported rule is applied to a connection or an interface, the system logs the following warning message when the connection or the interface goes into the UP state.

```
LOG warning, Shelf 1, Controller-1, Time: 02:23:31--
IP Filters: Not equal operation not supported for source port comparison
```

In this case, the faulty rule specifying the `neq` operator is not applied. The other rules of the filter are applied to the traffic stream.

The following commands show a legal workaround using the `less` and `gtr` comparison operator in two rules to accomplish the same effect as using the unsupported `neq` operator:

```
admin> set input-filters 1 valid-entry = yes
admin> set input-filters 1 Type = ip-filter
admin> set input-filters 1 forward = yes
admin> set input-filters 1 ip-filter protocol = 17
admin> set input-filters 1 ip-filter Src-Port-Cmp = less
admin> set input-filters 1 ip-filter source-port = 50
admin> set input-filters 2 valid-entry = yes
admin> set input-filters 2 Type = ip-filter
admin> set input-filters 2 forward = yes
admin> set input-filters 2 ip-filter protocol = 17
admin> set input-filters 2 ip-filter Src-Port-Cmp = gtr
admin> set input-filters 2 ip-filter source-port = 50
```

Explicit default filter rules

The filtering subsystem uses an implicit default rule at the end of the filtering process to affect all packets that do not match the specified input and output rules. If a filter specifies rules only in one direction, no filtering is applied to traffic in the other

direction. For example, if a filter specifies only input filter rules, all output packets are forwarded without any filtering.

The implicit default filter rule causes the system to drop all nonmatching packets in the specified direction. You can define your own explicit default rule to reverse this default filtering action. An explicit default rule has following characteristics:

- All filter fields of the default rule must be `0` (default values when the filter is created).
- The `valid-entry` field must be set to `yes`
- The `forward` entry must be set to the desired value according to the desired default behavior (`drop` or `forward`).
- The `Type` of the filter must be either `generic-filter` or `ip-filter`.

Note The IP2000 does not support generic packet filters, but you can specify a generic explicit default rule in an IP filter. This is the only supported use of `generic-filter` for connections terminating on the IP2000. A generic default rule affects *all* packet types that do not match the filter rules for a certain direction (input or output). For details, see "Sample filter using a generic explicit default rule" on page 10-7.

If a filter specifies more than one explicit default rule in the same direction, only the first one is taken into account.

Sample filter with no explicit default rule

When you do not define an explicit default rule, the system uses the implicit default which discards all packets that do not match the input and output filter rules. If a filter defines rules only in one direction, traffic in the other direction is not filtered. For example, the following sample filter specifies that UDP packets from a source port less than 50 should be forwarded. Because it does not specify an explicit default rule, this filter causes all packets that do not match input filter 1 to be dropped. Because this filter does not define output rules, all packets are forwarded in the output direction.

```
admin> new filter input-filter-1
FILTER/input-filter-1
admin> set input-filters 1 valid-entry = yes
admin> set input-filters 1 forward = yes
admin> set input-filters 1 Type = ip-filter
admin> set input-filters 1 ip-filter protocol = 17
admin> set input-filters 1 ip-filter source-address-mask = 255.255.255.255
admin> set input-filters 1 ip-filter source-address = 192.168.2.2
admin> set input-filters 1 ip-filter Src-Port-Cmp = less
admin> set input-filters 1 ip-filter source-port = 50
admin> write -f
FILTER/input-filter-1 written
```

Sample filter with explicit default rule

Defining an explicit default rule allows more flexibility in specifying which packets to drop. For example, the following sample filter specifies that UDP packets from a

source port less than 50 should be *dropped*, and includes an explicit default rule that causes all other incoming IP packets to be *forwarded*. Because this filter does not define output rules, all packets are forwarded in the output direction.

```
admin> new filter input-filter-2
FILTER/input-filter-2 read
admin> set filter-name = input-filter
admin> set input-filters 1 valid-entry = yes
admin> set input-filters 1 forward = no
admin> set input-filters 1 Type = ip-filter
admin> set input-filters 1 ip-filter protocol = 17
admin> set input-filters 1 ip-filter source-address-mask = 255.255.255.255
admin> set input-filters 1 ip-filter source-address = 192.168.2.2
admin> set input-filters 1 ip-filter Src-Port-Cmp = less
admin> set input-filters 1 ip-filter source-port = 50
admin> set input-filters 2 valid-entry = yes
admin> set input-filters 2 forward = yes
admin> set input-filters 2 Type = ip-filter
admin> write -f
FILTER/input-filter-2 written
```

Sample filter using a generic explicit default rule

To define an explicit default rule that affects *all* packets that do not match the filter rules in a certain direction, set the `Type` value to `generic-filter`.

If the explicit default filter rule is of type `ip-filter`, ARP packets (and other non-IP packets) will not be affected by the default. For example, if you want to forward all nonmatching packets including non-IP packets such as ARP packets, you must create a `generic-filter` rule as default with the action set to `forward`.

The following filter specifies a generic explicit default rule to allow forwarding of all incoming packets that do not match the input filter rules. Because the output direction does not specify an explicit default rule, all packets that do not match the output filter rules will be dropped.

```
admin> new filter input-output
FILTER/input-output read
admin> set input-filters 1 valid-entry = yes
admin> set input-filters 1 forward = no
admin> set input-filters 1 Type = ip-filter
admin> set input-filters 1 ip-filter protocol = 17
admin> set input-filters 1 ip-filter source-address-mask = 255.255.255.255
admin> set input-filters 1 ip-filter source-address = 192.168.2.2
admin> set input-filters 1 ip-filter Src-Port-Cmp = less
admin> set input-filters 1 ip-filter source-port = 50
admin> set input-filters 2 valid-entry = yes
```

```
admin> set input-filters 2 forward = yes
admin> set input-filters 2 Type = generic-filter
admin> set output-filters 1 valid-entry = yes
admin> set output-filters 1 forward = yes
admin> set output-filters 1 Type = ip-filter
admin> set output-filters 1 ip-filter protocol = 17
admin> set output-filters 1 ip-filter Src-Port-Cmp = less
admin> set output-filters 1 ip-filter source-port = 50
admin> write -f
FILTER/input-output written
```

Sample IP filters for the IP2000

The following sections present sample IP filters and illustrate some of the issues you might consider when writing your own IP filters. The sample filters presented here do not address the fine points of network security. You might want to use these filters as a starting point and augment them to address your security requirements.

Preventing IP address spoofing

Spoofing IP packets allows an intruder on a remote network to impersonate a local system's IP address. This section presents an example of an IP filter that prevents address spoofing when applied to a CPE interface.

The following commands create input filter #, which drops packets that have a local source address. In this example, the local network has an IP address of 192.100.50.128, with a subnet mask of 255.255.255.192. These values are just arbitrary examples. Because `forward` is set to `no` (the default), inbound packets with a source address on the LAN will be dropped.

```
admin> new filter ip-spoof
FILTER/ip-spoof read
admin> set input 1 valid = yes
admin> set input 1 type = ip-filter
admin> set input 1 ip-filter source-address-mask = 255.255.255.192
admin> set input 1 ip-filter source-address = 192.100.50.128
```

The next set of commands creates input filter #2, which drops packets with a source address equal to the loopback address (127.0.0.0).

```
admin> set input 2 valid = yes
admin> set input 2 type = ip-filter
admin> set input 2 ip-filter source-address-mask = 255.0.0.0
admin> set input 2 ip-filter source-address = 127.0.0.0
```

The next set of commands creates input filter #3, which explicitly accepts all remaining source addresses and forwards them to the local network. Except for `forward` = `yes`, the third filter uses all default values. Because `forward` is set to `yes`, the system forwards all remaining packets (those with nonlocal source addresses) to the LAN.

```
admin> set input 3 valid = yes
admin> set input 3 forward = yes
admin> set input 3 type = ip-filter
```

The next set of commands creates an output filter and sets the forwarding action to `yes`. This filter specifies the source mask and address for the local network. (Packets originating on the local network should be forwarded to the CPE.)

```
admin> set output 1 valid = yes
admin> set output 1 type = ip-filter
admin> set output 1 forward = yes
admin> set output 1 ip-filter source-address-mask = 255.255.255.192
admin> set output 1 ip-filter source-address = 192.100.50.128
admin> write
FILTER/ip-spoof written
```

An IP filter for more complex security issues

In this example, the local network supports a Web server, and the administrator needs to carry out the following tasks:

- Provide client access to the server's IP address.
- Restrict ingress traffic to all other hosts on the local network.

However, many local IP hosts need to access the Internet and use IP-based applications such as `telnet` or `ftp`, so their response packets need to be directed appropriately to the originating host. In this example, the Web server's IP address is 192.9.250.5. The filter will be applied in `connection` profiles as a data filter.

The following commands create the first input filter, which sets `forward` to `yes` and allows packets to reach the Web server's destination address at a destination TCP port that can be used for `telnet` or `ftp`:

```
admin> new filter web-access
FILTER/web-access read
admin> set input 1 valid = yes
admin> set input 1 forward = yes
admin> set input 1 type = ip-filter
admin> set input 1 ip-filter protocol = 6
admin> set input 1 ip-filter dest-address-mask = 255.255.255.255
admin> set input 1 ip-filter dest-address = 192.9.250.5
admin> set input 1 ip-filter dst-port-cmp = eql
admin> set input 1 ip-filter dest-port = 80
```

The next set of commands creates the second input filter, which allows inbound TCP packets in response to a local user's outbound `telnet` request, by specifying that TCP packets whose destination port number is higher than that of the source port are forwarded. (The `telnet` requests go out on port 23, and responses come back on some random port above port 1023.)

```
admin> set input 2 valid = yes
```

```
admin> set input 2 forward = yes
admin> set input 2 type = ip-filter
admin> set input 2 ip-filter protocol = 6
admin> set input 2 ip-filter dst-port-cmp = gtr
admin> set input 2 ip-filter dest-port = 1023
```

The next set of commands creates the third input filter, which allows inbound RIP updates, by specifying that inbound UDP packets are forwarded if the destination port number is higher than that of the source port. (For example, suppose a RIP packet goes out as a UDP packet to destination port 520. The response to this request goes to a random destination port above port 1023.)

```
admin> set input 3 valid = yes
admin> set input 3 forward = yes
admin> set input 3 type = ip-filter
admin> set input 3 ip-filter protocol = 17
admin> set input 3 ip-filter dst-port-cmp = gtr
admin> set input 3 ip-filter dest-port = 1023
```

The following commands create the fourth input filter. This filter uses all default values, which allows unrestricted use of `ping` and `traceroute`. Unlike TCP and UDP, ICMP does not use ports, so a port comparison is unnecessary.

```
admin> set input 4 valid = yes
admin> set input 4 forward = yes
admin> set input 4 type = ip-filter
admin> write
FILTER/web-access written
```

Applying a filter to IP interfaces

This section describes how to apply an IP filter to a PVC that terminates on the IP2000, and to the module's Ethernet interfaces.

Settings in `connection` and `ethernet` profiles

To apply a filter to an IP interface for a CPE device, set the following parameters (shown with their default settings):

```
[in CONNECTION/"":session-options]
data-filter = ""
ETHERNET { any-shelf any-slot 0 }
filter-name= ""
```

Parameter	Setting
`data-filter`	Name of a `filter` profile.
`filter-name`	Name of a `filter` profile.

Examples of applying a filter to a CPE interface

When you apply a filter in a CPE `connection` profile, it prevents certain inbound packets from reaching the LAN side of the system, or certain outbound packets from reaching the CPE router. Following is an example of applying an IP filter to a terminating PVC:

```
admin> read connection cpe-1
CONNECTION/cpe-1 read
admin> set active = yes
admin> set encapsulation-protocol = atm
admin> set ip-options remote-address = 3.3.3.200/30
admin> set atm-options vpi = 8
admin> set atm-options vci = 100
admin> set session-options data-filter = ip-spoof
admin> set atm-options nailed-group = 201
admin> write -f
CONNECTION/cpe-1 written
```

Example of applying a filter to a LAN interface

When you apply a filter in an Ethernet profile, it affects which packets are allowed to enter or leave the LAN interface. A filter applied to an Ethernet interface takes effect immediately. If you change any settings in a `filter` profile, the changes apply as soon as you save the `filter` profile.

Note Use caution when applying a filter to the Ethernet interface. You could inadvertently render the Stinger IP2000 inaccessible from the LAN.

The following set of commands applies a filter to the IP2000 Gigabit Ethernet interface:

```
admin> read ethernet { 1 8 2 }
ETHERNET/{ shelf-1 slot-8 2 } read
admin> set filter-name = ip-filter1
admin> write -f
ETHERNET/{ shelf-1 slot-8 2 } written
```

Administrative tools for filters

If you have access to the debug environment, you can use the debug-level `agrm -ifstat` command to verify that packets are being discarded and forwarded properly on a filtered interface. For details about using this command and enabling the debug environment, see “Network processor-related diagnostics” on page A-19

The system also supports the system-level `filterdisp` command for displaying information about filters in use on all terminating connections. With no command-line arguments, the command displays all active sessions and their filter names. For example,

```
admin> filterdisp
ID   Username    Src    Route-Filter  Data-Filter   Call-Filter   TOS-Filter
-------------------------------------------------------------------------
015  term        loc                 f1
021  pvc2        loc                  f1
022  pvc4        loc                  f1
023  pvc5        loc
<end user list> 4 active user(s)
```

The first column of the output displays a session ID number, followed by a username and the name of the filter. To display details for a particular session, specify the session ID as an argument on the `filterdisp` command line.

For example, the following sample output shows that no filters are applied to session 23:

```
admin> filterdisp 23
Hostname:       pvc5
No associated filters
```

The following sample output shows filters applied to an externally authenticated session:

```
admin> filterdisp 17
Hostname:       edleung
searching for external filters...
Externally obtained filters exist

Data Filter
Direction: Out

Forward = yes
Type = IP Filter
protocol = 0
source-address-mask = 0.0.0.255
source-address = 1.1.1.2
destination-address-mask = 0.0.0.0
destination-address = 0.0.0.0
Src-Port-Cmp = none
source-port = 0
Dst-Port-Cmp = none
dest-port = 0
tcp-estab = no
```

IP2000 Diagnostics

A

This appendix describes the `gmac`, `igmp`, and `pim` system-level commands. Information provided by the `gmac` command is also accessible to an external management utility through the `ip2kstats.mib` management information base (MIB). Information displayed by the pim command can also be accessed through the PIMv2 MIB (currently defined in `draft-ietf-pim-mib-v2-01.`txt) under `experimental 61`.

In addition, this appendix describes the following debug-level commands, which are not documented in the *Stinger Reference* and are not supported.

- `diag igmpsp`
- `diag igmp`
- `brtbls`
- `ifmgr`
- `diag brtbl`
- `sar`
- `agrm`

Caution Debug-level commands can be used to display low-level details about IP2000 operations. However, they are introduced into the system for development purposes, and are not part of the supported software environment. Use debug-level commands with caution!

Enabling the debug environment

To access the debug environment, log in using the default `super` (super-user) profile. For example:

```
admin> auth super
Password:
```

The default password for this account is `Ascend`. If the password has not been changed, you should change it now to prevent unauthorized super-user access to the system.

The debug environment contains many hidden commands and parameters that are not intended for general use. The debug environment is not supported, and the documentation provided here is not comprehensive.

Caution Under most circumstances, debug commands are not required for monitoring Stinger operations, and under some circumstances, these commands might produce undesirable results. Use the information with caution. Contact Lucent OnLine Customer Support at `http://www.lucent.com/support` with questions or concerns.

Gigabit Ethernet diagnostics

The `gmac` command provides diagnostic output about the Gigabit Ethernet media access controller (GMAC) driver. This command is new with the introduction of the IP2000.

gmac

Description Provide diagnostics on the Gigabit Ethernet (GigE) driver.

Permission level `system`

Usage `gmac [options]`

Command element	Description
`-v`	Show `gmac` version.
`-i [-u/d]`	With no additional option, initialize/reset the GigE port. `-i -u` Force GigE link up. `-i -d` Force GigE link down.
`-n`	Set up network processor to communicate with GMAC.
`-s`	Set up a SAR channel for communicating with GMAC .

Command element	Description
`-l [-i/e/d/p]`	Loopback.
	`-l -i` Set port for internal loopback.
	`-l -e` Set port for external loopback.
	`-l -d` Set port for no loopback
	`-l -p` Run loopback test for Ethernet power-on self test (POST).
`-p`	Ping test.
`-r`	Read a PHY register.
`-w`	Write to a PHY register.
`-d [-c/a/e]`	With no additional option, display all statistics.
	`-d -c` Clear GMAC statistics.
	`-d -a` Display ATM statistics.
	`-d -e` Display Ethernet statistics.
`-t`	Set debug level (0 through 3).
`-?`	Display a summary of commands.

Example The `gmac -n` command sets up the network processor for communication with the GMAC port:

```
admin> gmac -n
NP setup for gmac done.
```

Example The `gmac -s` command sets up a SAR channel for communicating with the GMAC port:

```
admin> gmac -s
GMAC: SAR conn. open with vpi = 0, vci = 200
```

Example The `gmac -v` command displays the GMAC version:

```
admin> gmac -v
GMAC version : 0x0b
```

Example The `gmac -i` command resets the Gigabit Ethernet port:

```
admin> gmac -i
gigE port reset.
```

Example The `gmac -d` command displays the total transmit and receive statistics for the GigE interface of the IP2000 controller. For details about the `gmac -d` output fields, see the descriptions of MIB objects in "Total transmit statistics" on page A-24 and "Total receive statistics" on page A-26. For example, the following command displays the current GMAC statistics:

```
admin> gmac -d
Gigabit Ethernet port statistics :

txOctetsLow     = 1040
txOctetsHigh    = 0
txGoodPackets   = 4
```

```
txPkt64         = 0
txPkt65127      = 0
txPkt128255     = 0
txPkt256511     = 4
txPkt5121023    = 0
txPkt1024Max    = 0
txPktDefer      = 0
txPktUndSz      = 0
txUnderFlow     = 0
txPfcf = 0
txPfcc = 0
txRfcf = 0
txRfcc = 0
txOverFlow      = 0
txAlmostFull    = 0

rxOctetsLow     = 1646718
rxOctetsHigh    = 0
rxGoodPackets   = 2059
rxPkt64         = 766
rxPkt65127      = 0
rx128255        = 0
rx256511        = 160
rx5121023       = 0
rx1024Max       = 1133
rxMacType       = 0
rxCrcErrors     = 0
rxUnderSize     = 0
rxOverSize      = 0
rxAlmostFull    = 0
rxOverRun       = 0
rxMulticastPackets      = 1896
rxBroadcastPackets      = 46
rxJabber        = 0
rxPfc   = 0
rxRfc   = 0
```

IGMP diagnostics

The `igmp` command provides information about IGMP multicast operations. The `profile` argument has been added to support `mcast-service` profiles.

The `diag igmpsp` and `diag igmp` commands are available only in the debug environment. See "Enabling the debug environment" on page A-2 for related information.

igmp

Description Displays or clears multicast information about Internet Group Management Protocol (IGMP) groups and clients.

Permission level `system`

Usage `igmp groups | clients | slots | profile| mbone | delete [`*`grp_addr`* `[`*`if_num`*`]]`

Command element	Description
`clients`	Display multicast clients.
`groups`	Display currently registered multicast group addresses and interfaces.
`slots`	Display multicast enable slots.
`profile`	Display multicast profiles.
`mbone`	Display multicast backbones.
`delete [`*`grp_addr`* `[`*`if_num`*`]]`	With the `delete` option alone, delete all currently registered multicast groups and their members. If a group address is specified, delete all members of that group. If a group address and interface number are specified, delete that member of the specified group.

Example The `igmp client` command shows local MBONE and IGMP client interfaces. For example:

```
admin> igmp client
IGMP Clients
        Client      Version  RecvCount  CLU    ALU
          2(Mbone)      2          0     0      0
          4(Mbone)      2          0     0      0
          1(Mbone)      2       3370     0      0
          5             2          0     0      0
          6             2          0     0      0
```

The output contains the following fields:

Field	Description
`Client`	In `igmp client` command output, the `Client` field displays the interface ID (the `ifIndex` value) on which the client resides. The 1 value represents the Gigabit Ethernet interface. Other numbers are WAN interfaces, numbered according to when they became active. The interfaces labeled `(Mbone)` receive multicast data from multicast routers.
`Version`	IGMP version.
`RecvCount`	Number of IGMP messages received on that interface.
`CLU/ALU`	CLU is current line utilization, and ALU is average line utilization. Both indicate the percentage of bandwidth used across this interface. If bandwidth utilization is high, some IGMP packet types are not forwarded.

Example The `igmp groups` command displays information about MBONE interfaces. Details about client member interfaces are maintained on the LIM itself. For example, the following command is invoked on the controller of a Stinger system with multicast clients on a DSL interface in slot 6 and an MBONE configured on the IP2000 Gigabit Ethernet interface:

```
admin> igmp groups
IGMP Group address Routing Table
Up Time: Od 1:25:05
Group Address   MemberIf  Expire time    Counts
239.100.100.5   *         (Mbone)        0::0 S2
                Slot 1:6
```

The next commands open a session with the DSL LIM and invoke `igmp groups` on the LIM:

```
admin> open 1 6
dadsl-atm-24-1/6> igmp groups
IGMP Group address Routing Table
Up Time: Od 1:23:46
Group Address   MemberIf  Expire time    Counts
239.100.100.5   6         00:04:07       0::0 S2
```

When the command is executed on the LIM, the output displays details about the corresponding client member interface. The output contains the following fields:

Field	Description
`Group address`	Multicast address used for the group. An asterisk indicates the IP multicast address being monitored. If a group has no members, the system forwards multicast traffic for the group to the MBONE interface (the default route).
`MemberIf`	Interface ID of multicast group members.
`Expire time`	When this membership expires. The system sends out IGMP queries every 60 seconds, so the expiration time is usually renewed. If the expiration time is reached, the system removes the entry from the table. If the field contains periods, this membership never expires. A string of periods means that the default route never times out.
`Counts`	Number of packets forwarded to the client, the number of packets dropped due to lack of resources, and the state of the membership. The state is displayed for debugging.

Example The `igmp slot` command displays information about slots supporting IGMP clients:

```
admin> igmp slot
   IGMP Client Slots
   Shelf:Slot         Group          SendCount
   1:8                230.0.0.9        0
   1:5                230.0.0.9        0
   1:2                230.0.0.9        0
```

The output contains the following fields:

Field	Description
Shelf:Slot	Shelf and slot card the MBONE connection is on.
Group	Interface number of connection.
SendCount	Number of packets sent across the interface.

Example The `igmp profile` command displays information about multicast service profiles.

```
admin> igmp profile

IGMP Service Profiles

          Service Name               : gold-service
          SNMP Trap                  : Enabled
          Call logging               : Disabled
          Filter Type                : MCAST_FILTER_INCLUSIVE
          Filter List                :
                 224.255.129.120
                 224.225.129.119

          Service Name               : bronze-service
          SNMP Trap                  : Enabled
          Call logging               : Disabled
          Filter Type                : MCAST_FILTER_INCLUSIVE
          Filter List                :
                 224.255.129.119
```

The output contains the following fields:

Field	Description
Service Name	Name of the multicast service profile.
SNMP Trap	Whether the system sends an SNMP trap when a multicast client joins or leaves a multicast group.
Call logging	Whether the system sends a call-logging packet when a multicast client session goes up or down.
Filter Type	Inclusive or exclusive multicast group filtering in the named profile.
Filter List	Multicast group addresses to be filtered.

Example The `igmp mbone` command displays information about the current MBONE interface.

```
admin> igmp mbone
   Mbone is currently:
   Slot 1:8 ifNum = 1
```

Example The `igmp delete` command deletes the specified registered multicast group and its clients.

```
admin> igmp delete 226.1.1.2
```

```
LOG notice, Shelf 1, Controller-1, Time: 10:25:53--
Multicast client 226.1.1.2 link DOWN interface number 12
LOG notice, Shelf 1, Controller-1, Time: 10:25:53--
Multicast client 226.1.1.2 link DOWN interface number 13
```

diag igmpsp

Description Enable low-level diagnostics on IGMP services profiles.

Permission level debug

Usage diag igmpsp

Example This example first uses a system-level command to view IGMP multicast service profile characteristics. It then uses the debug environment to display low-level diagnostic information.

```
admin> igmp profile
IGMP Service Profiles

          Service Name               : gold-service
          SNMP Trap                  : Disabled
          Call logging               : Disabled
          Filter Type                : MCAST_FILTER_INCLUSIVE
          Filter List                :
                 239.100.100.4
          Service Name               : bronze-service
          SNMP Trap                  : Enabled
          Call logging               : Disabled
          Filter Type                : MCAST_FILTER_EXCLUSIVE
          Filter List                :
                 239.100.100.4
```

The mcast-service profile named bronze-service has been applied to a connection profile that uses the first port of an ADSL LIM in slot 6, and the profile named gold-service has been applied to a connection on the second port of that LIM. The following commands open a session with the ADSL LIM and display diagnostics on the service profiles:

```
super> open 1 6

dadsl-atm-24-1/6> diag igmpsp
mcastJoinFilterVerify: Received CLASS D address 239.100.100.4 in report
_profileNameCompare: Compare ( bronze with bronze )
mcastJoinFilterVerify: [ bronze ] is multicast service profile for [ adsl-6-
1 ] connection
mcastJoinFilterVerify: Found 239.100.100.4 in EXCLUSIVE filter list

mcastJoinFilterVerify: Received CLASS D address 239.100.100.4 in report
_profileNameCompare: Compare ( gold-ser with gold-ser )
mcastJoinFilterVerify: [ gold-ser ] is multicast service profile for [ term-
6-2 ] connection

mcastJoinFilterVerify: Received CLASS D address 239.100.100.5 in report
_profileNameCompare: Compare ( bronze with bronze )
```

```
mcastJoinFilterVerify: [ bronze ] is multicast service profile for [ adsl-6-
1 ] connection

mcastJoinMaxClientVerify: ifnum 5, multicast membership count 0

_profileNameCompare: Compare ( bronze with bronze )
mcastJoinMaxClientVerify: [ bronze ] is multicast service for [ adsl-6-1 ]
connection
mcastClientLinkUp: Multicast client up trap

mcastJoinFilterVerify: Received CLASS D address 239.100.100.5 in report
_profileNameCompare: Compare ( bronze with bronze )
mcastJoinFilterVerify: [ bronze ] is multicast service profile for [ adsl-6-
1 ] connection
```

The following output indicates that the system rejected a request to join the group listed in an exclusive filter list in the `bronze-service` profile:

```
mcastJoinFilterVerify: Received CLASS D address 239.100.100.4 in report
_profileNameCompare: Compare ( bronze with bronze )
mcastJoinFilterVerify: [ bronze ] is multicast service profile for [ adsl-6-
1 ] connection
mcastJoinFilterVerify: Found 239.100.100.4 in EXCLUSIVE filter list
```

The next output indicates that the system accepted the message received from a client on the second LIM interface:

```
mcastJoinFilterVerify: Received CLASS D address 239.100.100.4 in report
_profileNameCompare: Compare ( gold-ser with gold-ser )
mcastJoinFilterVerify: [ gold-ser ] is multicast service profile for [ term-
6-2 ] connection
```

The following output indicates that the system accepted a request to join a group that is not listed in the exclusive filter list in the `bronze-service` profile:

```
mcastJoinFilterVerify: Received CLASS D address 239.100.100.5 in report
_profileNameCompare: Compare ( bronze with bronze )
mcastJoinFilterVerify: [ bronze ] is multicast service profile for [ adsl-6-
1 ] connection
```

diag igmp

Description Enable low-level diagnostics on IGMP protocol messages.

Permission level `debug`

Usage `diag igmp`

Example This example displays IGMP join messages from a client. The following output shows a group message to a client on LIM slot 6, port 6 and the client's response requesting to join multicast group 230.0.0.9:

```
super> diag igmp
igmp debug is ON
igmpParseMsg:   IGMP packet to 230.0.0.9 type 6 on interface 6 port 6
      Receiving Version 2 Response from 6
```

```
      Joining Group 230.0.0.9
      IGMP: Joining new group 230.0.0.9
         _sendIGMPTableUpdateMsg: sending IGMP_TAB_ADD to 1:8
         _sendUpdateMsgToShelf : client 6 join group 230.0.0.9 vRouterID 0
igmpParseMsg: IGMP packet to 230.0.0.9 type 6 on interface 6 port 6
      Receiving Version 2 Response from 6
      IGMP: Refreshing group 230.0.0.9 input ifNum 6
```

PIM-SM diagnostics

The `pim` command provides information about Protocol Independent Multicast-Sparse Mode (PIM-SM v2) operations.

pim

Description Displays PIM-related information.

Permission level `system`

Usage `pim [ group | rp | nbr | if` *`ifnum`* `| bsr | hash` *`group-addr`* `]`

Command element	Description
`groups`	Displays information about multicast groups.
`rp`	Displays group-RP mappings.
`nbr`	Displays information about PIM neighbor routers.
`if` *`ifnum`*	Displays PIM interface statistics.
`bsr`	Displays information about candidate BSRs or the elected BSR.
`hash` *`group-addr`*	Displays the best RP for a group or group range.

Example The `pim groups` command displays information about all multicast groups. For example:

```
admin> pim groups

Group Addr  RP/Source Addr upJPTimer  Tree(Rpt/Spt)
223.1.1.1   192.168.101.1  40         RPT
224.4.4.4   10.10.10.10    70         RPT
```

Output field (`pim groups`)	Description
`Group Addr`	Multicast group address for the entry
`RP/SourceAddr`	With the current software version, the RP/Source address is the RP's IP address.
`upJPTimer`	The interval in seconds following which the Stinger sends another PIM join/prune message.
`Tree(Rpt/Spt)`	With the current software version, the value of this field is always RPT (RP tree).

Example The `pim rp` command displays PIM information for candidate Rendezvous Points (C-RPs) for IP multicast groups. When the local router is the BSR, this information is obtained from received C-RP-Advertisements. When the local router is not the BSR, this information is obtained from received RP-Set messages. For example:

```
admin> pim rp
Group           RP-Address      RPF neighbor    Priority   holdtime
224.0.0.0/8       1.1.1.3         1.1.1.3           40       75:62
234.0.0.0/8       1.1.1.3         1.1.1.3           40       75:62
234.3.0.0/16      1.1.1.3         1.1.1.3           40       75:62
234.4.0.0/16      1.1.1.3         1.1.1.3           40       75:62
234.5.0.0/16      1.1.1.3         1.1.1.3           40       75:62
234.6.0.0/16      1.1.1.3         1.1.1.3           40       75:62
```

Static entries (if any) appear following a `Static Entries` label at the end of the list. Static entries represent the `pim-group-rp-mapping` profile entries. Fields in the command output have the following meaning:

Output field (`pim rp`)	**Description**
`Group`	The IP multicast group address and group mask for which this entry contains information about the C-RP.
`RP-Address`	IP address of the C-RP.
`RPF neighbor`	IP address of the Reverse Path Forwarding (RPF) neighbor-router to which a join message for this group would be directed to under certain circumstances. The RPF neighbor for an RP is calculated when the Stinger unit receives an IGMP join message for a group range that uses the RP.
`Priority`	Priority of this mapping, to be used in selecting a C-RP if multiple mappings are found for a group range.
`holdtime`	Holdtime of the C-RP. If the Stinger is the BSR, the first value in this field represents the holdtime received in the C-RP message for this RP and the second value shows the time left before this C-RP will be removed. If the Stinger is not the elected BSR, the first value in this field represents the holdtime received in BSR messages and second value will be zero.

Example The `pim nbr` command displays information about PIM neighbors on active PIM interfaces in the Stinger unit. For example:

```
admin> pim nbr
Neighbor   Interface   Priority  Holdtime   DR
1.1.101.2    2           2        100       YES
```

Fields in the command output have the following meaning:

Output field (`pim nbr`)	**Description**
`Neighbor`	The IP address of the PIM neighbor.

Output field (`pim nbr`)	Description
`Interface`	The value of `ifIndex` for the interface used to reach this PIM neighbor.
`Priority`	Hello priority of the neighbor. The 0 value indicates that the neighbor does not support the priority option, or the neighbor supports the priority option but has an assigned hello priority of 0.
`Holdtime`	Interval in seconds before the Stinger times out the neighbor if no hello message is received.
`DR`	Whether the neighbor is DR on the interface.

Example The `pim if` command displays information and statistics about the specified PIM interface. For example:

```
admin> pim if 1
pimHelloIntvl      30
pimHelloHoldtime   105
pimHelloPriority   1
pimJpIntvl         60
pimJpHoldtime      210
pimLanPruneDelay   5000
pimOdDelay         2500
pimDR               FALSE
genId              22305411
PIM Statistics
4 packets received
0 bad checksum packets received
0 bad version packets received
3 hello packet received
0 join/prune packets received
0 Boot strap packets received
1 C-RP Adv packets received
5 packets transmitted
4 hello packets sent
0 join/prune packets sent
1 boot strap packets sent
```

The statistics list the number of various packet types sent or received on this interface. They reach a maximum value and are then reset to zero. The `pimDR` field indicates whether the Stinger unit is DR on the interface. The other fields in `pim if` command output correspond to the `pim-options` configuration for the interface, as described in "PIM options in the ip-interface and connection profiles" on page 9-6.

Example The `pim bsr` command shows information about candidate BSRs within the domain if the Stinger is acting as candidate BSR, otherwise it shows information about the elected BSR. For example, the following output occurs when the Stinger unit has been elected BSR:

```
admin> pim bsr
Stinger BSR State : ELECTED_BSR
Details of CURRENT BSR:
BSR IP Address       : 1.1.1.101
BSR Interface        : 1
```

```
BSR Priority         : 255
BSR holdtime         : 57
BSR Current Frag Tag : 717
BSR HASH masklen     : 30
```

Fields in the command output have the following meaning:

Output field (`pim bsr`)	Description
`Stinger BSR State`	State of the system relevant to BSR election. If C-BSR is not enabled in the ip-global profile, this field displays `Stinger is not a C-BSR.`
`BSR IP Address`	The IP address of the bootstrap router (BSR) for the local PIM region.
`BSR Interface`	The value of `ifIndex` for the interface used to reach the BSR.
`BSR Priority`	Priority of the current BSR, from 0 to 255.
`BSR holdtime`	The bootstrap holdtime when the BSR is a C-RP in the local domain. If Stinger is elected BSR, holdtime represents the interval after which next BSM will be sent.
`BSR Current Frag Tag`	The value used in the current BSM to identify fragmented BSMs.
`BSR HASH masklen`	A value (30 by default) used to calculated the hash value for a group range when two RPs have the same priority

Example The `pim hash` command displays the IP address of the best RP for a group or group range. For example:

```
admin> pim hash 234.1.1.1
Best RP for group 234.1.1.1 is 1.1.1.10
```

To determine the best RP for a particular group, the system searches the dynamic list of RPs first. If the system does not find an RP in the dynamic list, the Stinger system looks into the static RP list. For dynamic entries, the system selects the best RP on the basis of the longest prefix match, RP priority, and the RP IP address. For static entries, the selection of best RP is based on longest prefix match.

VLAN-related diagnostics

The `brtbls` (bridge tables), `ifmgr`, `diag brtbls`, and `vlanstats` commands are available only in the debug environment. See "Enabling the debug environment" on page A-2 for related information.

brtbls

Description The `brtbls` command supports diagnostics related to VLAN bridge circuits.

Permission level `debug`

Usage brtbls [-c] |[-i *n*][-p *ifnum*]

Command element	Description
-c	Show all bridge circuits.
-i *n*	Show interfaces on bridge circuit *n*.
-p *ifnum*	Show partner information on *ifnum*.

Example This example displays bridge table information for VLAN bridge circuits.

```
super> brtbls -c
bridgeGroup          circuit      InterfaceList    items on list
     11              0x80c3f7a0        80a767b0           2
     12              0x80c3f510        80a768f0           2
2 circuits.
```

Example The next command displays interfaces on bridge circuit 11:

```
super> brtbls -i 11
    ifNum      ifType    ifName
     11           135    vlan11
     13           49      wan13
```

Example The next command shows paired bridge-circuit interface information for interface 13:

```
super> brtbls -p 13
[ifNum 13 iff 80c4aa88] <----> [ifNum 11 iff 80c4a678]
```

ifmgr

Description The ifmgr command displays interface-table entries, toggles the debug display, and marks an interface as enabled or disabled. You can enter this command only from the control module.

Permission level debug

Usage ifmgr [*options*]

Command element	Description
[-r *vrouter*]	Display routing entries. If a virtual router name is specified on the command line, the command displays only the table of the virtual router. If no virtual router name is specified, the command displays the tables for all virtual routers.
-d [*ifname*|*ifnum*]	Displays interface table entries.
-n *conn-profile name*	Display slot and interface number of the connection.
[up|down] [*ifnum*|*ifname*]	Enables or disables the specified interface.

Example The ifmgr -d command displays the interface table:

```
super> ifmgr -d
bif slot sif  u m p ifname host-name  remote-addr       local-addr
```

```
------------------------------------------------------------------------------
000 1:08 000    *   ie0        -       0.0.0.0/32      134.112.26.132/32
001 1:08 001    *   ie1        -       0.0.0.0/32       201.168.53.123/32
002 1:08 002    *   lo0        -       0.0.0.0/32           127.0.0.1/32
003 0:00 000    *   rj0        -       0.0.0.0/32           127.0.0.2/32
004 0:00 000    *   bh0        -       0.0.0.0/32           127.0.0.3/32
005 1:08 000    *   wanabe     -       0.0.0.0/32           127.0.0.3/32
006 0:00 000    *   local      -       0.0.0.0/32           127.0.0.1/32
007 0:00 000    *   mcast      -       0.0.0.0/32           224.0.0.0/32
008 0:00 000    -   tunnel0    -       0.0.0.0/32       134.112.26.132/32
009 0:00 000    *    vr0_main   -       0.0.0.0/32        134.112.26.132/32
010 0:00 000   -    sip0               0.0.0.0/32             0.0.0.0/32
011 1:06 001  *  p  wan11  adsl-6-1   30.30.7.30/32    134.112.26.132/32
012 1:06 005  *  p  wan12  adsl-6-2   10.10.7.10/32    134.112.26.132/32
013 1:14 0    *     ie1-14-1  -       0.0.0.0/32           60.60.7.60/32
014 1:14 002  *     ie1-14-2  -       0.0.0.0/32             0.0.0.0/32
015 1:06 004  *  p  wan15     pvc     0.0.0.0/32             0.0.0.0/32
016 1:14 062  *  p  wan16     ppp     20.20.7.20/32    134.112.26.132/32
<end>
```

Command element	Description
`bif`	Bundle interface number. There is one interface number per bundle, including Multilink Protocol Plus (MP+) connections. This number is the global interface-table number.
`slot`	Shelf and slot to which the interface is assigned.
`sif`	Slot interface.
`u`	Whether the interface is enabled (*) or disabled (-).
`m`	The interface is part of an Multilink Protocol (MP) bundle.
`p`	Whether the interface is permanent. A `p` indicates a permanent interface. A hyphen (-) or a blank indicates that the interface is not permanent. A permanent interface is an interface that is configured in the command-line interface and stored in Stinger NVRAM. All the Ethernet interfaces and the interfaces based on `connection` profiles are permanent. Transient interfaces are those the Stinger unit builds from RADIUS. These interfaces have no interface entry when the connection is not active.
`ifname`	Interface name. The name `ie1` is the GigE interface.
`host-name`	Hostname of remote device.
`remote-addr`	Remote address of device as configured in a `connection` profile.
`local-addr`	Address of the local interface.

Example The `next command` displays information about the Gigabit Ethernet interface:

```
super> ifmgr -d 1
iff             0x82ec86cc
inUse:          Yes
hostName:
dialoutName:
Authentication Source:  In: local       Out: local
ExternFilters:  No
ExternRoutes @  0
miscInfo @      0
reDirectDest:   0.0.0.0
DLCI routeId:   0
MP(P) id:       0
rtIf:   1:08:1
virtual id: 0, virtual next @ -1, virtual main @ -1
minor device:   1
device status:  0x221
output func:    0x801b5f18
input func:     0x801b61a0
mtu:            1500
ip_addr:        201.168.53.123
dstip_addr:     0.0.0.0
netmask:        255.255.255.0
net:            201.168.53.0
subnet:         201.168.53.0
bcast:          201.168.53.255
nbcast:         201.168.53.255
directed-bcast:         yes
management only:        no
macaddr:        00c07b65d579
inp_qcnt:       0
out_qcnt:       0
nexthop:        0.0.0.0
proxy_arp_mode: 0
proxy_arp_head: 0
vRouterID:      0
if_redirServer: 0.0.0.0
if_redirPort:   0
if_redirPort:   0
ATMP tunnel:    DISABLED
No associated connection profile
SNMP ifType:    6
multicastServiceProfile :
multicastMaxGroups      :       0
```

diag brtbls

Description Enable diagnostic `printf` input for bridge tables.

Permission level `debug`

Usage `diag brtbls`

Example The next commands enable and disable bridge table diagnostics.

```
super> diag brtbls
brtbls                        ( Bridge Tables diagnostic )
brtbls debug is ON
super> diag brtbls
brtbls                        ( Bridge Tables diagnostic )
brtbls debug is OFF
```

vlanstats

Description Display or clear VLAN statistics. For details about the `vlanstats` output fields, see the descriptions of MIB objects in "Virtual LAN (VLAN) statistics tables" on page A-28.

Permission level `debug`

Usage `vlanstats [-c] {shelf slot item} vlan-id`

Command element	Description
`-c`	Clear statistics for the specified VLAN.

Example The following command displays statistics about VLAN 1 on the first controller's GigE interface:

```
admin> vlanstats { 1 8 2 } 1
rxOctetsHigh      : 0
rxOctetsLow       : 0
rxFrames          : 0
rxUnicastFrames   : 0
rxMulticastFrames : 0
rxBroadcastFrames : 0

txOctetsHigh      : 0
txOctetsLow       : 0
txFrames          : 0
txUnicastFrames   : 0
txMulticastFrames : 0
txBroadcastFrames : 0
```

Example The following command clears the statistics for VLAN 1:

```
admin> vlanstats -c { 1 8 2 } 1
Statistics for VLAN 1 cleared
```

SAR-related diagnostics

The sar command is available only in the debug environment. See "Enabling the debug environment" on page A-2 for related information.

sar

Description Display the Segmentation and Reassembly (SAR) interface and routing tables, protocol statistics, and active sockets.

Permission level debug

Usage sar [*options*]

Command element	Description
-b	Show free transmission buffer list.
-c	Clear SAR statistics.
-i	Reset the SAR (destructive).
-l	List open channels.
-p	Configure transmission packet display.
-q	Configure receive packet display.
-m	Show SAR control memory information.
-r	Read shaper.
-s	Show SAR statistics.
-t	Execute SAR PHY test (control module).
-v	Show SAR virtual circuit table.
-w	Write shaper.
-x	List open connections.
-y	Loopback cell statistics (shelf).
-z [-d|-o|-i|-c|-a]	List open WAN connections (Stinger control module).

Example The following command analyzes contents of a frame on Gigabit Ethernet to check the VLAN ID:

```
admin> sar -p -a -100
SAR: now dumping the contents of all transmitted packets

admin> ping -c 1 20.1.2.10
PING 20.1.2.10 (20.1.2.10): 56 data bytes
tx 1/61(d) @ a1d46b00 packet len  102
TX packet: (task "_brouterPacketTask" at 0x81697c40, time: 8907.26) 102
octets @ 0xa1d46b00
  [0000]: 01 20 01 02 03 04 00 d0  52 01 02 04 81 00 00 64    . ......
R......d
```

```
  [0010]: 08 00 45 00 00 54 c9 2a  00 00 ff 01 c6 6e 14 01    ..E..T.*
.....n..
  [0020]: 02 04 14 01 02 0a 08 00  8a 47 23 a5 00 00 60 00    ........
.G#...'.
  [0030]: ea 12 00 00 00 00 00 00  00 00 00 00 00 00 00 00    ........
........
  [0040]: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00    ........
........
  [0050]: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00    ........
........
  [0060]: 00 00 00 00                                         ....
```

Network processor-related diagnostics

The `agrm` (Agere resource manager) command is available only in the debug environment. See "Enabling the debug environment" on page A-2 for related information.

agrm

Description Display diagnostics related to the Agere Network Processor Resource Manager.

Permission level `debug`

Usage `agrm [`*`options`*`]`

Command element	Description
`-sta options`	Dump statistics.
`-conn [`*`slot`*`] [`*`ifnum`*`]`	Dump connections.
`-rt [`*`slot`*`] [`*`ifnum`*`]` `-rt [`*`vrouter`*`] [-f]`	Dump routing entries. If a virtual router name is specified on the command line, the command displays only the table of the virtual router. If no virtual router name is specified, the command displays the tables for all virtual routers.
`-arp [`*`vrouter`*`]`	Dump ARP entries. If a virtual router name is specified on the command line, the command displays only the table of the virtual router. If no virtual router name is specified, the command displays the tables for all virtual routers.
`-ifstat -d|-c` *`ifnum`*	Display (`-d`) or clear (`-c`) interface statistics when debugging filter definitions on the IP2000. You can obtain the number for a given interface by using the `ifmgr -d` command in the debug environment.
`-mgrp`	Dump multicast groups.
`-did [`*`id`*`]`	Dump destination ID (DID) table contents.
`-squ` *`slot ifnum`*	Dump service queue.
`-rsc`	Dump the resources collection list.

Example The `agrm -mgrp` command displays multicast groups:

```
super> agrm -mgrp
Multicast Groups
GrpIp            Msize   Maxmtu Member    First     Last    LstInList nDrain
239.100.100.4      255     1550       1   32500    32500     32500         0
        Type Learn Tree             Act Value         Pattern
        MTU    Y   3075  0          0000060E 21  0 00107EF4 00000000
        MST    Y   3074  0          00107EF4 32  0 EF646404 00000000
```

Example The `agrm -conn` command displays network processor connections. The following command shows that an entry for the Gigabit Ethernet port in the network processor connections:

```
admin> agrm -conn
Connection Handles
if slot act cPort  dPort  type uctl    udata   qos    dctl    ddata   qos
1    8   Y  000000 000000  GE  0/200   0/200  default 0/200   0/200  default
```

Example The `agrm -rt` command displays the network processor's route table. To enable administrators to control the amount of information displayed in the routing tables, the `agrm` command supports a `-f` option for displaying the full network processor tree information. Without the `-f` option, the command displays a shorter form of the local or virtual router interface routes. For example:

```
admin> agrm -rt -f
Local If Routes
dest            gateway       if   did    needArp
0.0.0.0/0 134.112.26.1    0      0 N
        Type Learn Tree Act Value   Pattern
        LOC    Y   3074  0 000F423F  0 32 00000000 00000000
20.20.7.0/24 20.20.7.20   16      0 N
        Type Learn Tree Act Value   Pattern
        LOC    Y   3074  0 000F423F 24  8 00141407 00000000
20.20.7.20/32 20.20.7.20   16      0 N
        Type Learn Tree Act Value   Pattern
        LOC    Y   3074  0 000F423F 32  0 14140714 00000000
60.60.7.0/24 0.0.0.0   13      0 N
        Type Learn Tree Act Value   Pattern
        LOC    Y   3074  0 000F423F 24  8 003C3C07 00000000
60.60.7.60/32 0.0.0.0    6      0 N
        Type Learn Tree Act Value   Pattern
        LOC    Y   3074  0 000F423F 32  0 3C3C073C 00000000
127.0.0.0/8 0.0.0.0    4      0 N
        Type Learn Tree Act Value   Pattern
        LOC    Y   3074  0 000F423F  8 24 0000007F 00000000
127.0.0.1/32 0.0.0.0    6      0 N
        Type Learn Tree Act Value   Pattern
        LOC    Y   3074  0 000F423F 32  0 7F000001 00000000
127.0.0.2/32 0.0.0.0    3      0 N
        Type Learn Tree Act Value   Pattern
        LOC    Y   3074  0 000F423F 32  0 7F000002 00000000
134.112.0.0/16 0.0.0.0    0      0 N
        Type Learn Tree Act Value   Pattern
        LOC    Y   3074  0 000F423F 16 16 00008670 00000000
```

```
134.112.26.132/32 0.0.0.0    6      0 N
        Type Learn Tree Act Value   Pattern
        LOC    Y   3074  0 000F423F 32  0 86701A84 00000000
201.168.53.0/24 0.0.0.0    1      0 N
        Type Learn Tree Act Value   Pattern
        LOC    Y   3074  0 000F423F 24  8 00C9A835 00000000
201.168.53.123/32 0.0.0.0    6      0 N
        Type Learn Tree Act Value   Pattern
        LOC    Y   3074  0 000F423F 32  0 C9A8357B 00000000
224.0.0.0/4 0.0.0.0    7      0 N
        Type Learn Tree Act Value   Pattern
        LOC    Y   3074  0 000F423F  4 28 0000000E 00000000
224.0.0.1/32 0.0.0.0    6      0 N
        Type Learn Tree Act Value   Pattern
        LOC    Y   3074  0 000F423F 32  0 E0000001 00000000
224.0.0.2/32 0.0.0.0    6      0 N
        Type Learn Tree Act Value   Pattern
        LOC    Y   3074  0 000F423F 32  0 E0000002 00000000
224.0.0.9/32 0.0.0.0    6      0 N
        Type Learn Tree Act Value   Pattern
        LOC    Y   3074  0 000F423F 32  0 E0000009 00000000
255.255.255.255/32 0.0.0.0    0      0 N
        Type Learn Tree Act Value   Pattern
        LOC    Y   3074  0 000F423F 32  0 FFFFFFFF 00000000

Route Table
dest                  gateway      if    did    needArp
10.10.7.0/24          10.10.7.10   12    511          N
10.10.7.10/32         10.10.7.10   12    511          N
30.30.7.0/24          30.30.7.30   11    506          N
30.30.7.30/32         30.30.7.30   11    506          N
admin> agrm -rt vr1

Local If Routes for vrouter:vr1
Destination        Gateway         IF          DID   needArp
127.0.0.0/8        -               bh0_vr1     000000   N
127.0.0.1/32       -               local       000000   N
127.0.0.2/32       -               rj0_vr1     000000   N
202.168.53.0/24    -               ie1         000000   N
202.168.53.113/32  -               local       000000   N
224.0.0.9/32       -               local       000000   N

Route Table for vrouter:vr1
Destination        Gateway         IF          DID   needArp
20.20.7.0/24       20.20.7.20      wan25       000511   N
20.20.7.20/32      20.20.7.20      wan25       000511   N
202.168.53.151/32  202.168.53.151  ie1         000512   N

admin> agrm -rt vr1 -f

Local If Routes for vrouter:vr1
Destination        Gateway         IF          DID   needArp
```

```
127.0.0.0/8        -               bh0_vr1     000000   N
   Type Learn Tree Act Value   Pattern
   LOC    Y   3074  0 000F423F 18 24 0000017F 00000000
127.0.0.1/32       -               local       000000   N
   Type Learn Tree Act Value   Pattern
   LOC    Y   3074  0 000F423F 42  0 00000001 7F000001
127.0.0.2/32       -               rj0_vr1     000000   N
   Type Learn Tree Act Value   Pattern
   LOC    Y   3074  0 000F423F 42  0 00000001 7F000003
202.168.53.0/24    -               ie1         000000   N
   Type Learn Tree Act Value   Pattern
   LOC    Y   3074  0 000F423F 34  8 00000000 01CAA835
202.168.53.113/32  -               local       000000   N
   Type Learn Tree Act Value   Pattern
   LOC    Y   3074  0 000F423F 42  0 00000001 CAA83571
224.0.0.9/32       -               local       000000   N
   Type Learn Tree Act Value   Pattern
   LOC    Y   3074  0 000F423F 42  0 00000001 E0000009

Route Table for vrouter:vr1
Destination        Gateway         IF          DID   needArp
20.20.7.0/24       20.20.7.20      wan25       000511   N
20.20.7.20/32      20.20.7.20      wan25       000511   N
202.168.53.151/32  202.168.53.151  ie1         000512   N
```

Example The following `agrm -ifstat` command displays statistics for interface 1:

```
super> agrm -ifstat -d 1
Statistics for IF 1
Byte received             : 0
Unicast packet received   : 0
Input Packet discarded    : 0
Input errored packet      : 0
Input unknown protocol    : 0
Bytes transmitted         : 0
Unicast packet sent       : 0
Output packet discarded   : 0
Output packet error       : 0
```

The statistics accumulate from the time the interface enters to the UP state, until the interface is reset or the interface statistics are cleared.

Output field	Description
`Bytes received`	Number of bytes received/
`Unicast packet received`	Number of unicast packet received.
`Input Packet discarded`	Number of incoming packets discarded due to input filters.
`Input errored packet`	Number of errored packets received and discarded by the interface.
`Input unknown protocol`	Number of packet received and discarded by the interface for which the protocol was not understood.

Output field	Description
`Bytes transmitted`	Number of bytes transmitted on the interface.
`Unicast packet sent`	Number of unicast packet sent on the interface.
`Output packet discarded`	Number of packets for which the transmission has been canceled due to output filtering conditions.
`Output packet error`	Number of packets for which the transmission has been canceled due to output error conditions.

Example With the `-ifstat -c` option, you can use `agrm` to clear the interface statistics. For example:

```
super> agrm -ifstat -c 1
Statistics cleared for IF 1
```

Note If SNMP is also being used to obtain the same set of interface statistics, you should avoid clearing the statistics by using the `agrm -ifstat -c` command.

SNMP MIB for GMAC and VLAN statistics

The `ip2kstats.mib` MIB gathers statistics about the GMAC interface of the IP2000 control module, and also collects statistics on a per-VLAN basis. It is implemented as the following proprietary Lucent enterprise MIB:

```
ip2kStatsGroup   OBJECT IDENTIFIER ::= { ascend 51 }
```

The transmit and receive statistics represented in this MIB are also accessible in the command-line interface by using the `gmac -d` command. The VLAN statistics are also accessible in the command-line interface by using the debug-level `vlanstats` command.

History maintained at 15-minute intervals

The system maintains a history of the following fields by averaging them at fixed 15-minute intervals for the last 24 hours (96 intervals):

- Tx and Rx octets (transmit and receive traffic streams)
- Multicast traffic (receive side only)
- Unicast traffic (transmit and receive traffic streams)
- Broadcast traffic (receive side only)
- Cyclic redundancy check (CRC) errors (receive side only)

Gigabit Ethernet (GigE) statistics tables

Gigabit Ethernet statistics are represented in five MIB tables:

- GigE configuration
- GigE interval transmit statistics
- GigE interval receive statistics
- GigE total transmit statistics
- GigE total receive statistics

Gigabit Ethernet configuration

The `gigEConfigTable` is a configuration table for the IP2000 GigE interface. It is indexed by the interface index. This MIB table contains the objects shown in Table A-1:

Table A-1. GigEConfigTable MIB objects

MIB object	Description
`gigEValidIntervals`	The number of previous intervals for which data was collected. The value will be 96 unless the interface was brought online within the last 24 hours, in which case the value will be the number of complete 15-minute near-end intervals since the interface has been online.
`gigELastInitTime`	System time when the GigE port was last initialized. The time is represented in text format.

Interval transmit statistics

The `gigETxIntervalTable` is a table containing transmit statistics for the GigE interface in a 15-minute interval. It is indexed by the interface index and the interval number (contained in the `gigETXIntervalNumber` variable). This MIB table contains the objects shown in Table A-2:

Table A-2. GigETxIntervalTable MIB objects

MIB object	Description
`gigETxIntervalNumber`	A number between 1 and 96, where 1 is the most recently completed 15-minute interval and 96 is the 15-minute interval completed 23 hours and 45 minutes prior to interval 1.
`gigETxIntervalOctetsLow` `gigETxIntervalOctetsHigh`	The lower 32 bits and upper 32 bits of the 64-bit interval transmit packet byte counter, which contains a count of how many bytes were transmitted in error-free packets during the interval.
`gigETxIntervalUnicastPackets`	Total number of Unicast packets transmitted during the interval.

Total transmit statistics

The `gigETxTotalTable` is a table containing the total transmit statistics for the GigE interface. It is indexed by the interface index. The transmit counters in this table are

also displayed in the output of the `gmac -d` command. See "" on page A-27. This MIB table contains the objects shown in Table A-3:

Table A-3. GigETxTotalTable MIB objects

MIB object	Description
`gigETxTotalOctetsLow` `gigETxTotalOctetsHigh`	The lower 32 bits and upper 32 bits of the 64-bit transmit packet byte counter, which contains a total count of how many bytes have been transmitted in error-free packets.
`gigETxTotalGoodPackets`	Total count of packets transmitted without error.
`gigETxTotalPkt64`	Total count of transmitted 64-byte packets.
`gigETxTotalPkt65To127`	Total count of transmitted packets from 65 to 127 bytes in length.
`gigETxTotalPkt128To255`	Total count of transmitted packets from 128 to 255 bytes in length.
`gigETxTotalPkt256To511`	Total count of transmitted packets from 256 to 511 bytes in length.
`gigETxTotalPkt512To1023`	Total count of transmitted packets from 512 to 1023 bytes in length.
`gigETxTotalPkt1024ToMax`	Total count of transmitted packets from 1024 bytes up to the MAX bytes in length (1536 for non-jumbo packets and 9728 for jumbo packets).[a]
`gigETxTotalPktDefer`	Total count of packets that were deferred because the interframe gap was in excess of 96 bits when a packet was available for transmission. The IFG is a delay between code division multiple access/carrier detect (CDMA/CD) packets, intended to provide interframe recovery time for other CSMA/CD sublayers and for the physical medium.
`gigETxTotalPktUndSz`	Total count of packets less than 64 bytes in length.
`gigETxTotalUnderFlow`	Total count of packets that were truncated because of an empty FIFO.
`gigETxTotalPfcf`	Total count of Pause Flow Control packets that were sent because the receive FIFO exceeded its highwater mark. The Pause function is a mechanism for full duplex flow control.
`gigETxTotalPfcc`	Total count of Pause Flow Control packets that were sent because the client requested them.
`gigETxTotalRfcf`	Total count of Reset Flow Control packets that were sent because the receive FIFO went below its lowwater mark.

Table A-3. GigETxTotalTable MIB objects (Continued)

MIB object	Description
gigETxTotalRfcc	Total count of Reset Flow Control packets that were sent because the client requested them.
gigETxTotalOverFlow	Total number of packets in which a write from the physical signaling interface was attempted to a full transmit FIFO.
gigETxTotalAlmostFull	Total number of packets in which the transmit FIFO Almost Full flag was set.

a. Jumbo packets are not currently supported on the Gigabit Ethernet interface of the IP2000 controller.

Interval receive statistics

The gigERxIntervalTable is a table containing receive statistics for the GigE interface in a 15 minute interval. It is indexed by the interface index and the interval number (contained in the gigERXIntervalNumber variable). This MIB table contains the objects shown in Table A-4:

Table A-4. GigERxIntervalTable MIB objects

MIB object	Description
gigERxIntervalNumber	A number between 1 and 96, where 1 is the most recently completed 15 minute interval and 96 is the 15 minutes interval completed 23 hours and 45 minutes prior to interval 1.
gigERxIntervalOctetsLow gigERxIntervalOctetsHigh	The lower 32 bits and upper 32 bits of the 64-bit interval receive packet byte counter, which contains a count of how many bytes were received in error free packets during the interval.
gigERxIntervalUnicastPackets	Count of packets of the Unicast Type received during the interval.
gigERxIntervalMulticastPackets	Count of packets of the Multicast Type received during the interval.
gigERxIntervalBroadcastPackets	Count of packets of the Broadcast Type received during the interval.
gigERxIntervalCrcErrors	Count of packets that failed CRC received during the interval.

Total receive statistics

The gigERxTotalTable is a table containing the total receive statistics for the GigE interface. It is indexed by the interface index. The receive counters in this table are also displayed in the output of the gmac -d command. See "" on page A-27. This MIB table contains the objects shown in Table A-5:

Table A-5. GigERxTotalTable MIB objects

MIB object	Description
gigERxTotalOctetsLow gigERxTotalOctetsHigh	The lower 32 bits and upper 32 bits of the 64-bit receive packet byte counter, which contains a total count of how many bytes have been received in error free packets.
gigERxTotalGoodPackets	Total count of packets received without error.
gigERxTotalPkt64	Total count of received 64-byte packets.
gigERxTotalPkt65To127	Total count of received packets from 65 to 127 bytes in length.
gigERxTotalPkt128To255	Total count of received packets from 128 to 255 bytes in length.
gigERxTotalPkt256To511	Total count of received packets from 256 to 511 bytes in length.
gigERxTotalPkt512To1023	Total count of received packets from 512 to 1023 bytes in length.
gigERxTotalPkt1024ToMax	Total count of received packets from 1024 bytes up to the MAX bytes in length (1536 for non-jumbo packets and 9728 for jumbo packets).[a]
gigERxTotalMacType	Total count of MAC Type packets received
gigERxTotalCrcErrors	Total count of packets received that failed CRC.
gigERxTotalUnderSize	Total count of packets that were less than 64 bytes in length.
gigERxTotalOverSize	Total count of packets received that were greater than the MAX bytes in length (1536 for non-jumbo packets and 9728 for jumbo packets).
gigERxTotalAlmostFull	Total number of packets in which the receive FIFO exceeded its highwater mark.
gigERxTotalOverRun	Total number of receive packets in which a write to a full receive FIFO was attempted.
gigERxTotalMulticastPackets	Total number of receive packets that were of the Multicast Type.
gigERxTotalBroadcastPackets	Total number of receive packets that were of the Broadcast Type.
gigERxTotalJabber	Total count of receive packets that were classified as Jabber. Jabber is the condition of abnormally long transmissions, usually due to a fault condition.
gigERxTotalPfc	Total count of Pause Flow Control packets that were received. The Pause function is a mechanism for full duplex flow control.

Table A-5. GigERxTotalTable MIB objects (Continued)

MIB object	Description
gigERxTotalRfc	Total count of Reset Flow Control packets that were received.

a. Jumbo packets are not currently supported on the Gigabit Ethernet interface of the IP2000 controller.

Virtual LAN (VLAN) statistics tables

To capture and clear VLAN statistics, the following MIB tables are supported:

- GigE VLAN statistics
- GigE VLAN clear statistics

VLAN statistics

The GigEVlanStatTable is the VLAN statistics table for the IP2000 GigE interface. It is indexed by the interface index and by VLAN ID (contained in the gigEVlanId variable). The counters in this table are also displayed in the output of the vlanstats debug-level command. See "vlanstats" on page A-17.

This MIB table contains the objects shown in Table A-6:

Table A-6. GigEVlanStatTable MIB objects

MIB object	Description
gigEVlanId	VLAN ID. Identifies a specific VLAN on the GigE interface of the IP2000.
gigEVlanRxOctetsLow	Receive Frame Byte Counter Low Double Word.
gigEVlanRxOctetsHigh	Receive Frame Byte Counter High Double Word.
gigEVlanRxGoodFrames	Receive frame counter. Indicates the total number of frames received on the GigE interface for this VLAN.
gigEVlanRxUnicastFrames	Receive Unicast frame counter. Indicates the total number of unicast frames received for this VLAN.
gigEVlanRxMulticastFrames	Receive Multicast frame counter. Indicates the total number of multicast frames received for this VLAN.
gigEVlanRxBroadcastFrames	Receive Broadcast frame counter. Indicates the total number of broadcast frames received for this VLAN.
gigEVlanTxOctetsLow	Transmit frame byte counter Low Double word.
gigEVlanTxOctetsHigh	Transmit Frame Byte Counter High Double word.

Table A-6. GigEVlanStatTable MIB objects (Continued)

MIB object	Description
gigEVlanTxGoodFrames	Transmit frame counter. Indicates the total number of Ethernet frames transmitted on the GigE interface for this VLAN.
gigEVlanTxUnicastFrames	Unicast transmit frame counter. Indicates the total number of unicast frames transmitted for this VLAN.
gigEVlanTxMulticastFrames	Multicast transmit frame counter. Indicates the total number of multicast frames transmitted for this VLAN.
gigEVlanTxBroadcastFrames	Broadcast transmit frame counter. Indicates the total number of broadcast frames transmitted for this VLAN.

VLAN clear statistics

The GigEVlanClearStatTable is the table for clearing statistics for a given VLAN on the IP2000 GigE interface. It is indexed by the interface index and by VLAN ID (contained in the gigEVlanId variable). This table contains the objects shown in Table A-7:

Table A-7. GigEVlanClearStatTable MIB objects

MIB object	Description
gigEVlanClearStatCmd	Command to clear statistics for a particular VLAN. Values are: none(1)–No action. clearAllStats(1)–Clear all statistics. A GET operation on this object will always return none(1).

PIMv2 MIB support

The system provides SNMP MIB support for the PIM protocol as defined in draft-ietf-pim-mib-v2-01.txt (the PIMv2 MIB). The PIMv2 MIB is placed in the MIB tree under experimental 61. The current software supports PIMv2 MIB tables as shown in Table A-8:

Table A-8. Current level of support for PIMv2 MIB tables

PIMv2 MIB table	Support in this software version
pimInterfaceTable	YES
pimNeighborTable	YES
pimIpMRouteTable	NO
pimNextHopGroup	NO

Table A-8. Current level of support for PIMv2 MIB tables

PIMv2 MIB table	Support in this software version
pimRPTable	NO
pimRPSetTable	YES
pimCandidateRPTable	NO
pimComponentTable	YES

The snmpwalk, get, and getnext routines are supported for objects in the supported tables. Because Stinger units operate within a single PIM domain, the component index in pimComponentTable is always set to 1. For example, the following snmpwalk from an SNMP manager displays information from the pimInterfaceTable:

```
$ snmpwalk -m all -O 50.50.50.5 public
experimental.pimMIB.pimMIBObjects.pim.pimInterfaceTable
pimInterfaceAddress.2 = IpAddress: 1.1.1.2
pimInterfaceNetMask.2 = IpAddress: 255.0.0.0
pimInterfaceMode.2 = sparse(2)
pimInterfaceDR.2 = IpAddress: 1.1.1.2
pimInterfaceHelloInterval.2 = 60 seconds
pimInterfaceStatus.2 = active(1)
pimInterfaceJoinPruneInterval.2 = 60 seconds
pimInterfaceTrigHelloInterval.2 = 5 seconds
pimInterfaceHelloHoldtime.2 = 105 seconds
pimInterfaceLanPruneDelay.2 = on(1)
pimInterfacePropagationDelay.2 = 7500 milliseconds
pimInterfaceOverrideInterval.2 = 2500
pimInterfaceGenerationID.2 = on(1)
pimInterfaceJoinPruneHoldtime.2 = 210 seconds
```

The following snmpwalk command displays information from the pimNeighborTable:

```
$ snmpwalk -m all -O 50.50.50.5 public
experimental.pimMIB.pimMIBObjects.pim.pimNeighborTable
pimNeighborIfIndex.167837953 = 1
pimNeighborExpiryTime.167837953 = Timeticks: (98) 0:00:00.98
pimNeighborMode.167837953 = sparse(2)
pimNeighborLanPruneDelay.167837953 = 0
pimNeighborOverrideInterval.167837953 = 0
pimNeighborTBit.167837953 = off(0)
pimNeighborDRPresent.167837953 = true(1)
```

The following command from an SNMP manager displays information from the pimRPSetTable:

```
$ snmpwalk -m all -O s 50.50.50.5 public
experimental.pimMIB.pimMIBObjects.pim.pimRPSetTable
pimRPSetHoldTime.1.224.255.50397441 = 75
pimRPSetHoldTime.1.234.255.50397441 = 75
pimRPSetHoldTime.1.1002.65535.50397441 = 75
pimRPSetHoldTime.1.1258.65535.50397441 = 75
pimRPSetHoldTime.1.1514.65535.50397441 = 75
pimRPSetHoldTime.1.1770.65535.50397441 = 75
pimRPSetExpiryTime.1.224.255.50397441 = Timeticks: (56) 0:00:00.56
```

```
pimRPSetExpiryTime.1.234.255.50397441 = Timeticks: (56) 0:00:00.56
pimRPSetExpiryTime.1.1002.65535.50397441 = Timeticks: (56) 0:00:00.56
pimRPSetExpiryTime.1.1258.65535.50397441 = Timeticks: (56) 0:00:00.56
pimRPSetExpiryTime.1.1514.65535.50397441 = Timeticks: (56) 0:00:00.56
pimRPSetExpiryTime.1.1770.65535.50397441 = Timeticks: (56) 0:00:00.56
```

Following is sample output of an `snmpwalk` on `pimComponentTable`:

```
$ snmpwalk -m all -O s 50.50.50.5 public
experimental.pimMIB.pimMIBObjects.pim.pimComponentTable
pimComponentBSRAddress.1 = IpAddress: 1.1.1.101
pimComponentBSRExpiryTime.1 = Timeticks: (45) 0:00:00.45
pimComponentCRPHoldTime.1 = 0 seconds
pimComponentStatus.1 = active(1)
```

Index

A

B

C

D

E

F

G

H

I

L

M

N

O

P

Q

R

S

T

U

V

W